PXM

PROFIBUS DPV0/DPV1 Master

User Manual

PMEPXM0100

PMEPXM0100H

08/2019

Revision 1.04

ProSoft® TECHNOLOGY

Note: Before installing, configuring, operating, or maintaining the PMEPXM0100(H) products, please review this information and the information located on:
https://www.schneider-electric.com/en/product/PMEPXM0100 or
https://www.schneider-electric.com/en/product/PMEPXM0100H
for the latest software, documentation, and installation files specific to the PMEPXM0100(H) products.
For additional support, please contact Schneider Electric at
https://www.schneider-electric.com/en/work/support/
Installation and maintenance of the PMEPXM0100(H) products should be carried out by suitably trained personnel in accordance with applicable codes of practice. In case of malfunction or damage, no attempts of repair should be made. Your PMEPXM0100(H) product(s) should be returned for repair. Do not dismantle the product.

Notice

Read these instructions carefully and look at the equipment to become familiar with the device before trying to install, operate, service, or maintain it. The following special messages may appear throughout this documentation or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.

The addition of this symbol to a “Danger” or “Warning” safety label indicates that an electrical hazard exists which will result in personal injury if the instructions are not followed.

This is a safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

⚠ *DANGER*

DANGER indicates a hazardous situation which, if not avoided, **will result in** death or serious injury.

⚠ *WARNING*

WARNING indicates a hazardous situation which, if not avoided, **could result in**, death or serious injury.

⚠ *CAUTION*
CAUTION indicates a hazardous situation which, if not avoided, **could result in** minor or moderate injury.

NOTICE
NOTICE is used to address practices not related to physical injury.

PLEASE NOTE

Electrical equipment should be installed, operated, serviced, and maintained only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.
A qualified person is one who has skills and knowledge related to the construction and operation of electrical equipment and its installation and has received safety training to recognize and avoid the hazards involved.

BEFORE YOU BEGIN

Do not use this product on machinery lacking effective point-of-operation guarding. Lack of effective point-of-operation guarding on a machine can result in serious injury to the operator of that machine

⚠ *WARNING*
UNGUARDED EQUIPMENT • Do not use this software and related automation equipment on equipment which does not have point-of-operation protection. • Do not reach into machinery during operation. **Failure to follow these instructions can result in death, serious injury, or equipment damage.**

This automation equipment and related software is used to control a variety of industrial processes. The type or model of automation equipment suitable for each application will vary depending on factors such as the control function required, degree of protection required, production methods, unusual conditions, government regulations, etc. In some applications, more than one processor may be required, as when backup redundancy is needed.

Only you, the user, machine builder or system integrator can be aware of all the conditions and factors present during setup, operation, and maintenance of the machine and, therefore, can determine the automation equipment and the related safeties and interlocks which can be properly used. When selecting automation and control equipment and related software for a particular application, you should refer to the applicable local and national standards and regulations. The National Safety Council's Accident Prevention Manual (nationally recognized in the United States of America) also provides much useful information.

In some applications, such as packaging machinery, additional operator protection such as point- of-operation guarding must be provided. This is necessary if the operator's hands and other parts of the body are free to enter the pinch points or other hazardous areas and serious injury can occur. Software products alone cannot protect an operator from injury. For this reason, the software cannot be substituted for or take the place of point-of-operation protection.

Ensure that appropriate safeties and mechanical/electrical interlocks related to point-of-operation protection have been installed and are operational before placing the equipment into service. All interlocks and safeties related to point-of-operation protection must be coordinated with the related automation equipment and software programming.

NOTE: Coordination of safeties and mechanical/electrical interlocks for point-of-operation protection is outside the scope of the Function Block Library, System User Guide, or other implementation referenced in this documentation.

START-UP AND TEST

Before using electrical control and automation equipment for regular operation after installation, the system should be given a start-up test by qualified personnel to verify correct operation of the equipment. It is important that arrangements for such a check be made and that enough time is allowed to perform complete and satisfactory testing

⚠ ***WARNING***
EQUIPMENT OPERATION HAZARD Verify that all installation and set-up procedures have been completed. • Before operational tests are performed, remove all blocks or other temporary holding means used for shipment from all component devices. • Remove tools, meters, and debris from equipment. **Failure to follow these instructions can result in death, serious injury, or equipment damage.**

Follow all start-up tests recommended in the equipment documentation. Store all equipment documentation for future references.

Software testing must be done in both simulated and real environments.
Verify that the completed system is free from all short circuits and temporary grounds that are not installed according to local regulations (according to the National Electrical Code in the U.S.A, for instance). If high-potential voltage testing is necessary, follow recommendations in equipment documentation to prevent accidental equipment damage.
Before energizing equipment:

- Remove tools, meters, and debris from equipment.

- Close the equipment enclosure door.
- Remove all temporary grounds from incoming power lines.
- Perform all start-up tests recommended by the manufacturer.

OPERATION AND ADJUSTMENTS

The following precautions are from the NEMA Standards Publication ICS 7.1-1995 (English version prevails):

- Regardless of the care exercised in the design and manufacture of equipment or in the selection and ratings of components, there are hazards that can be encountered if such equipment is improperly operated.
- It is sometimes possible to misadjust the equipment and thus produce unsatisfactory or unsafe operation. Always use the manufacturer's instructions as a guide for functional adjustments. Personnel who have access to these adjustments should be familiar with the equipment manufacturer's instructions and the machinery used with the electrical equipment.
- Only those operational adjustments actually required by the operator should be accessible to the operator. Access to other controls should be restricted to prevent unauthorized changes in operating characteristics.

For professional users in the European Union

If you wish to discard electrical and electronic equipment (EEE), please contact your dealer or supplier for further information.

North American Hazardous Location Approval

SUITABLE FOR USE IN CLASS I, DIVISION 2, GROUPS A, B, C AND D HAZARDOUS LOCATIONS, OR NONHAZARDOUS LOCATIONS ONLY.

WARNING - EXPLOSION HAZARD - DO NOT DISCONNECT EQUIPMENT WHILE THE CIRCUIT IS LIVE OR UNLESS THE AREA IS KNOW TO BE FREE OF IGNITABLE CONCENTRATIONS.

WARNING - EXPLOSION HAZARD - SUBSTITUTION OF ANY COMPONENT MAY IMPAIR SUITABILITY FOR CLASS I, DIVISION 2.

ADAPTÉ POUR UNE UTILISATION EN CLASSE 1, DIVISION 2, GROUPES A, B, C ET D LIEUX DANGEREUX OU EXCLUSIVEMENT EMPLACEMENT NON DANGEREUX

AVERTISSEMENT - RISQUE D'EXPLOSION - NE PAS DECONNECTER L'EQUIPEMENT LORSQUE LE CIRCUIT EST ALIMENTE, A MOINS QUE LA ZONE SOIT CONTROLEE ABSENTE DE CONCENTRATION INFLAMMABLES.

AVERTISSEMENT - RISQUE D'EXPLOSION - REMPLACEMENT DE TOUT COMPOSANT PEUT NUIRE A LA CONFORMITÉ DE CLASS I, DIVISION 2.

ATEX Warnings and Conditions of Safe Usage

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

DO NOT OPEN WHEN ENERGIZED.

⚠ *CAUTION*
POSSIBLE HOT SURFACE Certain surfaces may be hot. **Failure to follow these instructions can result in injury or equipment damage.**

China RoHS Declaration Table

部件名称 Part Name	有害物质 - Hazardous Substances					
	铅 (Pb)	汞 (Hg)	镉 (Cd)	六价铬 (Cr (VI))	多溴联苯 (PBB)	多溴二苯醚 (PBDE)
金属部件 Metal parts	X	O	O	O	O	O
塑料部件 Plastic parts	O	O	O	O	O	O
电子件 Electronic	O	O	O	O	O	O
触点 Contacts	O	O	X	O	O	O
线缆和线缆附件 Cables & cabling accessories	X	O	O	O	O	O

本表格依据 SJ/T11364 的规定编制。

O: 表示该有害物质在该部件所有均质材料中的含量均在 GB/T 26572 规定的限量要求以下。

X: 表示该有害物质至少在该部件的某一均质材料中的含量超出 GB/T 26572 规定的限量要求。

This table is made according to SJ/T 11364.
O: Indicates that the concentration of hazardous substance in all of the homogeneous materials for this part is below the limit as stipulated in GB/T 26572.
X: Indicates that concentration of hazardous substance in at least one of the homogeneous materials used for this part is above the limit as stipulated in GB/T 26572.

CONTENTS

Revision History

Revision	Date	Comment
1.03	17 April 2019	Initial release
1.04	28 August 2019	Added PCM Enhanced IO Scan Option in EcoStruxure Control Expert Export.

1. PREFACE

1.1. INTRODUCTION TO THE PXM

This manual describes the installation, operation, and diagnostics of the ProSoft PXM PROFIBUS DPV0/DPV1 Master – PMEPXM0100 or PMEPXM0100H. The PXM allows the user to integrate PROFIBUS DP slave devices into Schneider Electric's M580 Control System. This will allow the M580 Control System to exchange process, alarming, and diagnostic data with PROFIBUS DP devices as well as provide parameterization and asset management of slave devices using Device Type Managers (DTMs).

1.2. PREREQUISITES

The PXM module operates in the M580 System with following requirements:

- EcoStruxure Control Expert: V14 or greater
  (+ ControlExpert_V140_HF_PMEPXM0100 hotfix)
- M580 CPU Firmware: 2.80 or greater

1.3. FEATURES

The PXM can exchange process data (DPV0) with up to 125 PROFIBUS DP slave devices which will be formatted into the engineering units in the M580 Control System by using the automatically generated Control Expert mapping imports.

The ProSoft Configurator for Modicon will allow the user to configure each PXM as well as each PROFIBUS DP slave device connected to the PXM for DPV0 communication. The utility will also automatically generate the mapping routines and structures (in either Function Block or Structured Text) which can be imported into Control Expert.

The PXM also provides DPV1 communication allowing the user to exchange DPV1 Class 1 and Class 2 data with each slave device. The PXM Gateway DTM can be used to configure and parameterize each slave device using Device Type Manager (DTM) technology.

The PXM will allow the user to monitor and extract DPV1 alarms from each slave device on the connected PROFIBUS DP fieldbus from the M580 controller.

The PXM provides a range of statistics and tools to provide a detailed diagnostic overview of each PXM which speeds-up system commissioning. The Configuration Utility allows the user to do a PROFIBUS DP packet capture of the running fieldbus which can be used to analyse the

bus behaviour and packets received. The PXM also provides global and device specific statistics.

Each PXM connection to the M580 controller can be customized to the required data size. This provides the user with a range of EtherNet/IP connection sizes and counts to limit the amount of memory used by each PXM.

The PXM also allows the user to customize the required security level for an application by enabling or disabling certain protocols as well as having a configurable Access Control List. In addition to this the PXM can log up to 2048 events into non-volatile memory (NV) which can later be offloaded to a SysLog Server.

The PXM can be used in one of two modes; Standalone or HSBY.

Standalone
In this mode a single PXM is connected a single M580 controller. The PXM can be run in the local rack or a remote rack (using either the controller connection or NOC).

HSBY
In this mode the PXM can be used in a redundant M580 Control Architecture. Each PXM will be located in the local rack of each redundant M580 controller. This will allow the PXM to switch with the M580 controllers in an HSBY system when needed. The PXM will provide a bumpless transfer when switching from Primary PXM to Standby PXM when a switch over event occurs.

Safety
The module cannot be mounted on a rack with safety modules as it is a non-interfering class 2 device.

1.4. ARCHITECTURE

The PXM can be configured to operate in one of three architectures; Standalone, HSBY, or Remote.

The figure below provides an example of the typical network setup in a Standalone architecture.

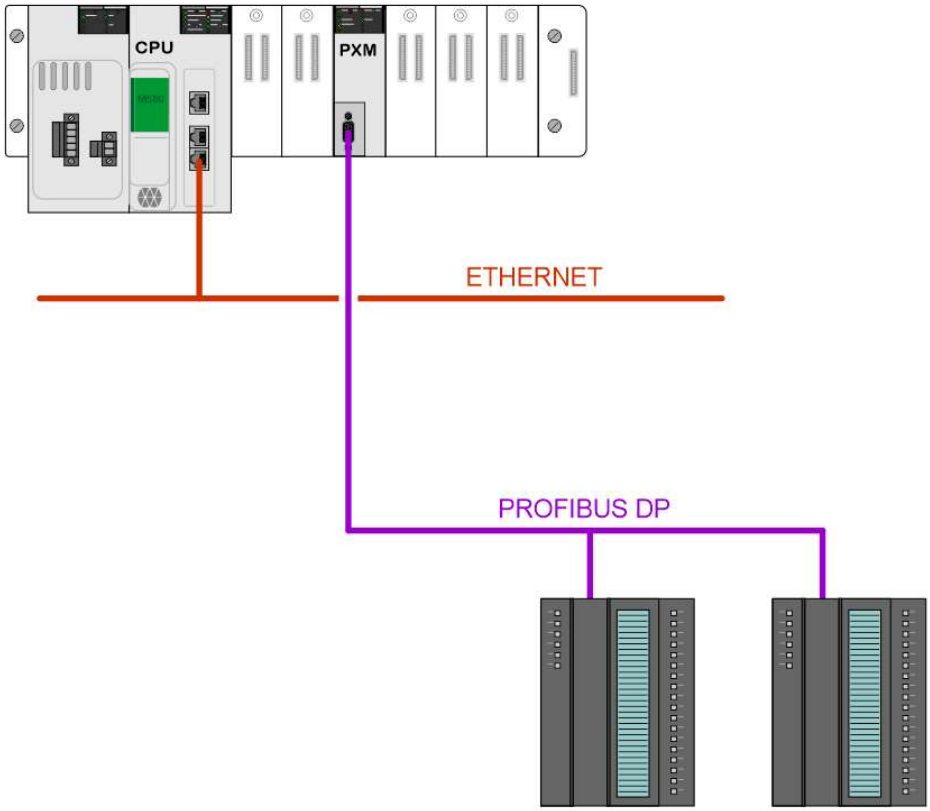


Figure 1.1 - PXM Standalone architecture

Alternatively, the PXM can be configured to operate in a Remote Rack of the M580 system.

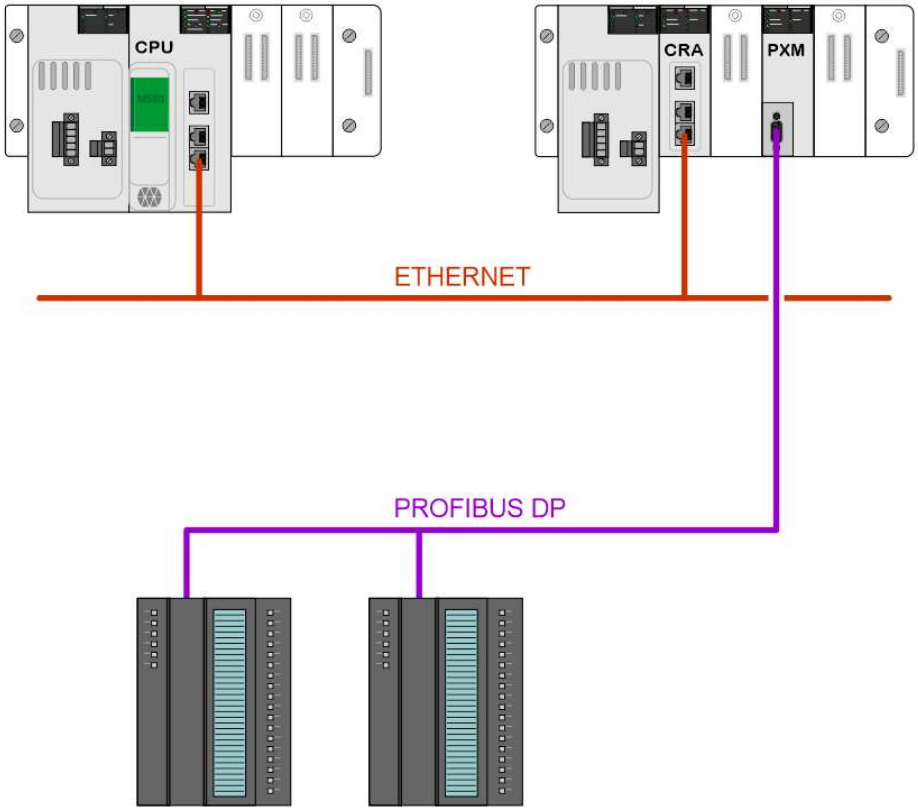


Figure 1.2 - PXM Standalone architecture in a Remote Rack

When the M580 control system is operating in a redundant HSBY architecture, each PXM will operate in the local rack as shown below:

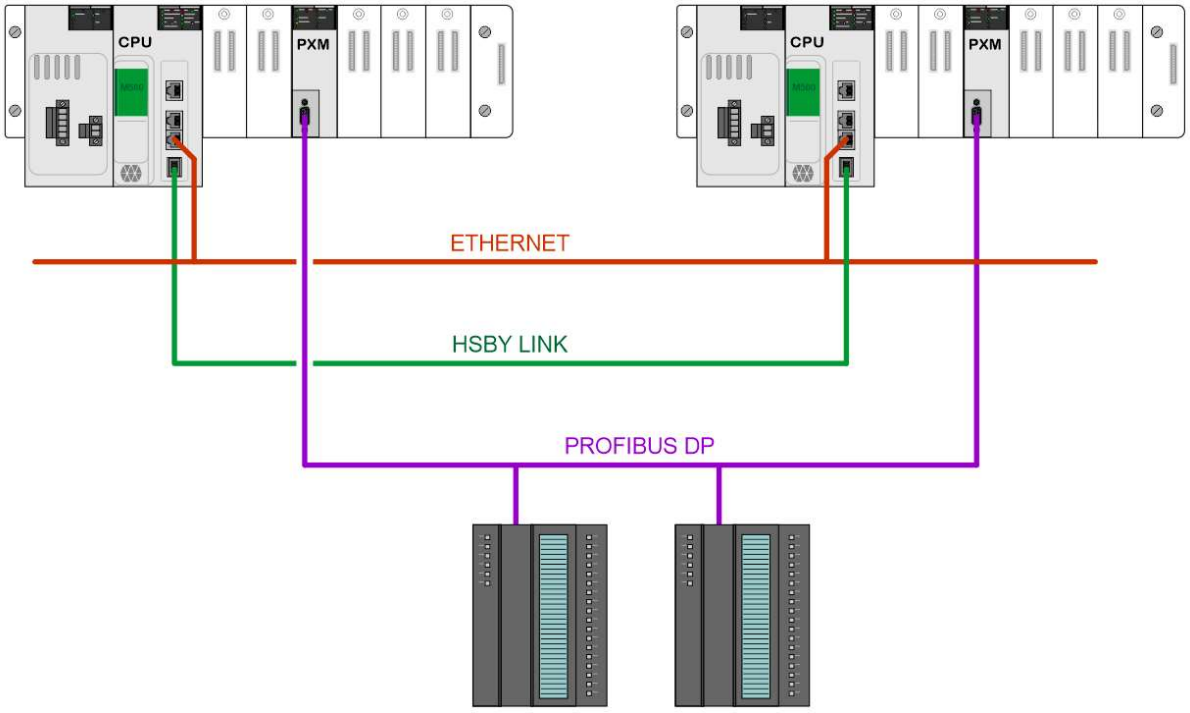


Figure 1.3 - PXM HSBY architecture

1.5. ADDITIONAL INFORMATION

The following web sites contain additional information that can assist the user with the module installation and operation, including the required ProSoft Configurator for Modicon configuration software.

Resource	Link
PMEPXM0100	https://www.schneider-electric.com/en/product/PMEPXM0100
PMEPXM0100H	https://www.schneider-electric.com/en/product/PMEPXM0100H

Table 1.1 - Additional Information

1.6. SUPPORT

Technical support is provided via the Web (in the form of user manuals, FAQ, datasheets etc.) to assist with installation, operation, and diagnostics.

For additional support the user can use either of the following:

Resource	Link
Contact Us web link	https://www.schneider-electric.com/en/work/support/

Table 1.2 – Support Details

2. INSTALLATION

2.1. MODULE LAYOUT

The PXM has one RS485 PROFIBUS DP port at the front of the module and one M580 backplane port at the back of the module, as shown in the figure below. The front port is used to connect to the PROFIBUS DP fieldbus and the backplane port is used to connect to the M580 backplane. All the required power is derived from the M580 backplane.

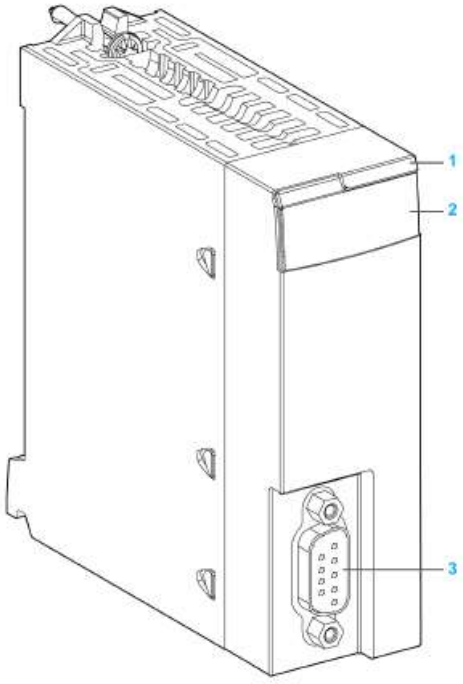


Figure 2.1 - PXM Front and Side view

Number	Element	Function
1	Module name	ePXM0100 (Standard) ePXM0100H (Harsh)
2	LED array	LED indication to diagnose the module
3	SUB-D 9 female connector	PROFIBUS DP Port

Table 2.1 – Module layout

The module provides seven diagnostic LEDs as shown in the front view figure below. These LEDs are used to provide information regarding the module system operation, the Backplane

interface, and the PROFIBUS DP fieldbus interface. See the *Diagnostics* section for details on each LED state.

The module provides two DIP-switches at the back of the enclosure as shown in the figure below.

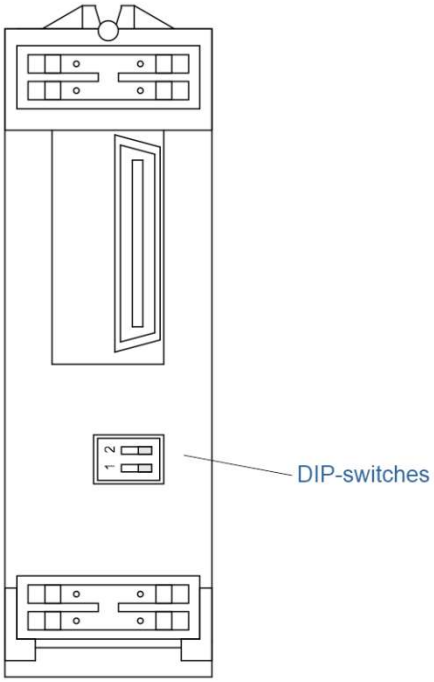


Figure 2.2 - PXM Back view

DIP Switch	Description
DIP Switch 1	This DIP switch is used to reject the configuration in NV memory as well as the configuration received from the Head module (using TFTP). This action resets the module to Factory Defaults. The module will then wait for new configuration to be downloaded to it.
DIP Switch 2	Used to force the module into “Safe Mode”. When in “Safe Mode” the module will not load the application firmware and will wait for new firmware to be downloaded. This should only be used in the rare occasion when a firmware update was interrupted at a critical stage.

Table 2.2 - DIP Switch Settings

2.2. MODULE MOUNTING

The PXM module will connect directly to the M580 backplane. Note that it does not use the X80 backplane connector and will only power up on an Ethernet backplane. In an M580 architecture, you can mount the PXM module on a local rack or a remote drop.

Follow the step below to mount the module on the backplane:

a. Insert the locating pins on the bottom of the module into the corresponding slots in the rack.

b. Use the locating pins as a hinge and pivot the module until it is flush with the rack. (The twin connector on the back of the module inserts the connectors on the rack).

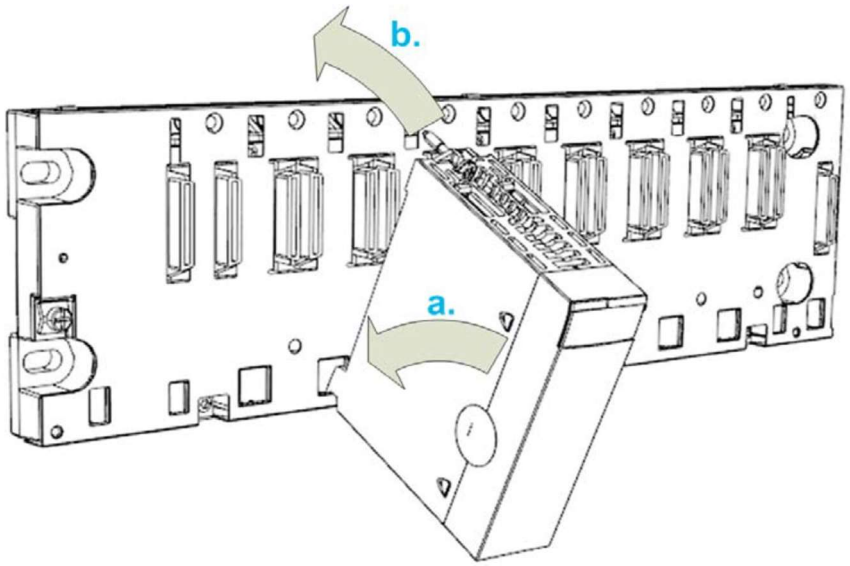


Figure 2.3 – Mounting the PXM to the backplane

Tighten the retaining screw to hold the module in place on the rack:

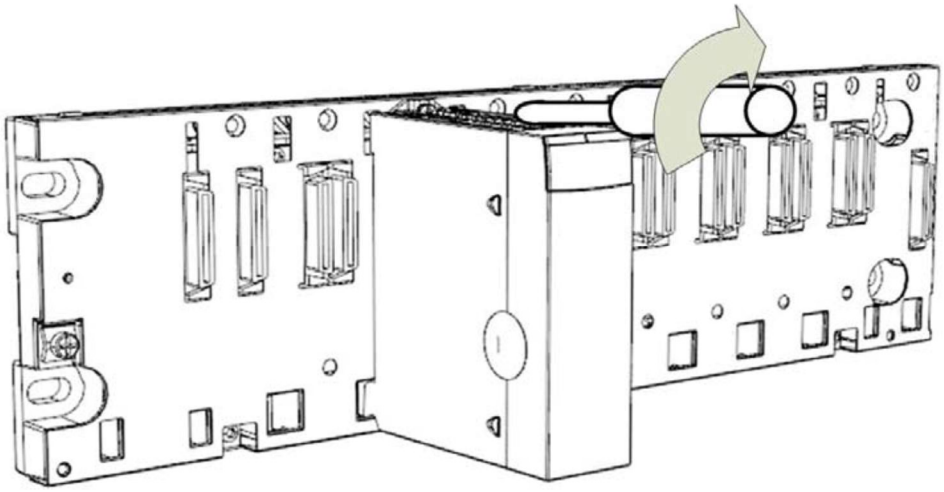


Figure 2.4 – Tightening the PXM to the backplane

NOTICE
EQUIPMENT DAMAGE The maximum tightening torque is 1.5 Nm (1.11 lb-ft). **Failure to follow these instructions may result in equipment damage.**

2.3. BACKPLANE CONNECTOR

The Ethernet bus interface at the back of the PXM module connects to the Ethernet backplane connector when you mount the module in the rack (see *Module Mounting* section). The module is powered by the backplane. It is hot swappable, that is, it may be installed and uninstalled without turning off the power supply to the rack.

The X Bus connector of the backplane is not present nor required. The module uses the Ethernet bus on the Ethernet backplane to manage the connectivity to the Ethernet I/O scanner.

The module communicates with a PC that is connected to the Ethernet network using an asset management, a network manager, or a web browser.

2.4. PROFIBUS DP PORT (RS485)

The PROFIBUS DP port uses a female DB9 connector. This provides connection for the communication conductors, cable shielding and +5Vdc output power.

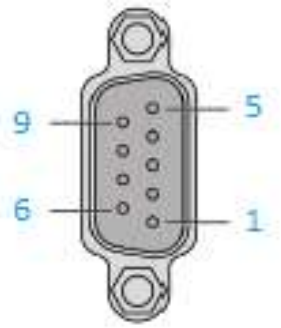


Figure 2.5 - PXM PROFIBUS DP (RS485) DB9 connector

Pin	Signal	Description
1	-	Not connected
2	-	Not connected
3	RxD/TxD-P	Data received and transmit (+)
4	CNTR-P	Control signal to repeater (+)
5	DGND	Reference potential for +5Vdc
6	VP	+5Vdc for terminating resistors (active termination)
7	-	Not connected
8	RxD/TxD-N	Data received and transmit (-)
9	-	Not connected

Table 2.3 – DB 9 Connector layout

3. SETUP

3.1. SETUP INTRODUCTION

The setup of the PXM requires configuration in both Control Expert and the ProSoft Configurator for Modicon (PCM).

The figure below provides an overview of the required steps to configure a new PXM module.

Although it is not important whether the user starts with the Control Expert configuration or the PCM configuration, it is important that the Control Expert configuration is transferred to the M580 controller, before the PXM can be downloaded.

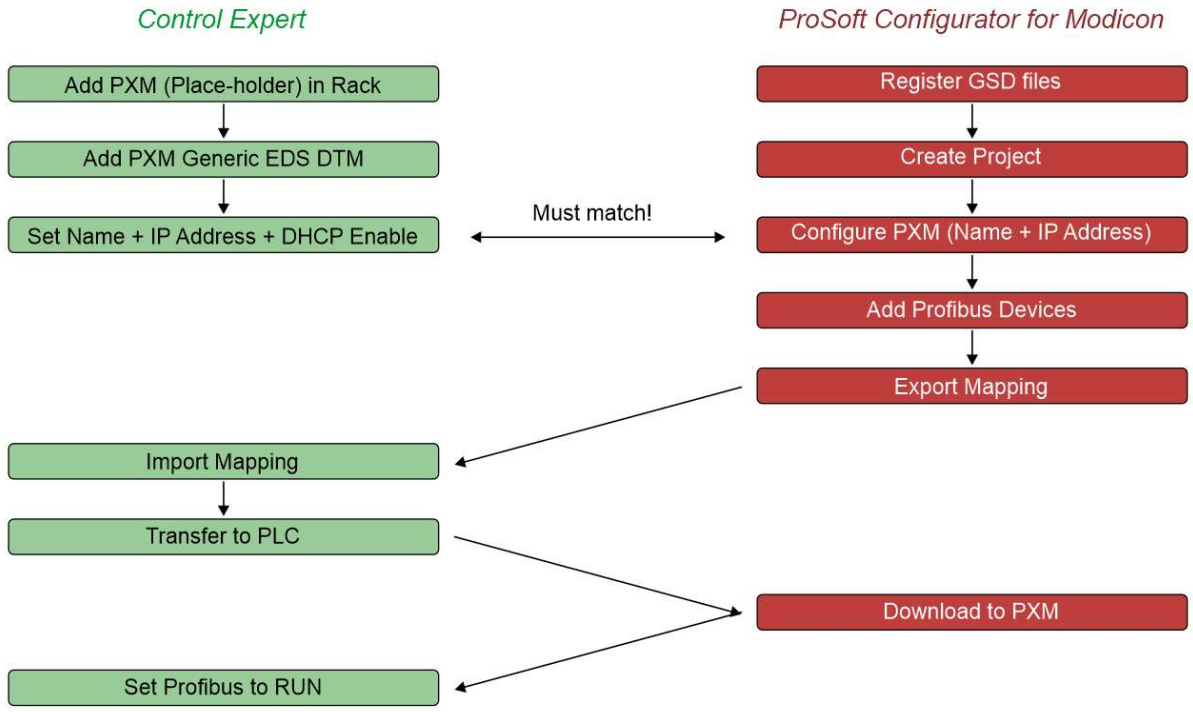


Figure 3.1 – PXM Configuration Overview

3.2. INSTALL CONFIGURATION SOFTWARE

All the PROFIBUS network setup and configuration of the PXM module is achieved by means of the ProSoft Configurator for Modicon.

Figure 3.2 - ProSoft Configurator for Modicon Installation

3.3. NETWORK PARAMETERS

The module Ethernet network parameters (e.g. IP address) will be managed by the Head module in the local M580 rack. See the *Control Expert Configuration* section.

3.4. GSD FILE MANAGEMENT

Each PROFIBUS device has a GSD file that is required to provide information needed to configure the device for data exchange. The ProSoft Configurator for Modicon manages the GSD library which is used for adding devices to the PXM.

The GSD File Management Tool is opened by selecting *GSD File Management* under the Tool menu in the configuration utility.

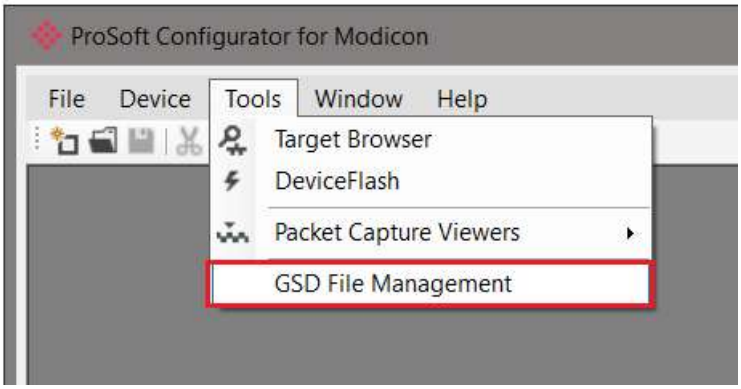


Figure 3.3 – Launching the GSD File Management Tool

Once the tool has been opened a list of slave devices already registered using their GSD files.

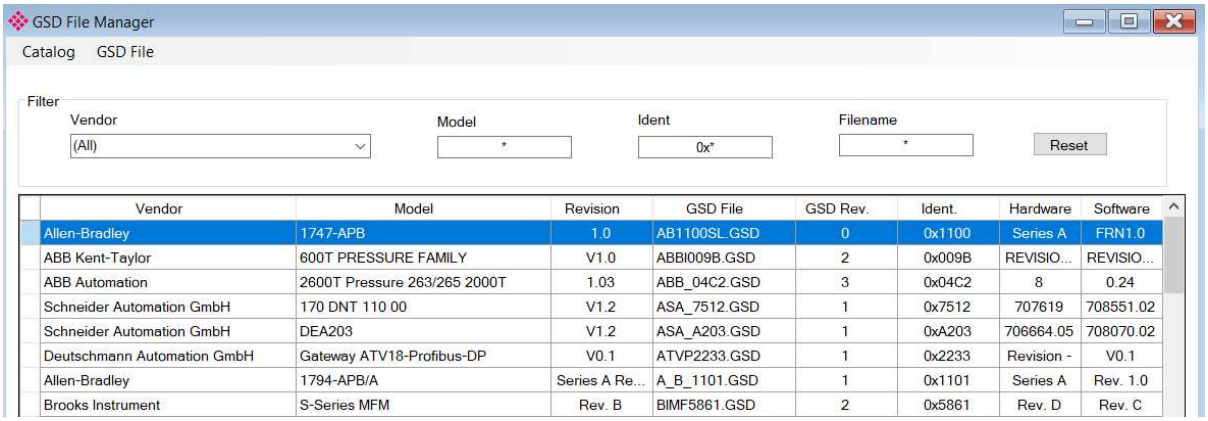


Figure 3.4 – GSD File Management Tool

To add a GSD file the user will need to select the *Add* option under the GSD File menu.

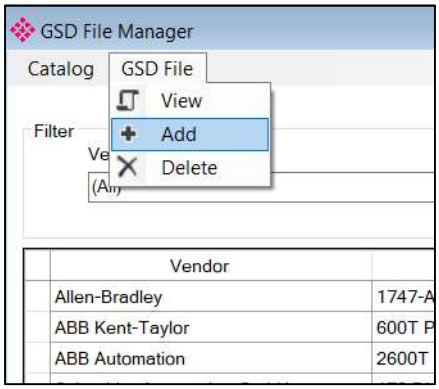


Figure 3.5 – GSD File Adding

The required GSD file will need to be selected as shown below:

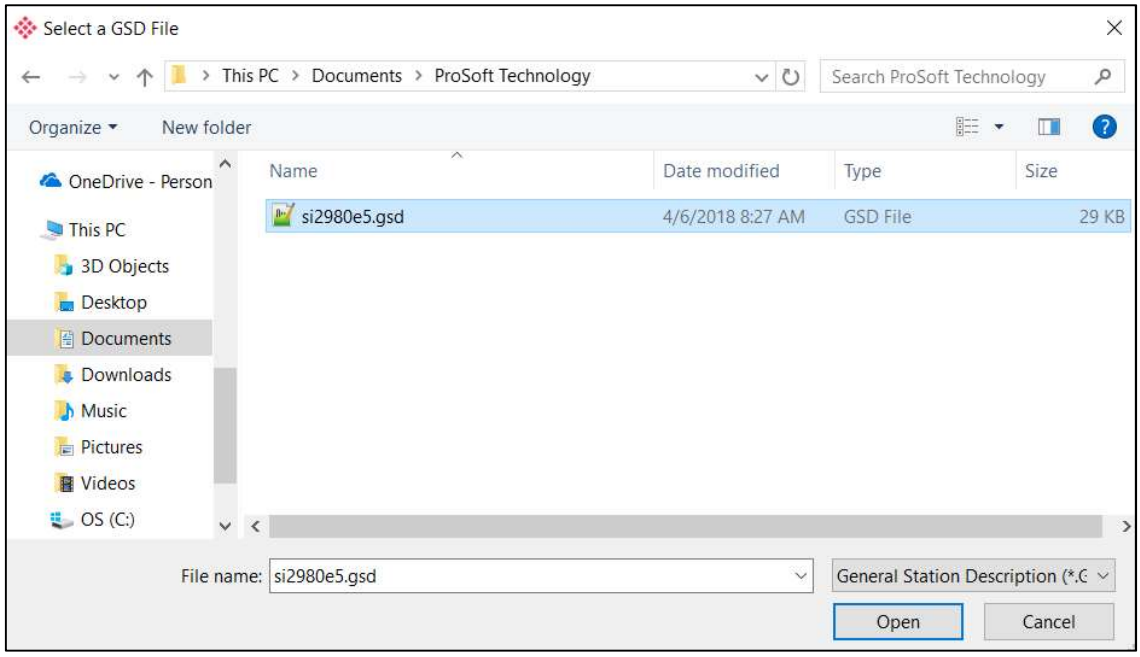


Figure 3.6 – GSD File Adding

Once the file has been selected the GSD File Management tool will add the slave device to the device list and recompile the GSD catalog.

A GSD catalog can be reused by another ProSoft Configurator for Modicon by exporting the GSD catalog on one ProSoft Configurator for Modicon and importing it in another. This is done by selecting either *Import* or *Export* under the Catalog menu as shown below:

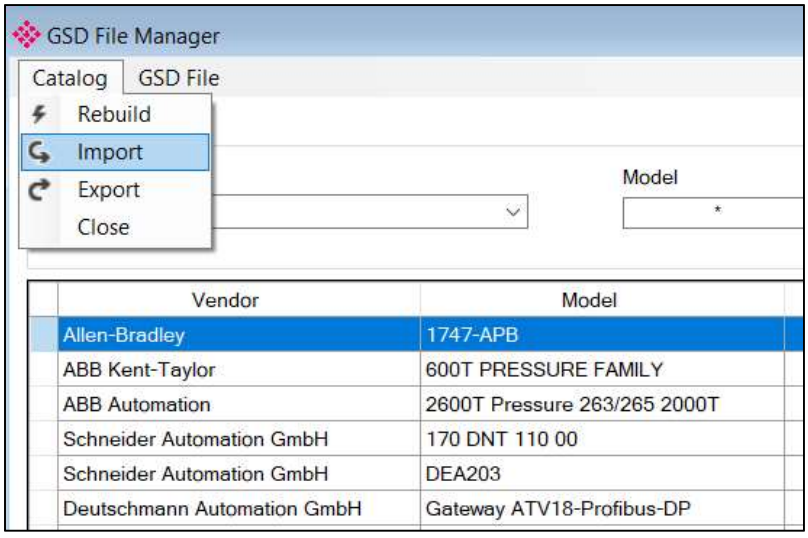


Figure 3.7 – GSD Catalog importing

3.5. PXM TYPE LIBRARY

The PXM requires the use of a number of system DFBs and DDTs. These are contained within the PXM Type Library which must be registered before using a PXM in a Control Expert application.

3.5.1. COMPATIBILITY

The PXM Type Library has the following minimum requirements:

- EcoStruxure Control Expert: V14 or greater
- M580 CPU Firmware: 2.80 or greater

3.5.2. INSTALLATION

Download the PXM Type Library and unzip it to a suitable folder.

In the Windows Start menu, under the ***Schneider Electric*** … ***EcoStruxure***, select the ***Types Library Update*** item.

Figure 3.8 – Launch Type Library Update utility

NOTICE
UNEXPECTED BEHAVIOUR A Type Library cannot be registered if Control Expert is running. Be sure to close Control Expert before starting this process. **Failure to follow these instructions may result in an unexpected behaviour.**

The Types Library Update utility will open.

Figure 3.9 – Type Library Update utility

Use the Browse button ("…") to navigate to the ***Family.dsc*** file in the PXM Type Library folder. Then click Ok.

The successfully registration will then be indicated by the following prompt:

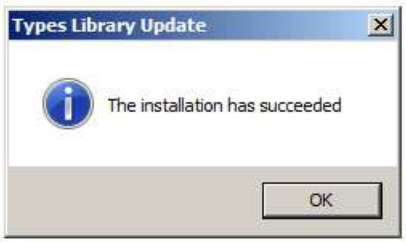


Figure 3.10 – PXM Type Library Installation Succesful

The imported library can be viewed in Control Expert by selecting the ***Types Library Manager*** item under the ***Tools*** menu. In the Type Library Manager, select the PMEPXM folder which can be found in the following folder:
<LibsetV14.0>\Profibus

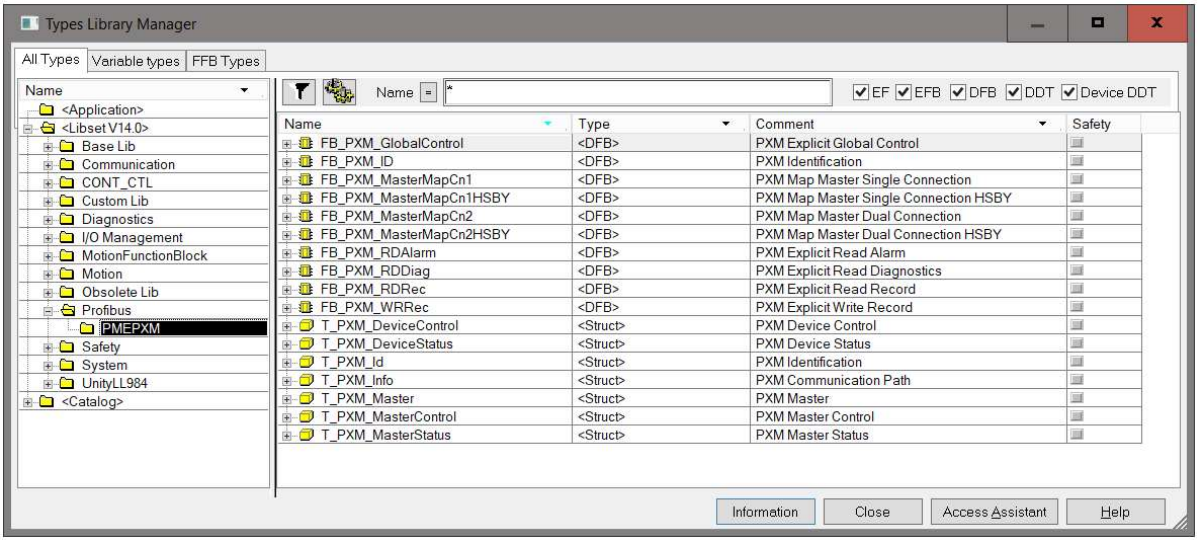


Figure 3.11 – Type Library Manager – PXM

3.5.3. UPDATING PROJECT

If a newer revision of the PXM Type Library is installed, then the existing project will need to be updated. To update the project, open the ***Types Library Manager*** item under the ***Tools*** menu. Right-click on the PMEPXM folder and select the ***Compare Project with Library*** option.

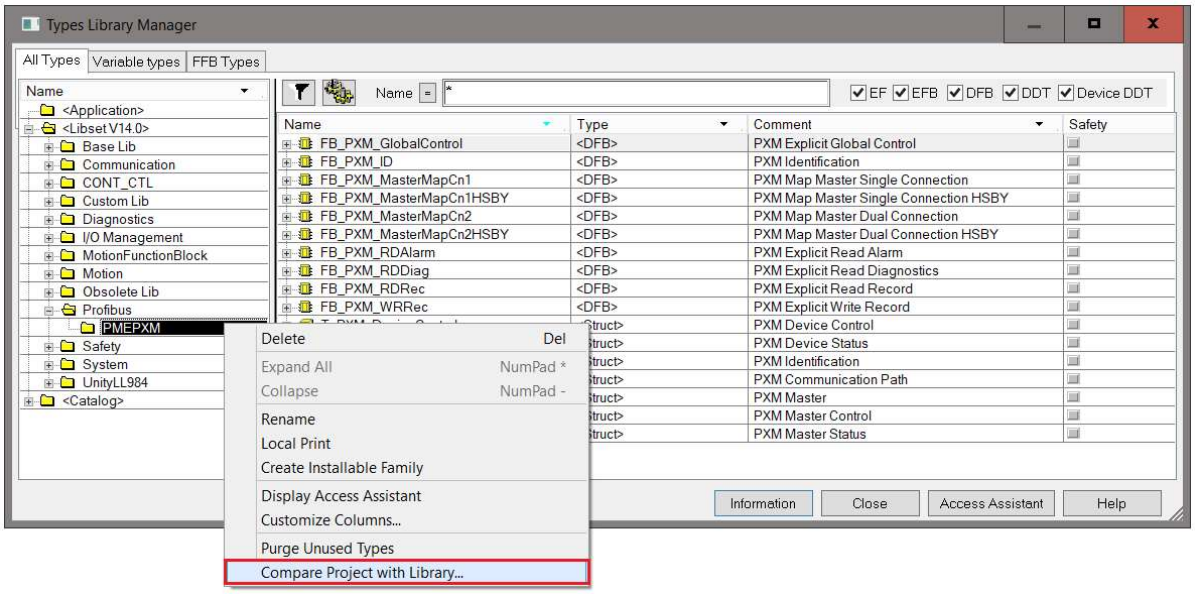


Figure 3.12 – Type Library Manager – Compare with project

The Library Version Management window will open showing any differences between the project and the updated PXM Type Library. To update, select the Update All button.

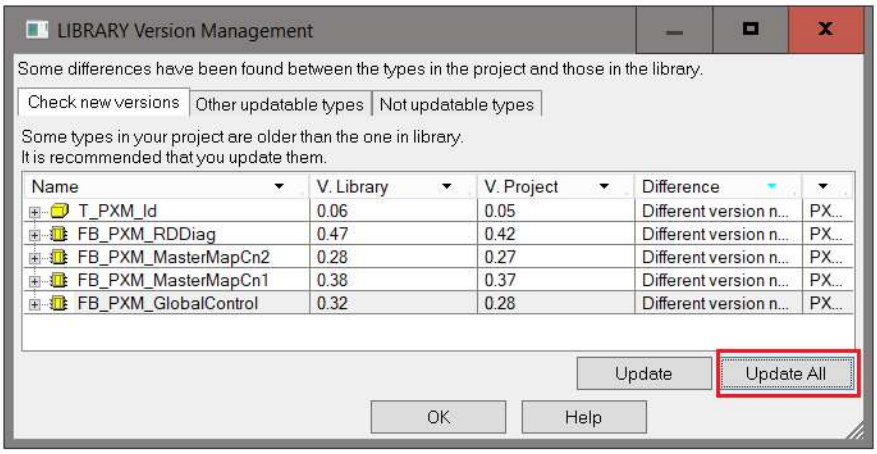


Figure 3.13 – Type Library Manager – Update Project

3.5.4. Mandatory Settings in Control Expert

Any project making use of the PXM Type Library must have the ***Allow Dynamic Arrays*** option enabled.

This can be enabled by selecting the ***Project Settings*** option under the Control Expert ***Tools*** menu. In the Project Settings window, select the ***Variables*** left menu item and then check the ***Allow Dynamic Arrays (ANY_ARRAY_XXX)*** option.

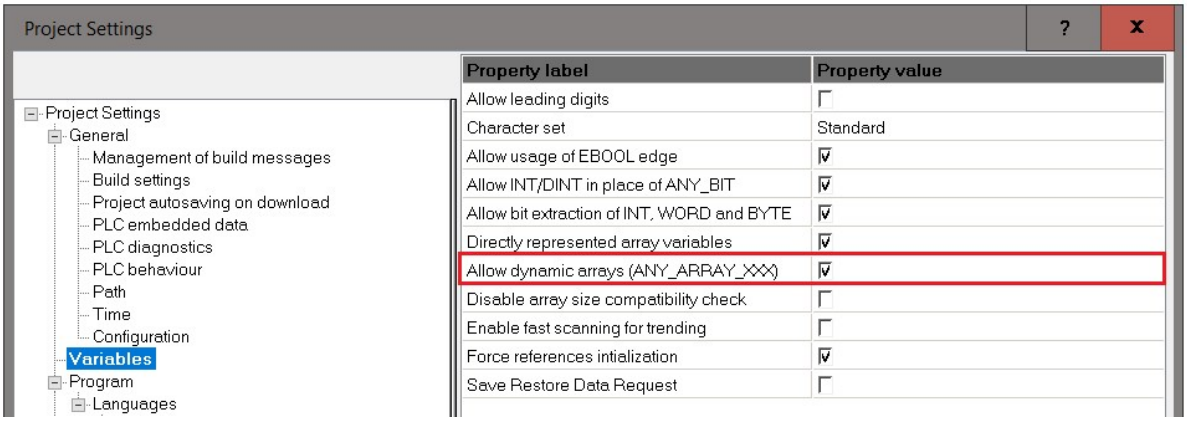


Figure 3.14 – Project Settings

3.5.5. Library Content

The PXM Type Library contains the following:

Type	DFB	Description
DFB	FB_PXM_MasterMapCn1	Master Mapping for Standalone Single Connection
	FB_PXM_MasterMapCn2	Master Mapping for Standalone Dual Connection
	FB_PXM_MasterMapCn1HSBY	Master Mapping for HSBY Single Connection

	FB_PXM_MasterMapCn2HSBY	Master Mapping for HSBY Dual Connection
	FB_PXM_ID	Identification for Explicit messaging
	FB_PXM_RDDiag	Explicit Read Diagnostics
	FB_PXM_RDRec	Explicit Read Record
	FB_PXM_WRRec	Explicit Write Record
	FB_PXM_RDAlarm	Explicit Read Alarm
	FB_PXM_GlobalControl	Explicit Global Control
DDT	T_PXM_Master	Master mapping structure (Status, Control and Input/Output Data)
	T_PXM_MasterControl	Master Control
	T_PXM_MasterStatus	Master Status
	T_PXM_DeviceControl	PROFIBUS Slave Device Control
	T_PXM_DeviceStatus	PROFIBUS Slave Device Status
	T_PXM_Id	Explicit Identification
	T_PXM_Info	Explicit connection path

Table 3.1 – PXM Type Library Content

For more details on the Mapping DFBs and DDTs see section 4.3.3.
For more details on the Explicit Messaging DFBs and DDTs see section 4.5.

3.6. CREATING A NEW PROJECT

Before the user can configure the module, a new ProSoft Configurator for Modicon project must be created. Under the File menu, select New.

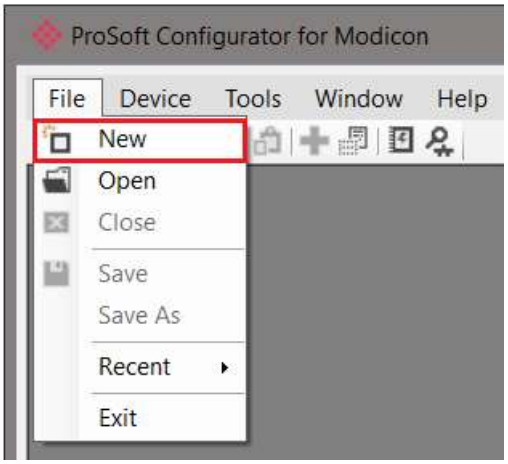


Figure 3.15 - Creating a new project

A ProSoft Configurator for Modicon project will be created, showing the Project Explorer tree view. To save the project use the Save option under the File menu.

A new device can now be added by selecting Add under the Device menu.

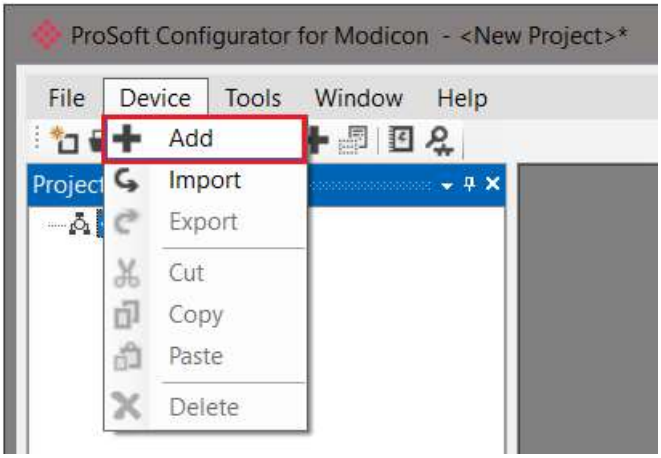


Figure 3.16 - Adding a new device

In the Add New Device window select the PXM PROFIBUS Master and click the Ok button.

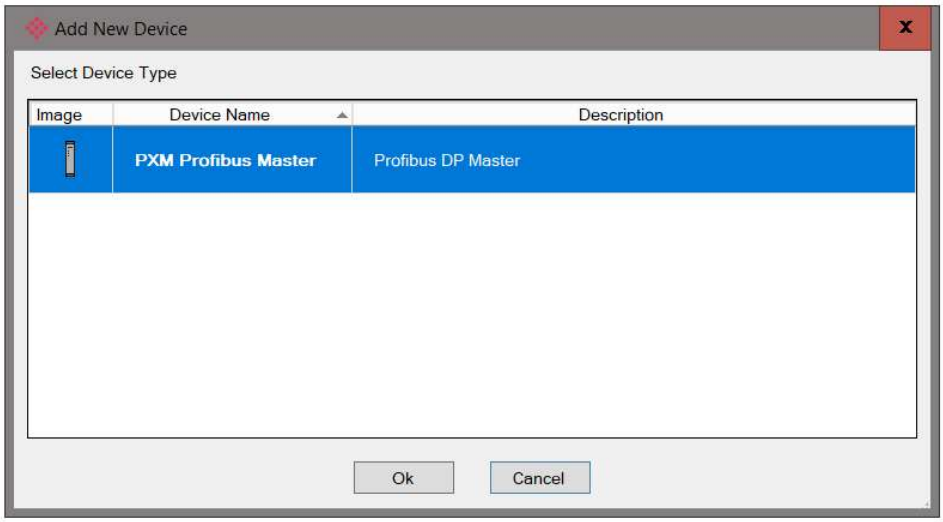


Figure 3.17 – PXM PROFIBUS Master

The PXM will appear in the Project Explorer tree as shown below, and its configuration window opened.

The PXM configuration window can be reopened by either double clicking the module in the Project Explorer tree or right-clicking the module and selecting *Configuration*.

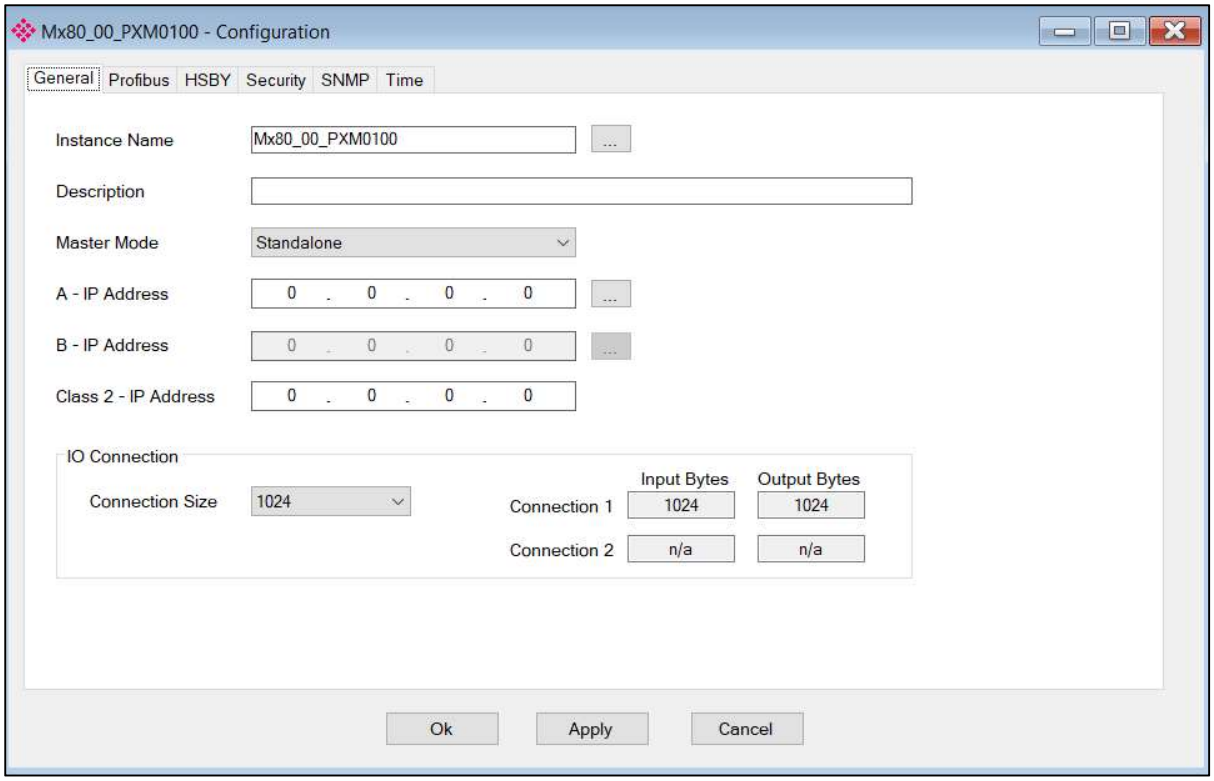


Figure 3.18 – PXM configuration

Refer to the additional information section in this document for the ProSoft Configurator for Modicon Utility's installation and operation documentation.

3.7. PXM PARAMETERS

The PXM parameters will be configured by the ProSoft Configurator for Modicon . **Refer** to the additional information section for documentation and installation links for ProSoft's Configurator for Modicon.

3.7.1. GENERAL

The General configuration is shown in the figure below. The PXM General configuration window is opened by either double clicking on the module in the tree or right-clicking the module and selecting *Configuration*.

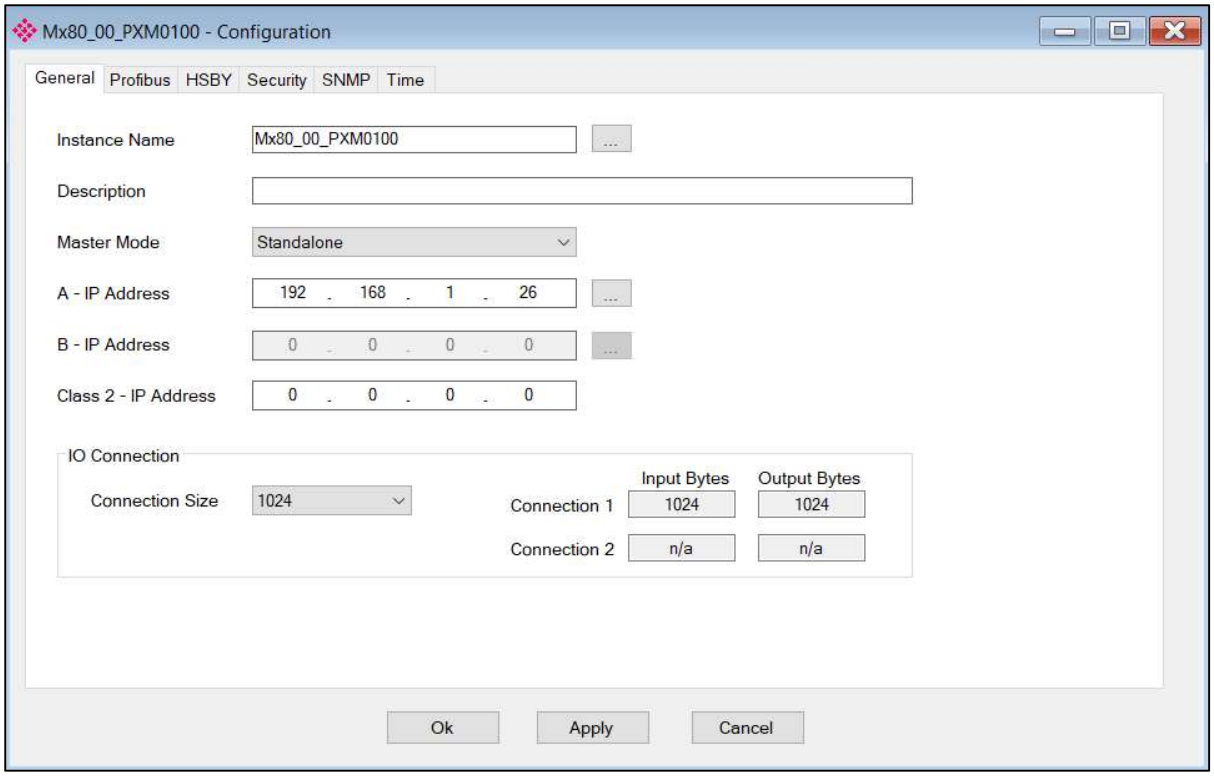


Figure 3.19 – PXM General configuration

The General configuration consists of the following parameters:

Parameter	Description
Instance Name	This instance name is used to identify the specific PXM module, and must conform to the specific naming convention.
Description	This parameter is used to provide a more detail description of the application for the module.
Master Mode	The PXM can operate in one of three modes: **Quiet** This mode allows the user to connect the PXM to an active bus and run a DP packet capture. In this mode the PXM will not communicate on the DP Bus but rather only listen. **Standalone** In this mode the PXM is the DP Master on the bus and connected to a non-HSBY M580 controller. This mode will **not** support any form of redundancy. **HSBY**

	In this mode the PXM will operate in conjunction with the HSBY M580 controllers providing DP Master redundancy. When the HSBY M580 swaps from the active to standby controller the PXM will provide similar functionality and the Standby PXM module in HSBY will take over the DP network,
A - IP Address	The IP address of the target module. The user can use the target browse button to launch the target browser to the select the PXM on the network.
B - IP Address	When the PXM is operating in HSBY mode this is the IP address of the other partner PXM module.
Class 2 – IP Address	This a second IP address that is assigned to the module and can be used for DPV1 Class 2 messaging (e.g. DTM). When operating in HSBY mode only the active DP Master will have this IP address enabled. When a HSBY swap occurs, the new active DP Master will enable this IP address and the new standby DP Master will disable this IP address. The Class 2 IP address is only available when one of the PXM modules is in a Primary role. When both PXM are in a Standby role (e.g. disconnected from the PLC, or PLCs in STOP) then the Class 2 IP address will not be available.
IO Connection	The PXM can connect to the M580 controller using a range of IO Connection Sizes and Counts. Note that when the connection size is greater than 1024 bytes the PXM will consume two Class 1 EtherNet/IP connections. • 256 bytes • 512 bytes • 1024 bytes • 1536 bytes • 2048 bytes • 2560 bytes

Table 3.2 - General configuration parameters

NOTICE
THE PXM WILL NOT BE ABLE TO COMMUNICATE WITH THE M580 CONTROLLER The configured Instance Name will need to match the name given in Control Expert for the PXM DTM or the PXM will not communicate with the M580 controller. See the Instantiate PXM DTM section. **Failure to follow these instructions may result in an unexpected behaviour.**

The configured Master Mode must match the system configuration in Control Expert.

⚠ WARNING
UNINTENDED EQUIPMENT OPERATION • Do not apply a Standalone configuration in a HSBY system. • Do not apply a HSBY configuration in a Standalone system. Applying a Standalone configuration in an HSBY system, or vice versa, can have unexpected consequences. **Failure to follow these instructions can result in death, serious injury, or equipment damage.**

The PXM ***Instance Name*** parameter can either be entered manually or built using the Instance Name utility. To select the latter option, click on the Build button adjacent to the Instance Name.

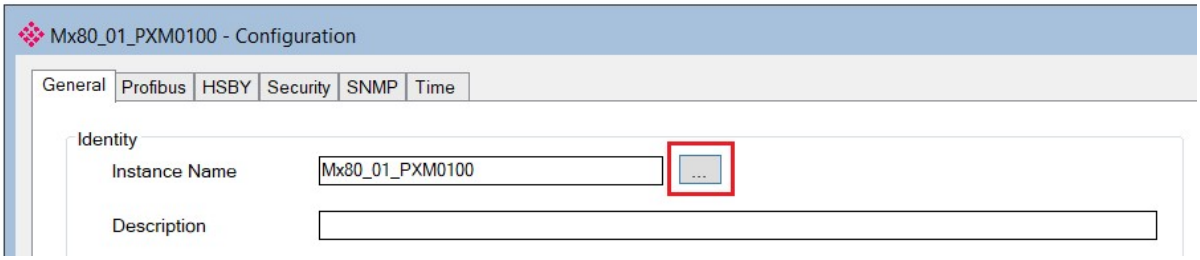


Figure 3.20 – Instance Name Build

The Instance Name builder can be used to build the correct Instance Name for the following arhitectures:

- Standalone – Local Rack
- Standalone – Remote Rack
- HSBY – Local Rack

An example of each is shown below.

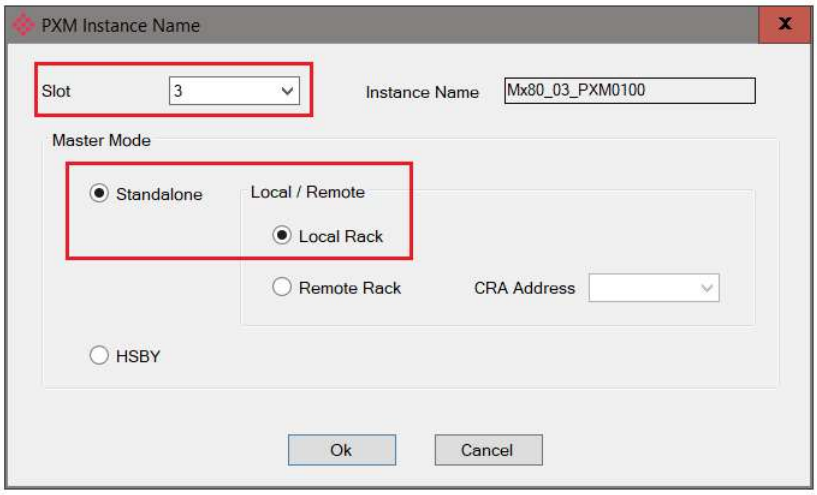


Figure 3.21 – Instance Name Builder – Standalone - Local Rack

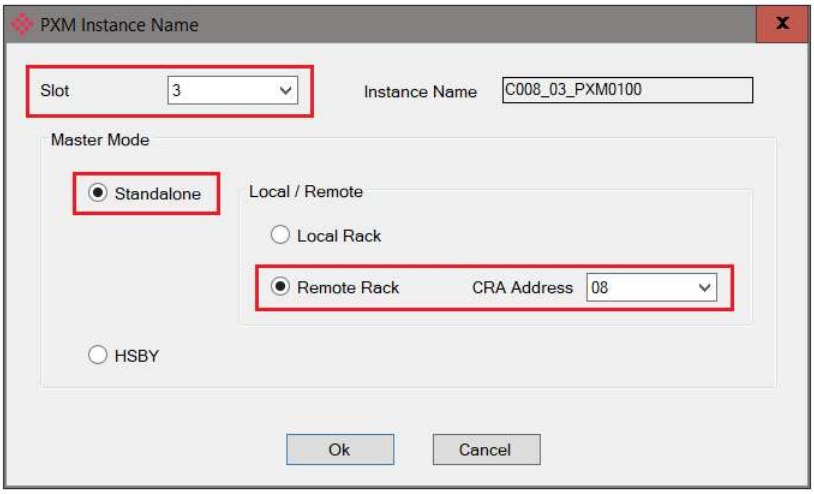


Figure 3.22 – Instance Name Builder – Standalone - Remote Rack

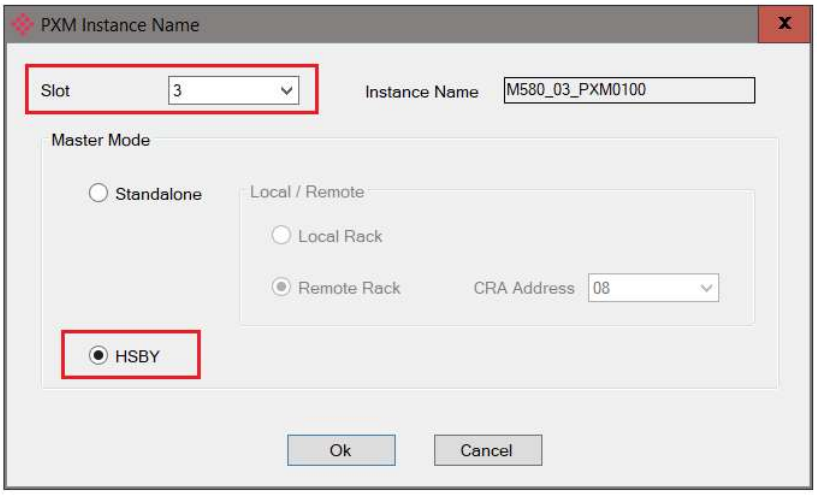


Figure 3.23 – Instance Name Builder – HSBY (Local Rack)

3.7.2. PROFIBUS

The PROFIBUS configuration is shown in the figure below. The PXM PROFIBUS configuration window is opened by either double clicking on the module in the tree or right-clicking the module and selecting *Configuration*.

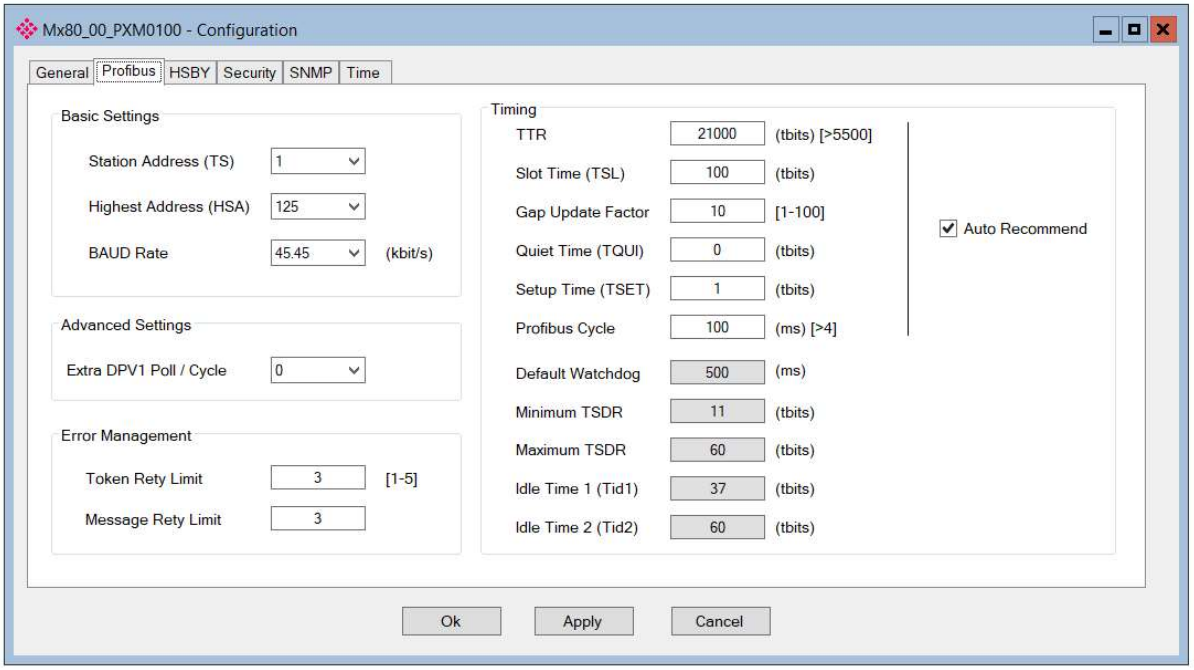


Figure 3.24 – PXM PROFIBUS configuration

The PROFIBUS configuration consists of the following parameters:

Parameter	Description
	Basic Settings
Station Address (TS)	**PROFIBUS Station Address** for the PXM module. TS should be different than any other slaves address on the PROFIBUS network, it should also be less-than or equal to the HSA below: Min: 0 Max: 126 Default: 1
Highest Address (HSA)	**Highest Station Address**. This is the highest station address of the active stations (masters). Passive stations (slaves) can have a higher address than the HSA. A low HSA is better for PROFIBUS performance. Min: 1 Max: 126 Default: 126
BAUD Rate	**Baud Rate** (in Kbps) of the PROFIBUS network: 9.6, 19.2, 45.45, 93.75, 187.5, 500, 1500, 3000, 6000 or 12000 Kbps. The baud rate selected

	should be supported by all slaves in the configuration. The baud rate should be selected depending on the cable length, see chapter "***PROFIBUS DP***"
	Advanced Settings
Extra DPV1 Poll / Cycle	The number of additional DPV1 Polls (Class 2) per PROFIBUS Cycle. This parameter should be equal or greater to the maximum number of simultaneous explicit DPV1 messages that may be sent. Increasing this parameter results in faster Asset Management DTM updates.
	Error Management
Token Retry Limit	**Token Retry Limit** is the number of times that a PROFIBUS Master tries to pass the token before deciding that a station is not there. Value must be in the following range: Min: 0 Max: 15 Default: 3
Message Retry Limit	**Message Retry Limit** is the number of telegram repetitions if the address doesn't react. Value must be in the following range: Min: 0 Max: 15 Default: 1
	Timing
TTR	**Target Rotation Time** indicates the maximum time available for a token circulation (time for PROFIBUS token to be passed to another master and be back). It takes in account the number of slaves with their IO size (data exchanges telegram), different telegrams needed and their duration times (FDL status, global control, pass token), all mandatory timing with respect to the PROFIBUS standard (time slot, min and max Tsdr, Tqui, Tset, …) and a safety margin which allows bandwidth for acyclic messages (DPV1, …). Min: 0 Max: 16777215
Slot Time (TSL)	**Slot Time** (in tbits) is the maximum time the PXM will wait, after the transmission of a request, for the reception of the first byte (Tchar) of an answer. (It allows detecting a timeout.) It can be increased when repeaters are used in the PROFIBUS network topology. The value must respect the rule: Min: 37 Max: 16383
Gap Update Factor	**Gap Update Factor:** The range of addresses between 2 consecutive active stations is called GAP. This GAP is submitted to a cyclic check during which the system identifies the station condition (not ready, ready or passive). Min: 1 Max: 100
Quiet Time (TQUI)	**Quiet time** (in tbits) is the time that a station may need to switch from sending to receiving. It must respect the rule: TQUI < MIN_TSDR Min: 0 Max: 255

Setup Time (TSET)	**Setup Time** (in tbits) is the reaction time on an event. Calculation of TSET must respect the rule: Min: 1 Max: 494
Default Watchdog	**Default Devices Watchdog** (in ms) value defines the watchdog value assigned by default to all devices in the configuration.
PROFIBUS Cycle	**PROFIBUS Cycle** (in ms) (read/Write) field defines the cyclic time the master will respect between two IO Data Exchange sequences. This parameter can be increased by the user when the PROFIBUS network load does not allow the processing of acyclic requests.
Min TSDR	**Smallest Station** (in tbits) is the minimum time that a PROFIBUS DP slave must wait before it may answer. It must respect the rule: TQUI < MIN_TSDR Min: 11 Max: 1023
Max TSDR	**Largest Station** (in tbits) is the maximum time that a PROFIBUS DP slave may take in order to answer. Calculation of MAX_TSDR must respect the rule: Min: 37 Max: 65525
Idle Time 1 (Tid1)	**Time Idle1** (in tbits) is the time between the acknowledgement frame or token frame reception and the transmission of the next frame. Tid1 = Max(Tsyn+Tsm, MIN_TSDR) with Tsyn= 33 Tsm= 2 + 2* TSET + TQUI
Idle Time 2 (Tid2)	**Time Idel2** (in tbits) is the time between the transmission of an unconfirmed packet and the transmission of the next packet. Tid2 = Max (Tsyn+Tsm, MAX_TSDR) with Tsyn= 33 Tsm= 2 + 2* TSET + TQUI

Table 3.3 - PROFIBUS configuration parameters

When the user changes the BAUD rate all the PROFIBUS timing parameters and HSBY parameters will change to the default values for that specific BAUD Rate.

NOTICE
UNEXPECTED BEHAVIOUR Ensure that all timing parameters are correct after making any BAUD rate changes. **Failure to follow these instructions may result in an unexpected behaviour.**

3.7.3. HSBY

The HSBY configuration is shown in the figure below. The HSBY mechanism is described in chapter 5.
The PXM HSBY configuration window is opened by either double clicking on the module in the tree or right-clicking the module and selecting *Configuration*.

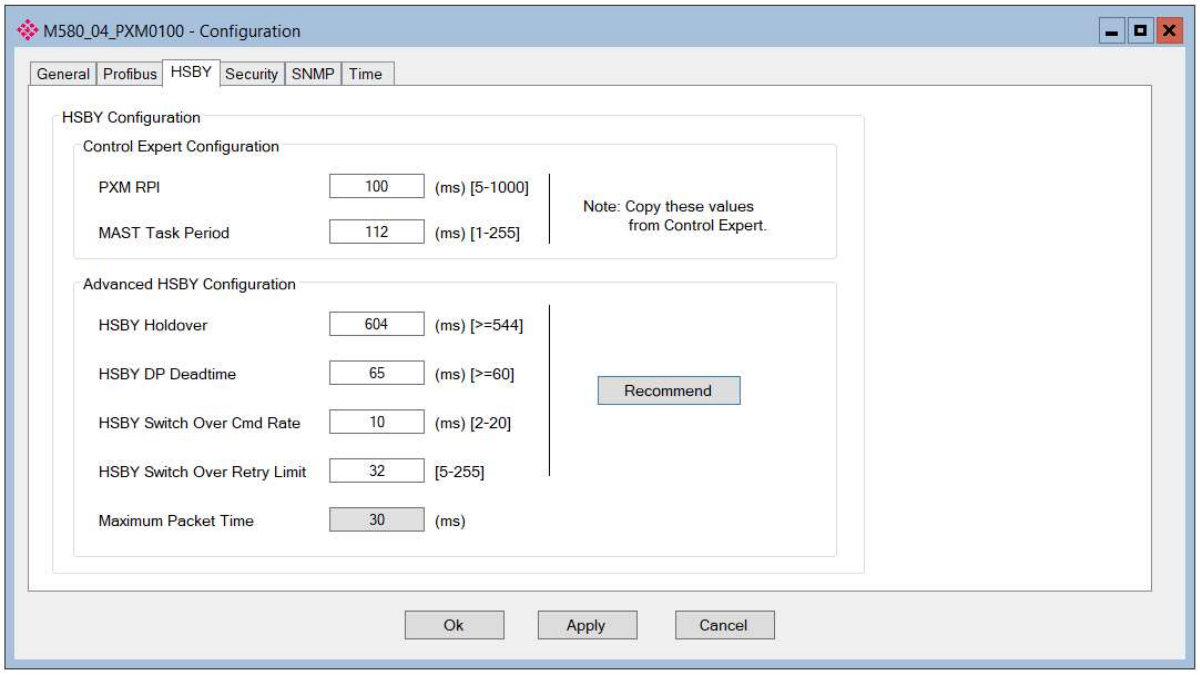


Figure 3.25 – PXM HSBY configuration

The HSBY configuration consists of the following parameters:

Parameter	Description
PXM RPI (ms)	The PXM's RPI (Requested Packet Interval) as configured in the Control Expert project. Valid values are 5-1000 milliseconds. This parameter is not used directly in the PXM's configuration, but is used to calculate HSBY parameters when the ***Recommend*** button is used.
MAST Task Period (ms)	The period of the MAST Task as configured in the Control Expert project. Valid values are 1-255 milliseconds. This parameter is not used directly in the PXM's configuration, but is used to calculate HSBY parameters when the ***Recommend*** button is used.
HSBY Holdover (ms)	This is the amount of time the active PXM will keep running the PROFIBUS DP network without an EtherNet/IP Class 1 connection from a running Primary M580 PLC located in the local rack.

	The above state can occur when the PXM switches from the active M580 controller to the standby M580 controller. During this switch over there is a period where the PXM will operate the DP network without a connection to a Primary M580 controller where the last received data (from a M580 controller) is being used for the Holdover time before the PXM sets the DP network to OFFLINE. See chapter 5 for configuration guidelines.
HSBY DP Deadtime (ms)	The DP Deadtime is the amount of time the standby PXM will wait when the DP network is quiet before taking over as the DP Master.
HSBY Switch Over Cmd Rate (ms)	This is the rate at which the Standby PXM sends a switch over request to the active PXM. This value will depend on the BAUD rate selected, but generally the faster (i.e. lower) this parameter is set the faster the switch over will be.
HSBY Switch Over Retry Limit	This is the retry limit before the standby PXM takes over the DP network if it has not received confirmation from the active PXM to take over the DP network.

Table 3.4 - HSBY configuration parameters

⚠ *WARNING*

INCORRECT BEHAVIOR OF HSBY SYSTEM : SWAP MAY FAIL AND PXM MAY INTERMITTENTLY RUN

In an HSBY system the HSBY parameters must be configured correctly and must match the configuration inside Control Expert.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

3.7.4. SECURITY

The Security configuration is shown in the figure below. The PXM Security configuration window is opened by either double clicking on the module in the tree or right-clicking the module and selecting *Configuration*.

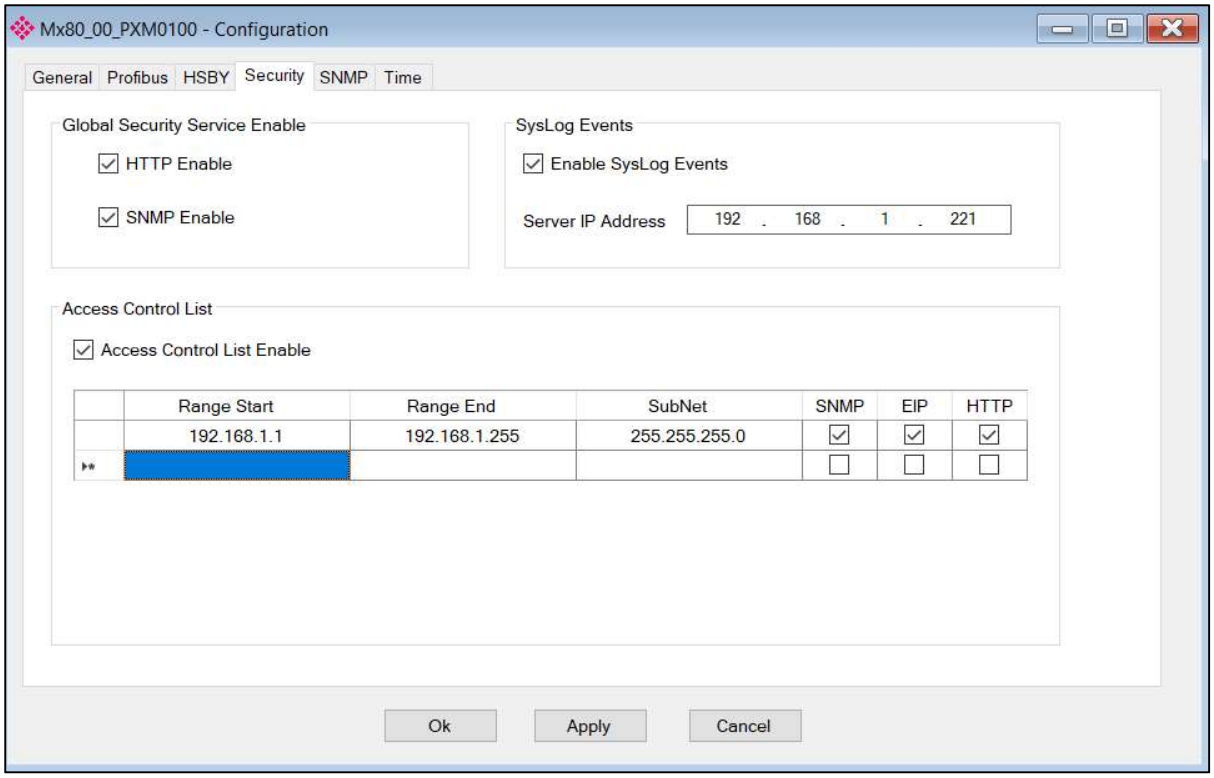


Figure 3.26 – PXM Security configuration

The Security configuration consists of the following parameters:

Parameter	Description
HTTP Enable	Enabling this will allow the PXM to respond to HTTP requests when a user wants to view the Webserver. If this has been disabled, the user will not be able to view the Webserver.
SNMP Enable	Enabling this will allow the PXM to respond to SNMP requests. If this has been disabled, a SNMP server will not be able to see the PXM.
Access Control List (ACL)	The ACL will allow the user to allow certain IP address ranges to only access certain protocols. This can be enabled by selecting the Access Control Enable option. **NOTE:** The Global Security Services above will override any ACL rule. **Range Start** This is the starting IP address of the range specific to the rules in the line item (e.g. allowing HTTP). **Range End** This is the end IP address of the range specific to the rules in the line item.

	Subnet The subnet mask that will be applied to the received IP address to check if it is in the range. **SNMP, EIP, HTTP** These are the protocols that will be allowed for a specific IP range.
Enable SysLog Events	The PXM can log up to 2048 events internally in NV memory. When enabling SysLog Events the PXM will unload these events to a SysLog Server.
SysLog Server IP Address	This is the IP address of the SysLog Server. The PXM will connect to the SysLog server using TCP Port 601.

Table 3.5 - Security configuration parameters

NOTICE

PXM WILL NOT OPERATE CORRECTLY

- Do not exclude the PLC's IP address when configuring the Access Control List. If the PLC is excluded, it will not establish a connection with the PXM.
- Do not exclude the PLC's "IP Address A" when configuring the Access Control List. This address is used for the PXM's explicit messaging.
- Adjust the SysLog server settings to allow the connection of PXM to the SysLog server with TCP Port 601.

Failure to follow these instructions may result in an unexpected behaviour.

⚠ WARNING

INCORRECT BEHAVIOUR OF A HSBY SYSTEM: SWAP MAY FAIL

In an HSBY system, do not exclude the two partner PXM's IP addresses when configuring the Access Control List.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

In case the PXM module is no longer contactable due to an invalid ACL configuration being downloaded to it, the following steps should be followed:

1. Remove the module and set DIP Switch 1 to the ON position
2. Reinsert the module.
3. Download corrected configuration to the module (PCM)
4. Using PCM, Upload the corrected configuration to the FDR server.
5. Remove the module and set DIP Switch 1 to the OFF position
6. Reinsert the module.

3.7.5. SNMP

The SNMP configuration is shown in the figure below. The PXM SNMP configuration window is opened by either double clicking on the module in the tree or right-clicking the module and selecting *Configuration*.

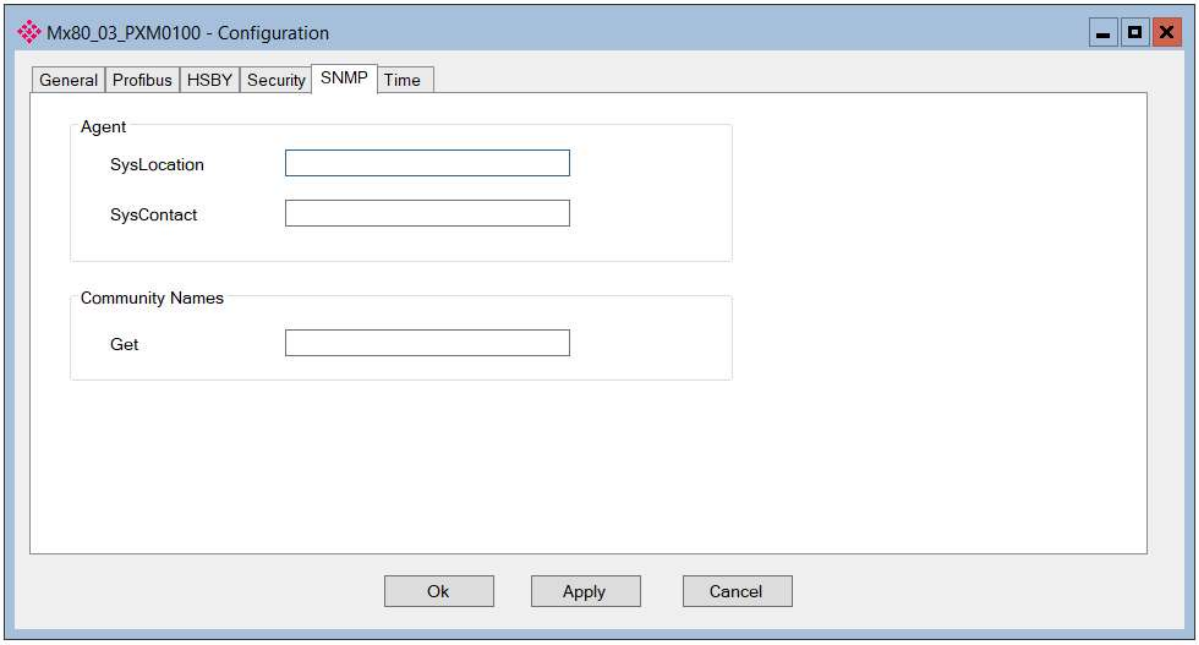


Figure 3.27 – PXM SNMP configuration

The SNMP configuration consists of the following parameters:

Parameter	Description
Agent SysLocation	Physical location of the module.
Agent SysContact	Contact name of the person responsible for maintaining the module.
Community Name - Get	Community name for the read commands

Table 3.6 - SNMP Configuration parameters

3.7.6. TIME

The PXM can synchronize its local clock with an NTP (Network Time Protocol) server. This allows the SysLog events to be logged with an accurate timestamp. The Time configuration is shown in the figure below. The PXM Time configuration window is opened by either double clicking on the module in the tree or right-clicking the module and selecting *Configuration*.

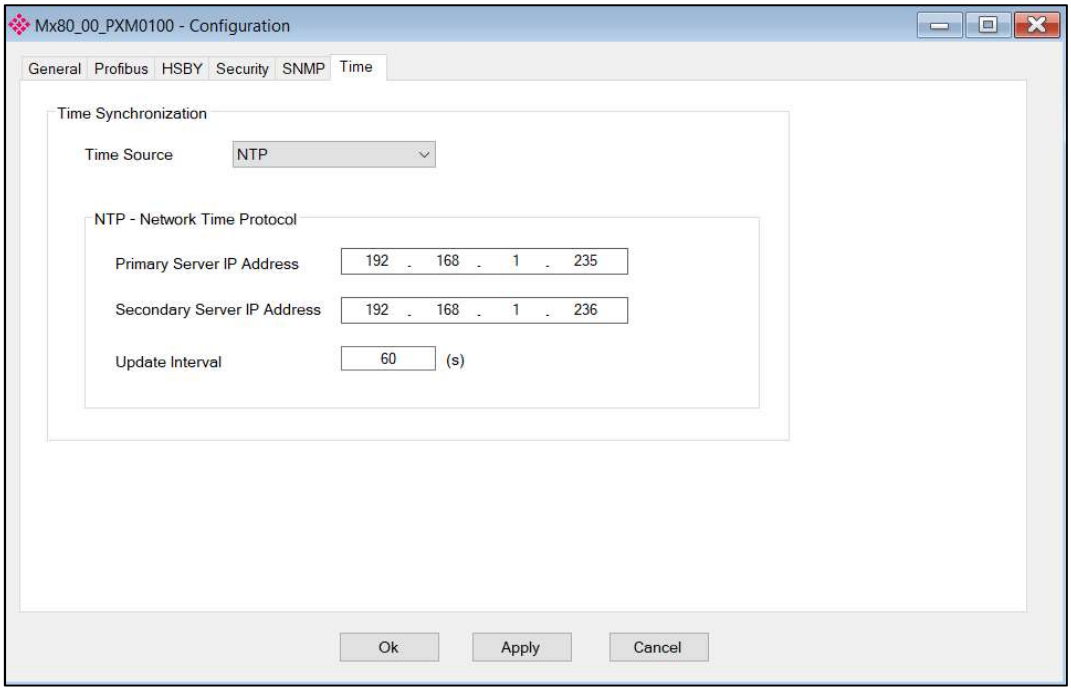


Figure 3.28 – PXM Time configuration

The Time configuration consists of the following parameters:

Parameter	Description
Time Source	The can select one to two options for the time source: **None** No Time synchronization will occur when this mode is set. **NTP** The PXM will attempt to synchronize to an NTP time server.
Primary Server IP Address	The IP address of the primary NTP server. If the primary is not available and there is a secondary NTP server at the Secondary IP address, then the PXM will attempt to synchronize to the Secondary NTP Server.
Secondary Server IP Address	The IP address of the secondary NTP server.
Update Interval	This is how often the PXM will synchronize its internal clock with the NTP time server.

Table 3.7 - Time configuration parameters

3.8. VERIFY CONFIGURATION

The PXM's configuration can be verified at any time by right-clicking on the PXM in the project tree and selecting the ***Verify Configuration*** option.

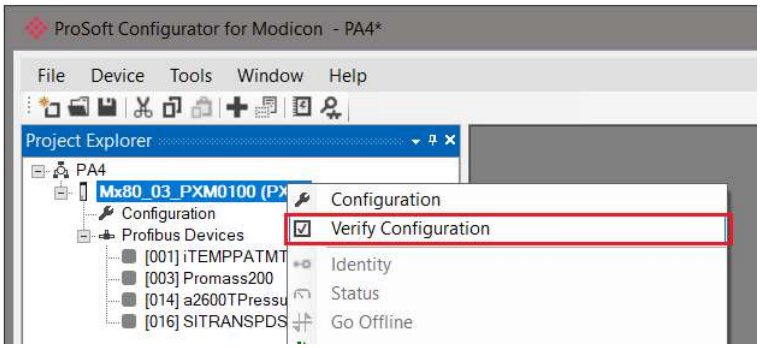


Figure 3.29 - Selecting Verify Configuration

Once the project configuration has been checked, the verification results are displayed. Each verification item is categorized as one of the following:

- ***Info*** – Information Only
- ***Warning*** – User to take note.
- ***Error*** – Invalid configuration that will be prevent configuration download.

The total count of errors and warnings are displayed at the bottom of the window.

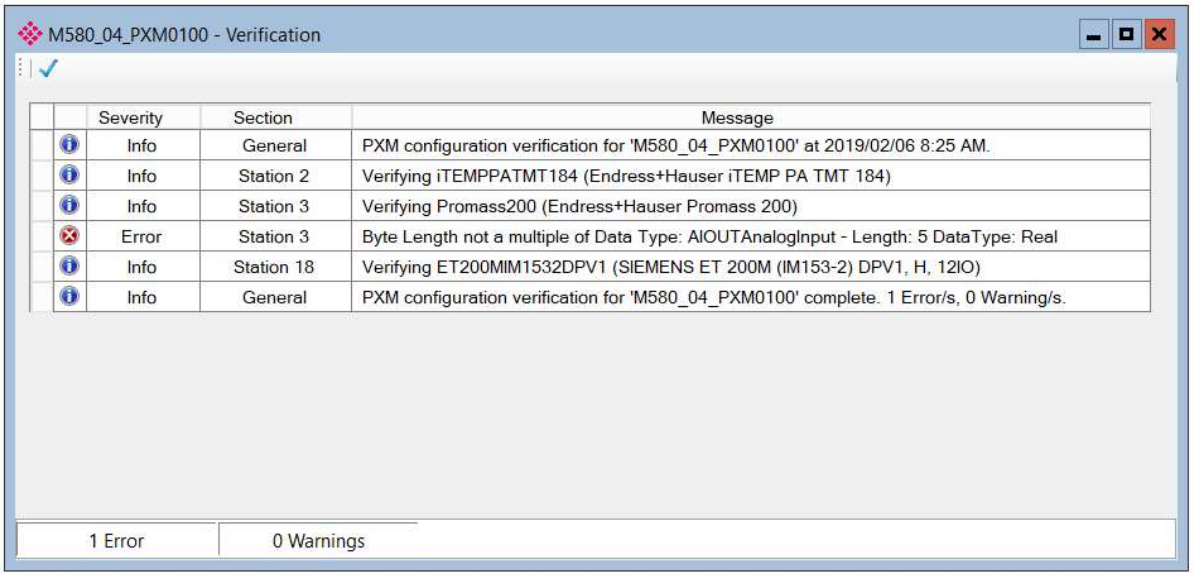


Figure 3.30 – Verification Results

Each time a module download is selected, the configuration will first be verified. Should any warnings or errors be generated then the verification result window will be displayed. Should any errors be generated, the download process will be aborted. All the possible Verification Notifications are listed in section 12.4.

3.9. MODULE DOWNLOAD

Once the PXM configuration has been completed, it must be downloaded to the module. The configured IP address of the module will be used to connect to the module.

To initiate the download, right-click on the module and select the Download option.

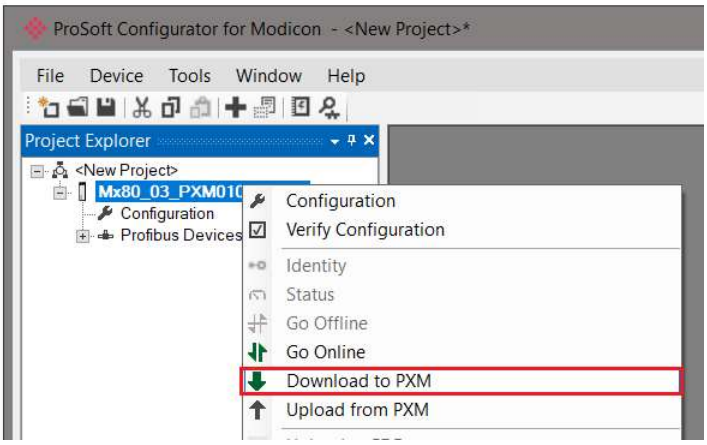


Figure 3.31 - Selecting Download

Once complete, the user will be notified that the download was successful.

Figure 3.32 - Successful download

Within the ProSoft Configurator for Modicon environment the module will be in the Online state, indicated by the green circle around the module. The module is now configured and will start operating immediately.

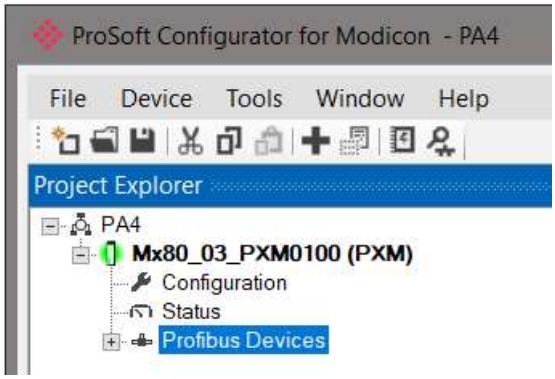


Figure 3.33 - Module online

3.10. DEVICE DISCOVERY (ONLINE)

Once the online with the PXM in the ProSoft Configurator for Modicon the user will be able to scan the PROFIBUS network for slave devices.

NOTICE
UNEXPECTED BEHAVIOUR Check that all PROFIBUS parameters have been configured correctly. Incorrect parameters (e.g. BAUD rate) will result in the PXM not communicating with slave devices on the PROFIBUS network. **Failure to follow these instructions may result in an unexpected behaviour.**

3.10.1. DISCOVERY

The slave device discovery can be found by selecting the *Discovered Nodes* tab in the PXM status window.

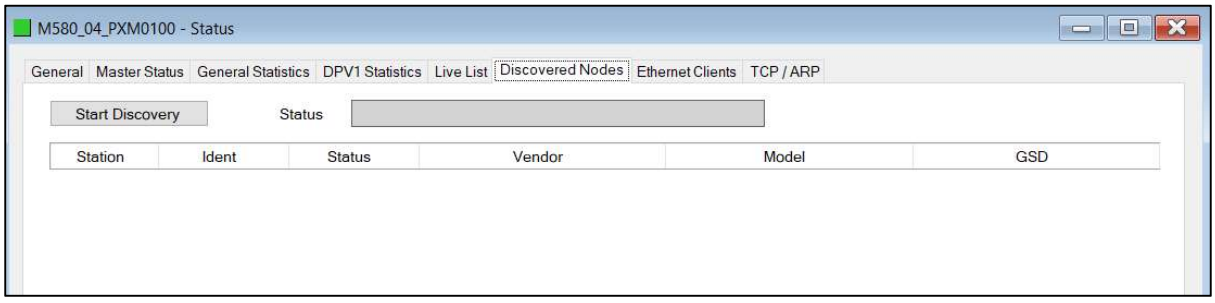


Figure 3.34 –Device Discovery

To start a new device discovery the *Start Discovery* button must be pressed. Once the discovery is done the slave devices found will be listed below. The time to scan the bus will

depend on the BAUD Rate selected. The higher the BAUD rate the faster the bus discovery scan time will be.

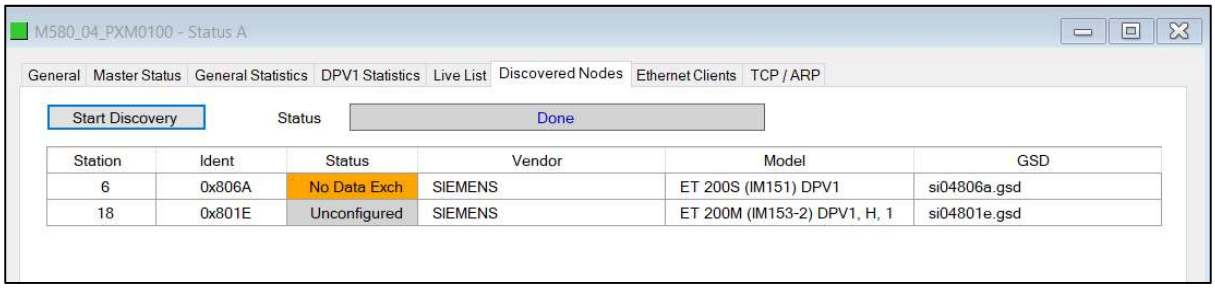


Figure 3.35 –Devices Found

The status of the discovered slave devices will be one of the following:

Status	Description
Data Exch	The device is configured and is exchanging data.
No Data Exch	The device is configured but not exchanging data.
Ident Mismatch	The device type configured at that station address is different from the device discovered.
Unconfigured	The device is not configured.

Table 3.8 – Discovered Device Status

If a device has been found that is not currently in the PXM configured device list the user will be able to add the device from this window by right-clicking on the device and selecting *Add Device*.

NOTICE
UNEXPECTED BEHAVIOUR The appropriate GSD file will need to be already registered before a device can be added to the PXM configuration. **Failure to follow these instructions may result in an unexpected behaviour.**

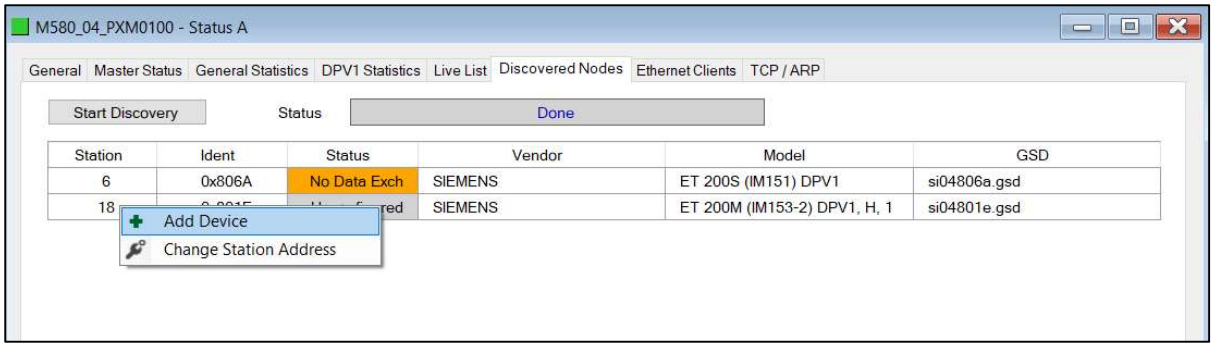


Figure 3.36 – Adding the Field Devices Found

The user will need to select the GSD file add the device to the PXM configured device list.

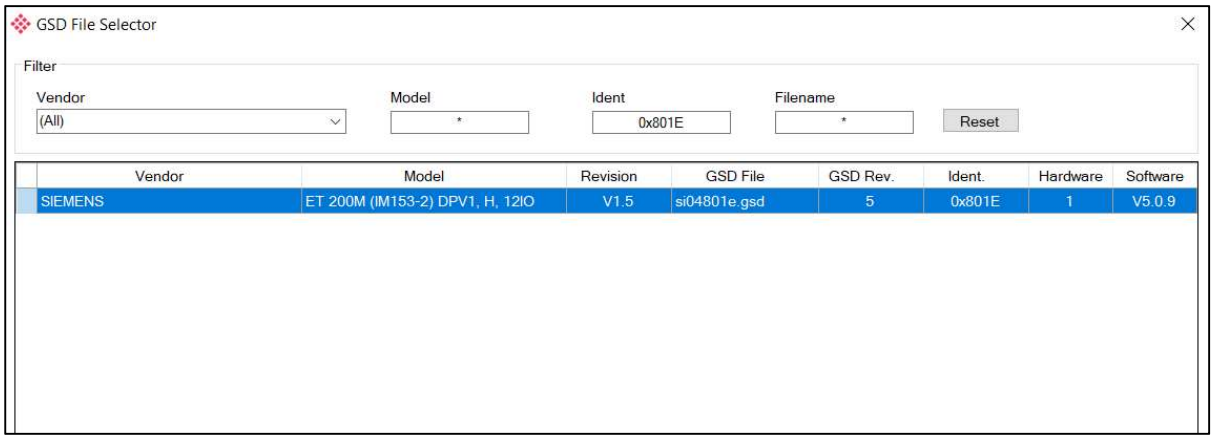


Figure 3.37 – Selecting the GSD for the slave device

Once the devices have been correctly set up (as well as the correct mapping is in Control Expert) the devices will show up as exchanging data.

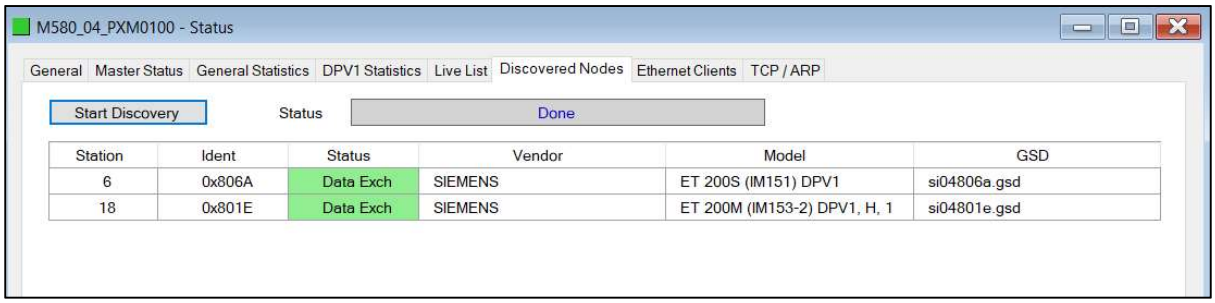


Figure 3.38 – Discovering running devices

The Discovered Nodes list will be cached and display the same information until the Start Discovery is again selected.

3.10.2. Device Station Address Change

Certain devices can be set up to allow remotely changing of the station address. Devices with this option set generally defaults to station address 126. The user can change the station address of a device (if the device is correctly setup) by right-clicking on the device in the Discovery Lost and selecting *Change Station Address*.

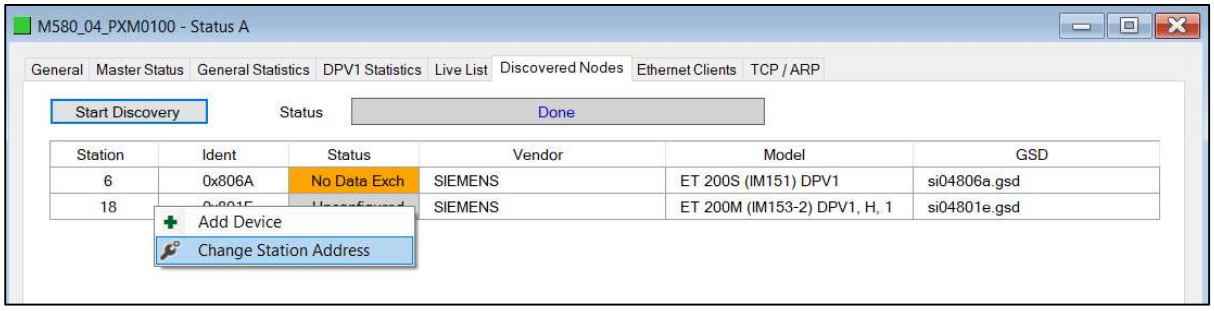


Figure 3.39 – Changing Station Address

Next the user will need to select the new station address for the device. Once selected press the Set button.

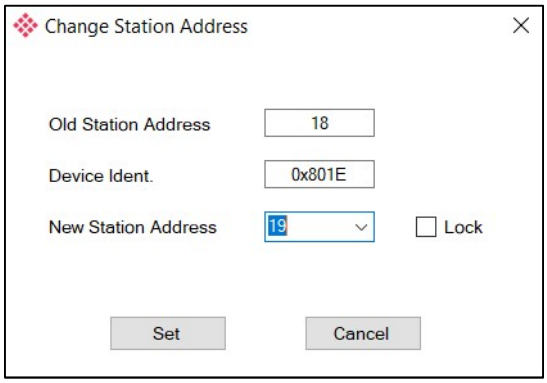


Figure 3.40 – Selecting new Station Address.

Once the request has been sent the user can either start a new network discovery to confirm the address has changed or monitor the LiveList (see the *Diagnostics* section).
The amount of time for the device to appear at the new station address is device dependent. In the LiveList there will be a period where both node addresses show up while the original station address is timing out.

Check that the new station address is correct.

⚠ WARNING
CAUTION COMMUNICATION FAILURE Do not set the station address to an address that is already present on the PROFIBUS network. **Failure to follow these instructions can result in death, serious injury, or equipment damage.**

The slave device will need to be in the correct state before it will accept a command to change its station address (e.g. Not be in data exchange state).

NOTICE
UNEXPECTED BEHAVIOUR Check that slave device is in the correct state before proceeding. **Failure to follow these instructions may result in an unexpected behaviour.**

3.11. ADDING PROFIBUS DP DEVICES

The user will need to add each PROFIBUS device to the PXM which can then be configured. This is done by right-clicking on the *PROFIBUS Devices* item in the tree and selecting *Add PROFIBUS Device*.

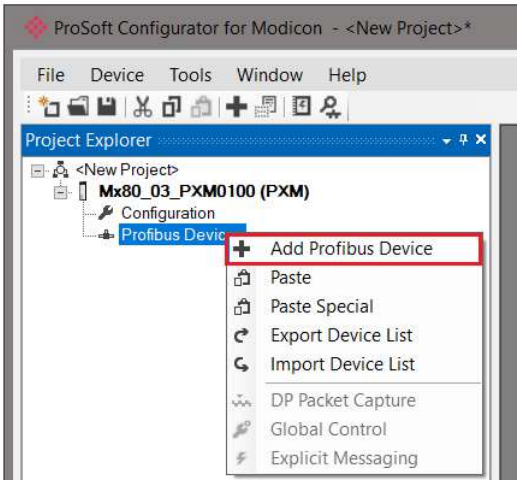


Figure 3.41 – Adding a PROFIBUS Field Device

Next the user will need to select the device to be added to the PXM. This is done by selecting the device from the GSD File Selector and pressing *Ok*.

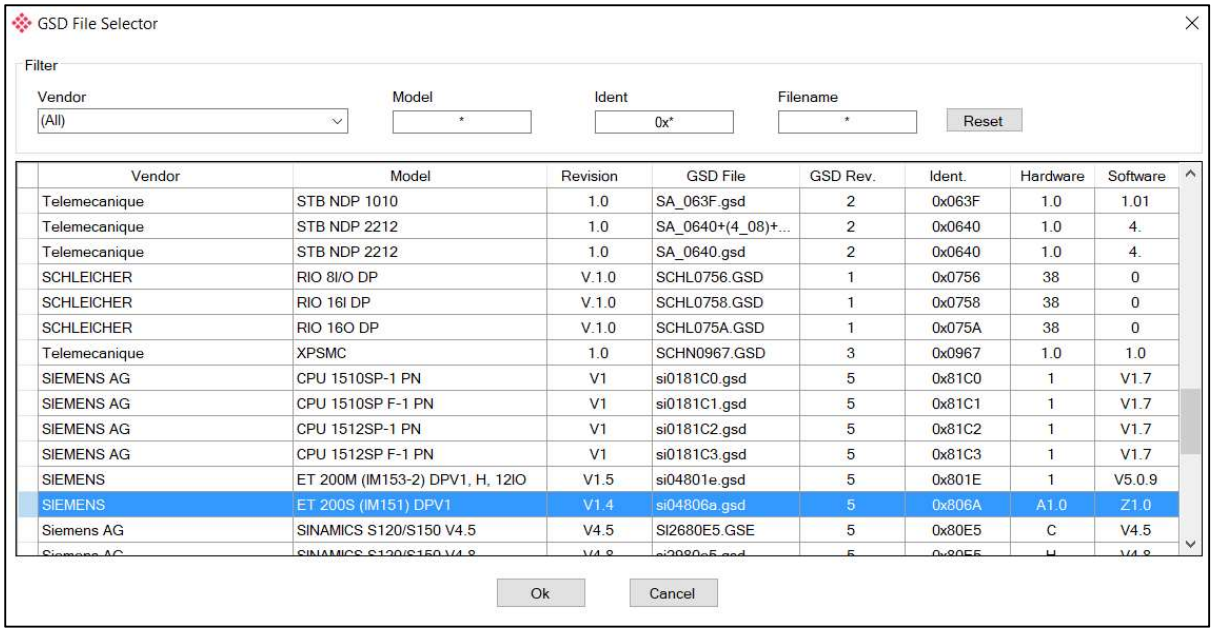


Figure 3.42 – Selecting a PROFIBUS Field Device

Once the device has been added the General Configuration page will be opened and the device will be added at the first open PROFIBUS Station Address.

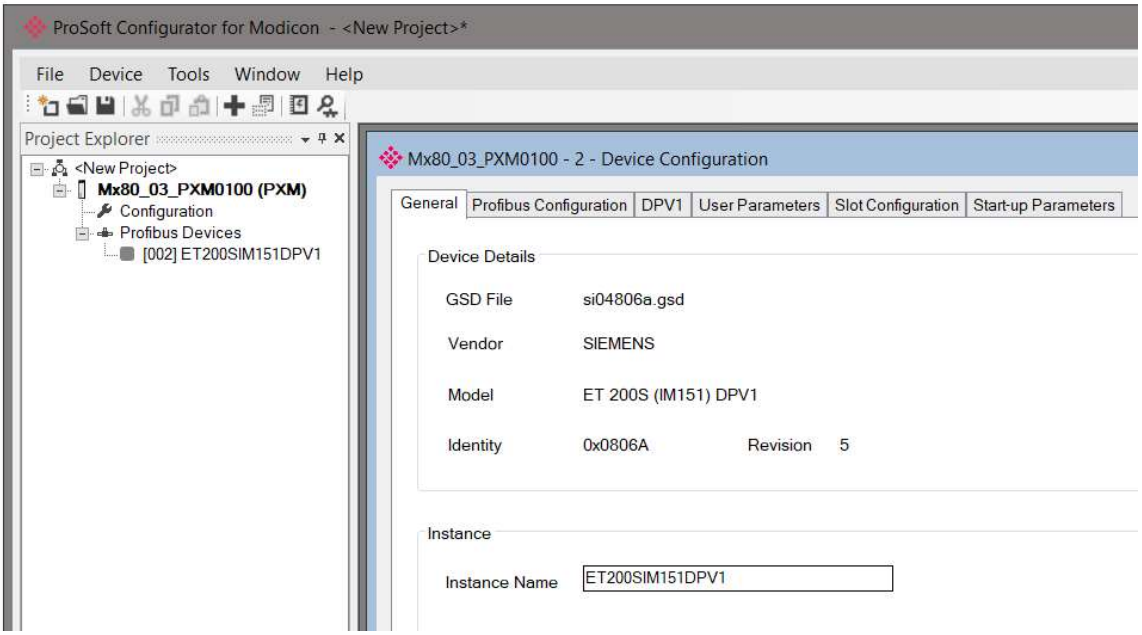


Figure 3.43 – PROFIBUS Field Device Added

3.11.1. GENERAL

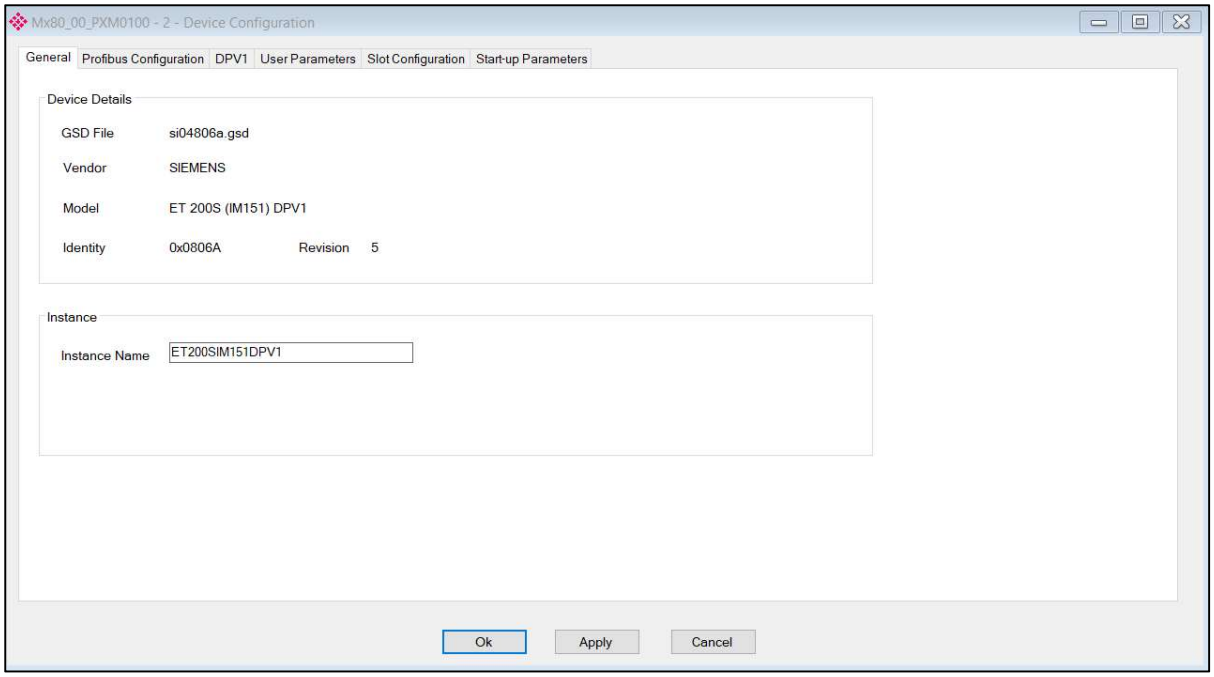


Figure 3.44 – Field Device General configuration parameters

The General configuration consists of the following parameters:

Parameter	Description
Instance Name	The device instance name which will be used to create the Tag names and DDTs in Control Expert.

Table 3.9 –Device General configuration parameters

The General configuration is shown in the figure below. The Device General configuration window is opened by either double clicking on the slave device in the tree or right-clicking the slave device and selecting *Configuration*.

3.11.2. PROFIBUS CONFIGURATION

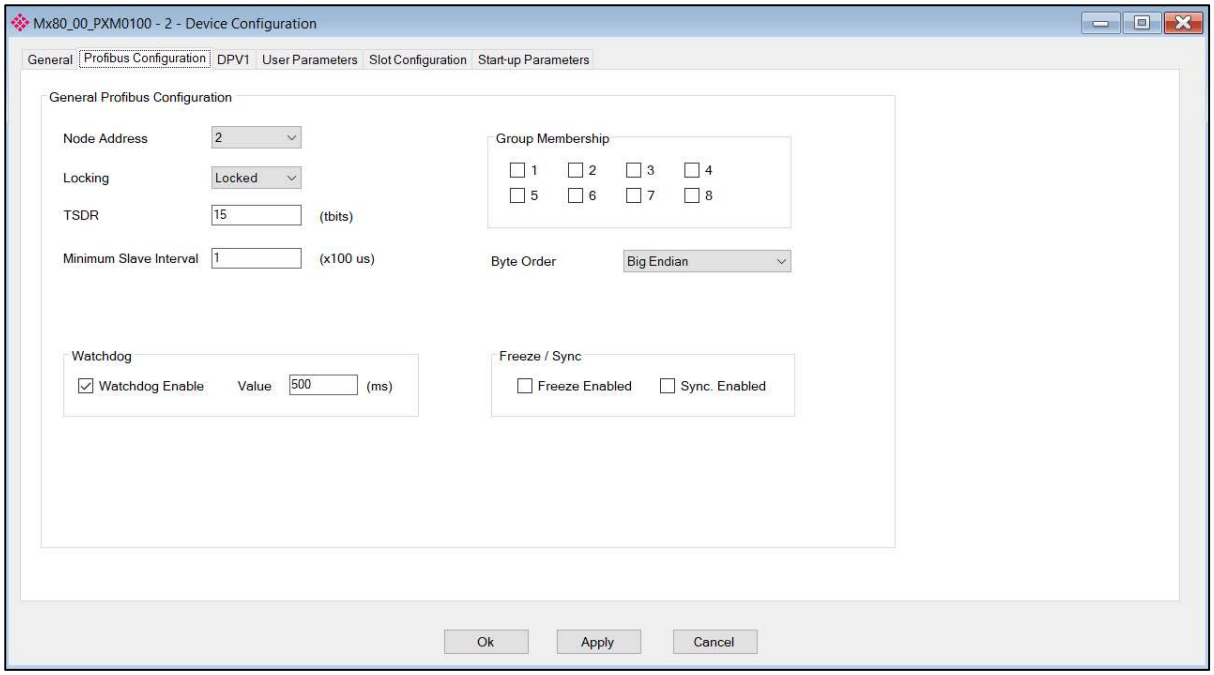


Figure 3.45 – Field Device PROFIBUS configuration parameters

The PROFIBUS configuration consists of the following parameters:

Parameter	Description
Node Address	This is the station address configured for the added device. This is the address the PXM will use to look for and configure the device for Data Exchange.
Locking	This parameter will configure the device to lock or unlock certain parameters for other DP Masters. **Locked** DP slave is locked for other masters. **Unlocked** DP slave is not locked for other masters. **Overwrite** Min TSDR and user parameters can be overwritten by another master.
TSDR	This parameter is the minimum time (in tbits) that a PROFIBUS-DP slave must wait before it responds. It must respect the rule: Min: 11 Max: 1023 Default: 11
Minimum Slave Interval	This is the minimal time (x 100 us) that the PROFIBUS must wait between two IO data exchanges with this device. The default value proposed comes from the GSD File. Min: 1

	Max: 65535
Watchdog Enable	Enables the watchdog for the slave device data exchange. The slave device monitors the data exchange rate (PROFIBUS Cycle) and it must be less than the Watchdog Value else the slave device will change back into a unconfigured state.
Watchdog Value	Is used to monitor cyclic communication and must be significantly higher than the time required for one PROFIBUS cycle. If a slave does not receive a request frame for a period of time longer than the watchdog time, it will revert to its initial, power-up state and cyclic communication will have to be reestablished. The minimum and default values are defined by the PXM Default Watchdog setting in the PXM PROFIBUS configuration.
Group Membership	Specifies which groups the slave belongs to. A slave can be in multiple groups at a time (from 1 through 8). Groups are used by the master when it sends a Sync or Freeze command.
Byte Order	Specifies the byte order when mapping the PROFIBUS device data on the Control Expert DFB (Derived Function Block) Big Endian – High byte first Little Endian – Low byte first
Freeze Enabled	User data transmission Synchronization control commands enable the synchronization of inputs. Freeze Mode field is unchecked by default.
Sync Enabled	User data transmission Synchronization control commands enable the synchronization of outputs. Sync Mode is unchecked by default.

Table 3.10 – Field Device PROFIBUS configuration parameters

The PROFIBUS configuration is shown in the figure below. The Device PROFIBUS configuration window is opened by either double clicking on the slave device in the tree or right-clicking the slave device and selecting *Configuration*.

3.11.3. DPV1

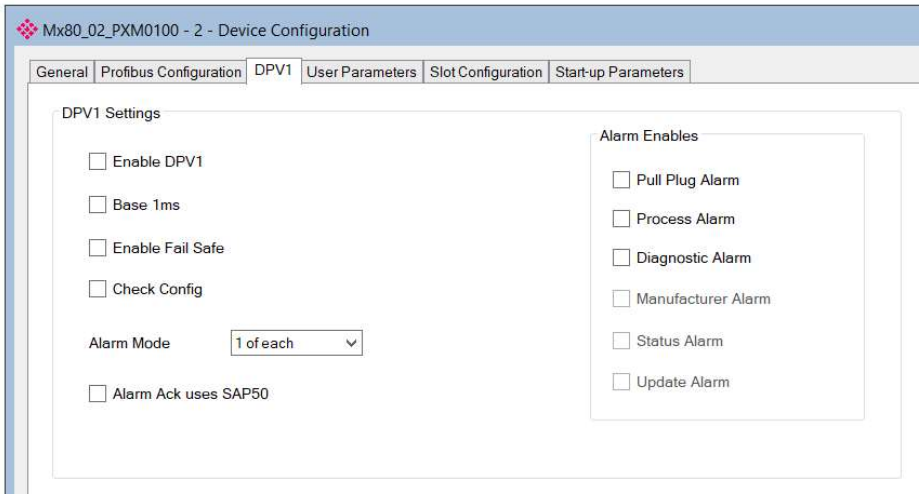


Figure 3.46 – Device DPV1 configuration parameters

The DPV1 configuration consists of the following parameters, the availability and default value of which are derived from the GSD file:

Parameter	Description
Enable DPV1	Indicates if the slave supports DPV1 Class 1 access (read and write) or alarms. **If the device does not support these DPV1 services, this parameter must be unchecked**. The default value is based on the information provided by the GSD File.
Base 1ms	Indicates if the device should use the 1ms base time for watchdog time calculation. See the chapter "**PROFIBUS Settings**" below for watchdog time calculation. By default, the field will be unchecked which sets the watchdog base to 10 ms. **Note**: the watchdog value is always shown in the configuration panel in ms regardless of this time base setting.
Enable Fail Safe	The failsafe mode determines the behavior of the DP Slave outputs when the PROFIBUS Master is in CLEAR state: • If the slave is configured to be failsafe and supports this feature, then it will apply its own fallback value (the Master sends outputs with 0 length data) • If not, the Master sends output data at 0 If this feature is supported by the device, the check box must be checked. **If the device does not support it, this parameter must be unchecked**. The default value is based on the information provided by the GSD File.
Check Config	This checkbox is used to define the reaction to the reception of configuration data. If the check box is not set, the check is as described in EN 50170. If the check box is set, the check is made according to a specific user definition. By default, the field will be unchecked.
Alarm Mode	This parameter specifies the maximum number of possible active alarms for the device.
Alarm Ack uses SAP50	This will force the PXM to use Service Access Point (SAP) 50 to acknowledge alarms.
Alarm Enables	Enables specific alarms for the slave device that the slave device will report on if active.

Table 3.11 – Device DPV1 configuration parameters

The DPV1 configuration is shown in the figure below. The Field Device DPV1 configuration window is opened by either double clicking on the slave device in the tree or right-clicking the slave device and selecting *Configuration*.

3.11.4. User Parameters

The User Parameter configuration consists of the device specific user configuration. This is extracted from the device GSD file and can be used configure device specific parameters. When one of the parameters is changed the User Parameter Data will be updated which is sent to the device in the Set Parameter telegram.

The User Parameter configuration is shown in the figure below.

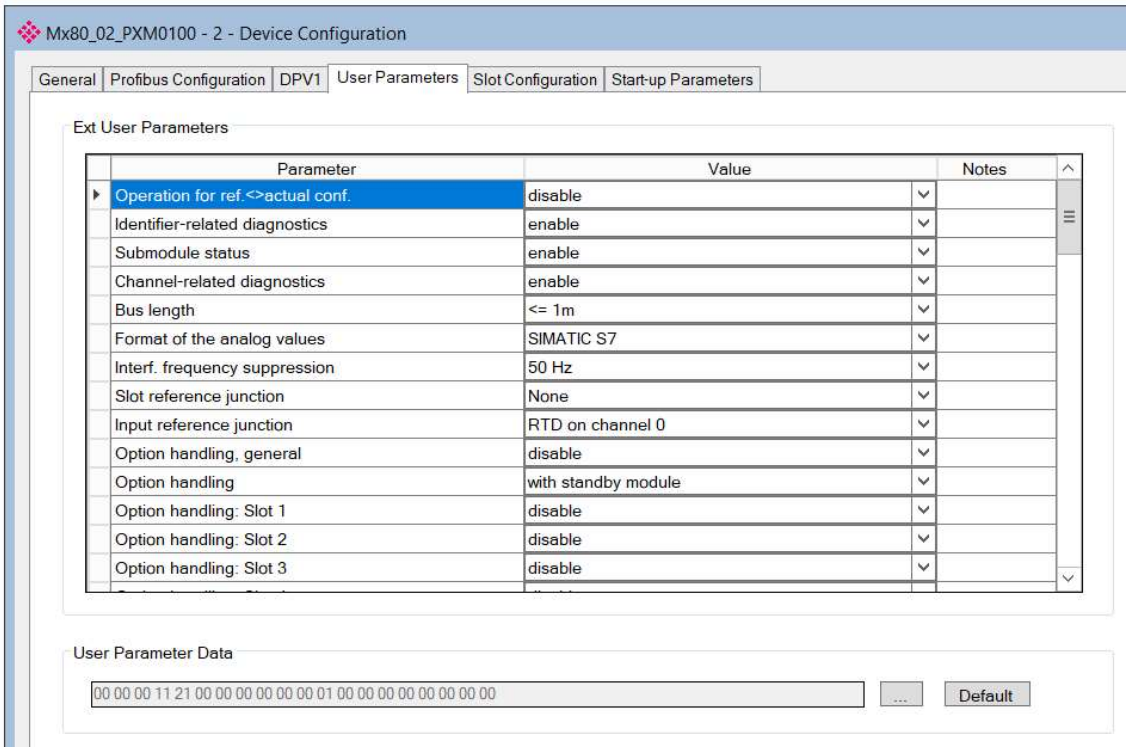


Figure 3.47 – Device User Parameter configuration parameters

3.11.5. Slot Configuration

Each slave device can have multiple slots that can be configured. A slot can be a place holder for a process variable or a placeholder for a specific piece of hardware. In the below example the PROFIBUS slave device added is an IO adapter which can have multiple additional IO connected which will be represented as additional slots.

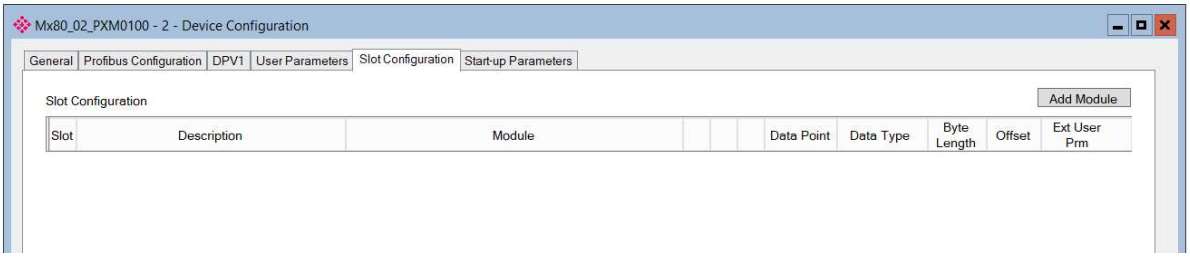


Figure 3.48 – Field Device Slot configuration start

When adding a slot, the data format and size will be configured for that specific slot as shown below.

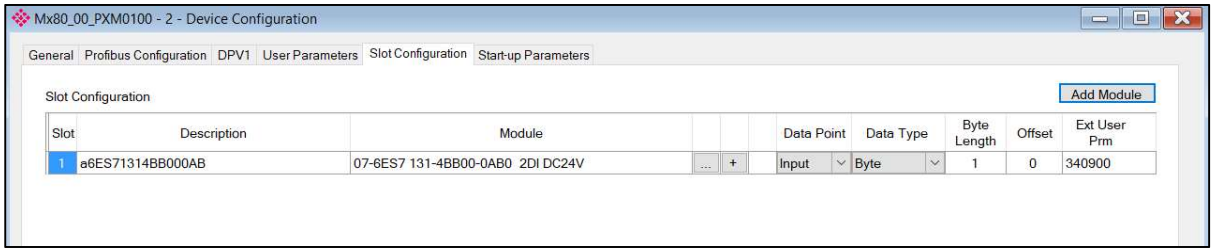


Figure 3.49 – Device Slot configuration added

Each slot also has the option to provide parameters specific to that slot and can be changed by clicking the "…" button as shown below:

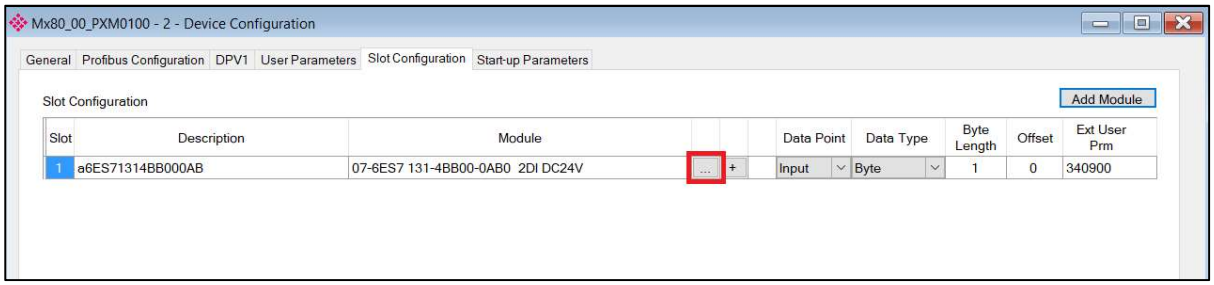


Figure 3.50 – Device Slot configuration additional parameters launch

Once the slot parameters have been updated the user can click the *OK* button which will updated the Extended User Parameters and return to the Slot Configuration page.

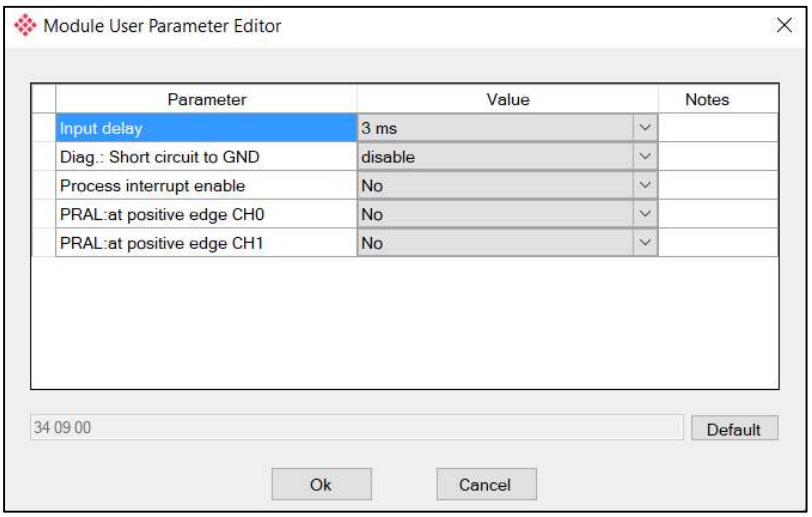


Figure 3.51 – Device Slot configuration additional parameters

3.11.6. START-UP PARAMETERS

Each slave device can have a set of start-up parameters associated with it which will be updated once Data Exchange is active using DPV1 Class 1 messaging. Thus, the user can have specific parameters that must be updated after the device is initialized for data exchange which will simplify device replacement.

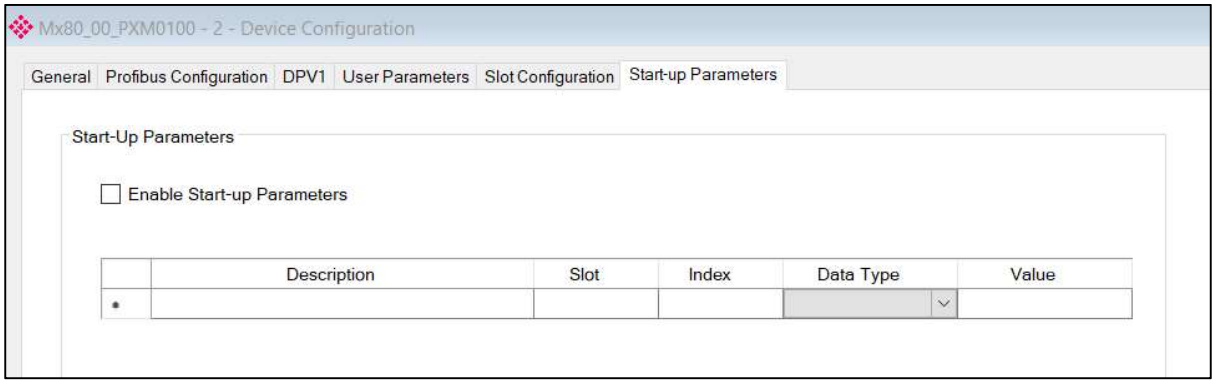


Figure 3.52 – Device Start-up Parameters

The user will need to enable the Start-up parameters by selecting the *Enable Start-Up Parameters* checkbox. Then the user will need to enter the required start-up parameters as shown below.

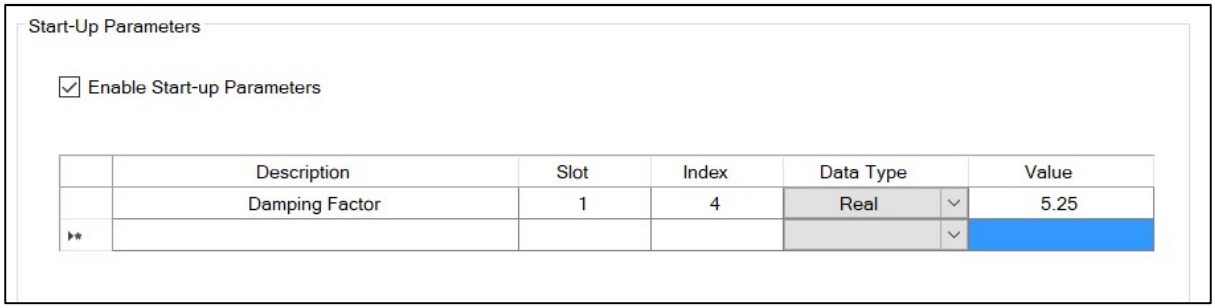


Figure 3.53 – Device Start-up Parameters Example

Once the slave device has been successfully parameterized and configured for Data Exchange the PXM will update one parameter at a time for each slave device.

3.11.7. MAPPING REPORT

A mapping report can be generated by for the PXM by right-clicking on the PXM in the project tree and selecting Mapping Report. The report comprises a short summary of the configured connections and the connection offsets and CRCs of each slave device.

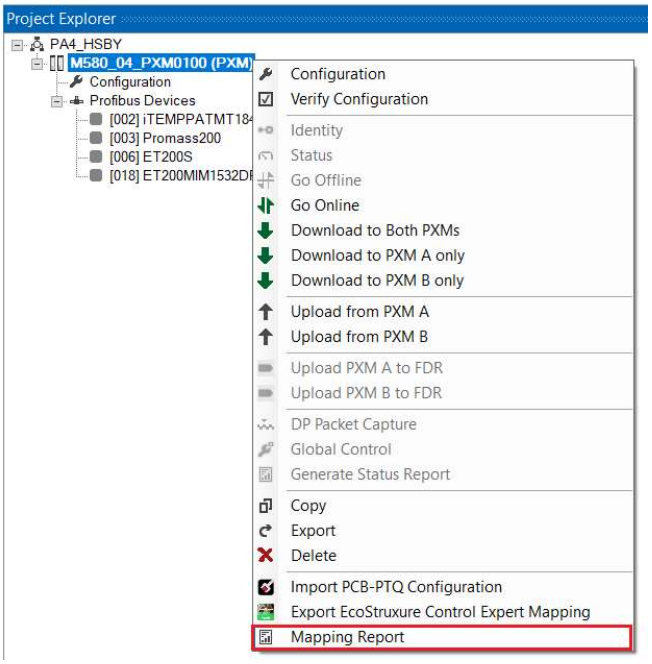


Figure 3.54 – Select Mapping Report

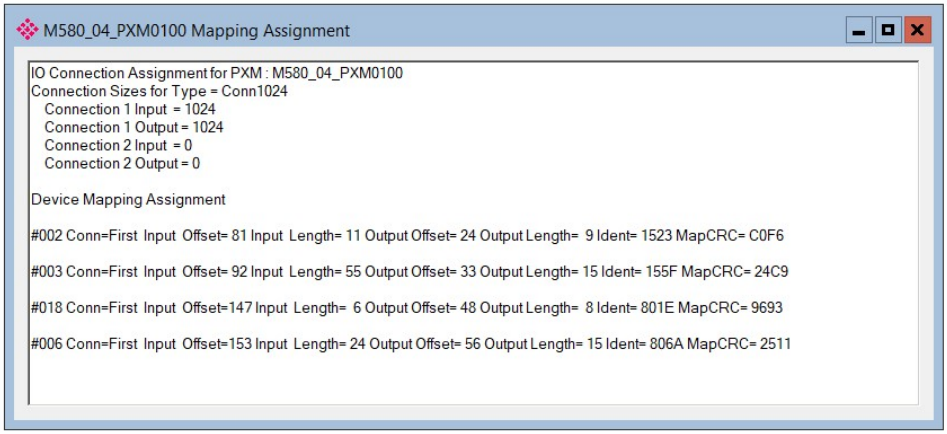


Figure 3.55 – Mapping Report

3.12. PROFIBUS DEVICE BULK INSTANTIATION

The ProSoft Configurator for Modicon provides two methods for bulk instantiating of PROFIBUS Devices:

- Copy and Paste / Paste Special
- Import/Export Device Lists

These functions allow large PROFIBUS networks to be configured in short time.

3.12.1. COPY, PASTE AND PASTE SPECIAL

To Copy a device right-click on the device and select the ***Copy*** option.

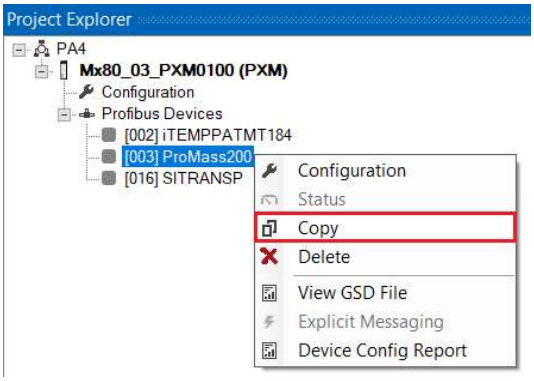


Figure 3.56 – Copy Device

To Paste a single copy of the previously copied device, right-click on the ***PROFIBUS Devices*** tree item and select the ***Paste*** option.

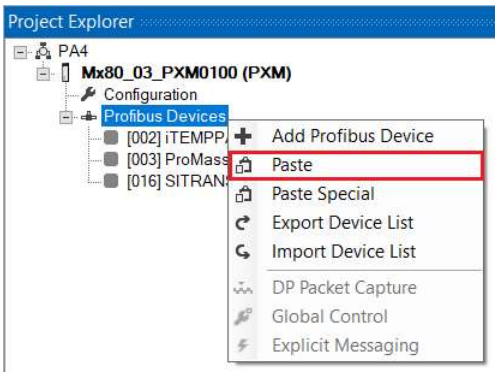


Figure 3.57 – Paste (single)

The Paste function will create a new instance of the previously copied device and assign it the first available Station Address. The instance name will be modified to ensure it is unique. The user can then edit the new device as required.

For additional options to bulk instantiate the previously copied device, right-click on the ***PROFIBUS Devices*** tree item and select the ***Paste Special*** option.

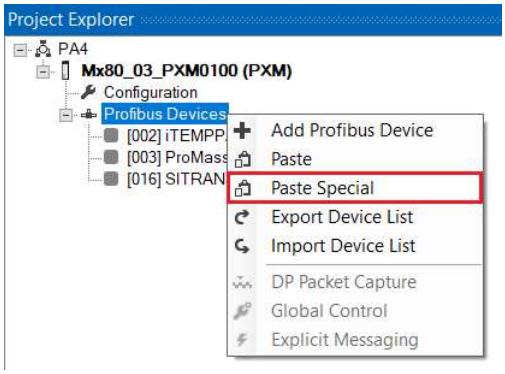


Figure 3.58 – Paste Special

The Paste Special options window provides a number of user-options.

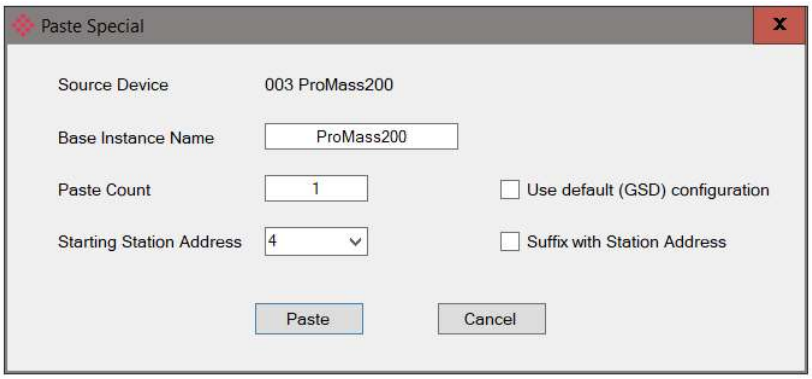


Figure 3.59 – Paste Special Options

Parameter	Description
Source Device	The Station Address and Instance Name of the source device. (Read-Only)
Base Instance Name	The Base Name or Prefix of the instantiated device/s.
Paste Count	The number of new device instances to be created.
Starting Station Address	The Station Address of the first instantiated device. The others will follow sequentially thereafter.
Use default (GSD) Configuration	If checked, then any customization of the source device will be ignored and the basic configuration from the GSD file will be used. If unchecked, then the customization of the source device will be used.
Suffix with Station Address	If checked, then the Station Address will be appended to the name of each instantiated device. If unchecked, then an incremental suffix will be appended to the name of each instantiated device.

Table 3.12 – Paste Special Parameters

3.12.2. Import / Export Device Lists

The instantiation of multiple PROFIBUS devices can also be achieved by importing of a CSV (comma separated variable) file. The format of the file is as follows:

Column	Description
Station ID	The Station Address of the device.
Instance Name	The Instance Name of the device.
GSD File / Copy of Station ID	To specify a new device, use the full GSD filename. To specify the new device to be a copy of an existing device, use the Station Address of an existing device.

Table 3.13 – Import/Export Device List Columns

The simplest way to generate the Device List is to first export the existing devices. This can be achieved by, right-clicking on the ***PROFIBUS Devices*** tree item and selecting the ***Export Device List*** option.

The created file can then be saved to a user-configurable location.

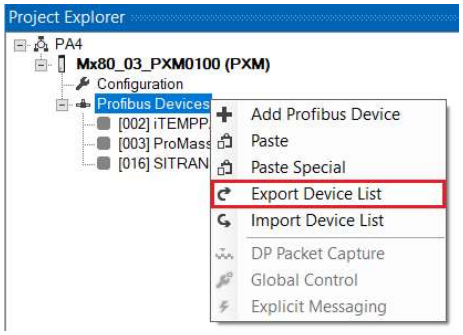


Figure 3.60 – Export Device List

Once the file has been modified (using Microsoft Excel or similar) then it can be imported by right-clicking on the ***PROFIBUS Devices*** tree item and selecting the ***Import Device List*** option.

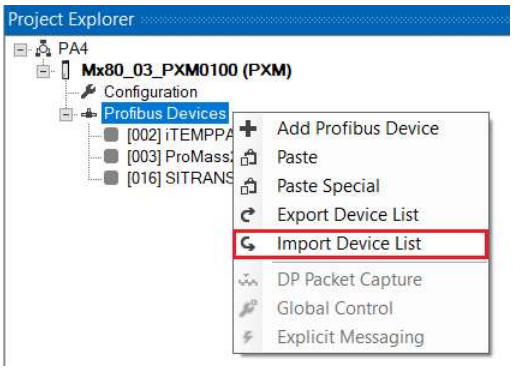


Figure 3.61 – Import Device List

3.13. CONTROL EXPERT CONFIGURATION

The PXM can be easily integrated into Schneider's M580 architecture. The PXM will need to be added to the M580 backplane in Control Expert and configured as per the configuration in

the ProSoft Configurator for Modicon. The following sections must carefully be followed to ensure that PXM correctly added in Control Expert.

When several M580 systems are connected on the same Ethernet network it is possible for one M580 CPU to establish a connection to the incorrect PXM if two PXMs were located in the same slot of different M580 systems.

⚠ WARNING
UNINTENDED EQUIPMENT OPERATION Do not use a PXM in a multi-M580 system in a flat-architecture before ensuring all architecture rules have been validated. **Failure to follow these instructions can result in death, serious injury, or equipment damage.**

3.13.1. CONFIGURE PROJECT SETTINGS

After creating a new M580 project, it is important to enable the use of Dynamic Arrays, as this functionality is required by the PXM mapping Derived Function Blocks.

This can be enabled by selecting the ***Project Settings*** option under the Control Expert ***Tools*** menu. In the Project Settings window, select the ***Variables*** left menu item and then check the ***Allow Dynamic Arrays (ANY_ARRAY_XXX)*** option.

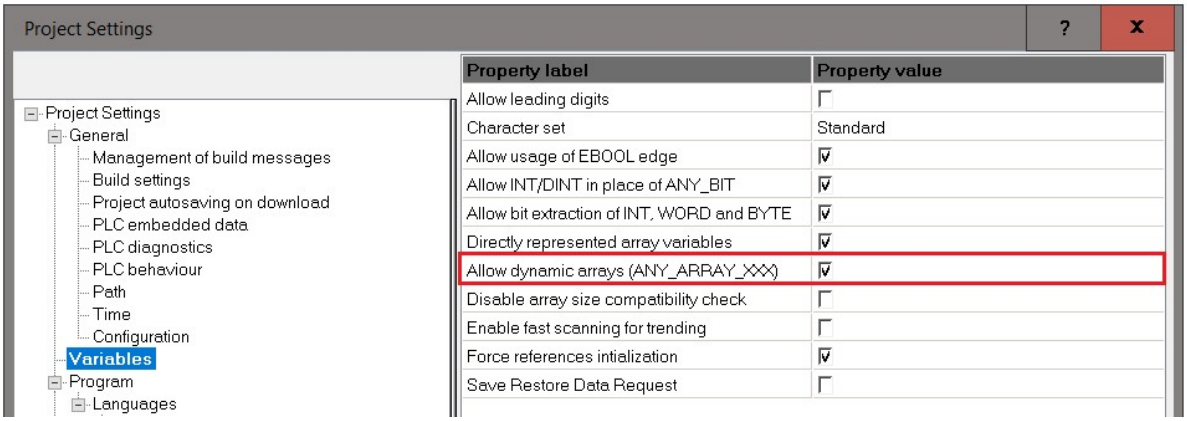


Figure 3.62 – Project Settings

3.13.2. INSTANTIATE PXM (PLACEHOLDER)

To instantiate the PXM device, right click on the preferred slot position and select ***New Device***.

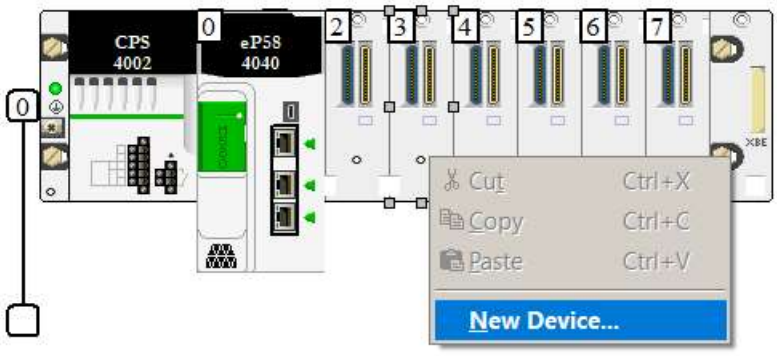


Figure 3.63 – Adding PXM in Control Expert

Within the list, expand the ***Third party products*** and select the ***PME PXM 0100***.

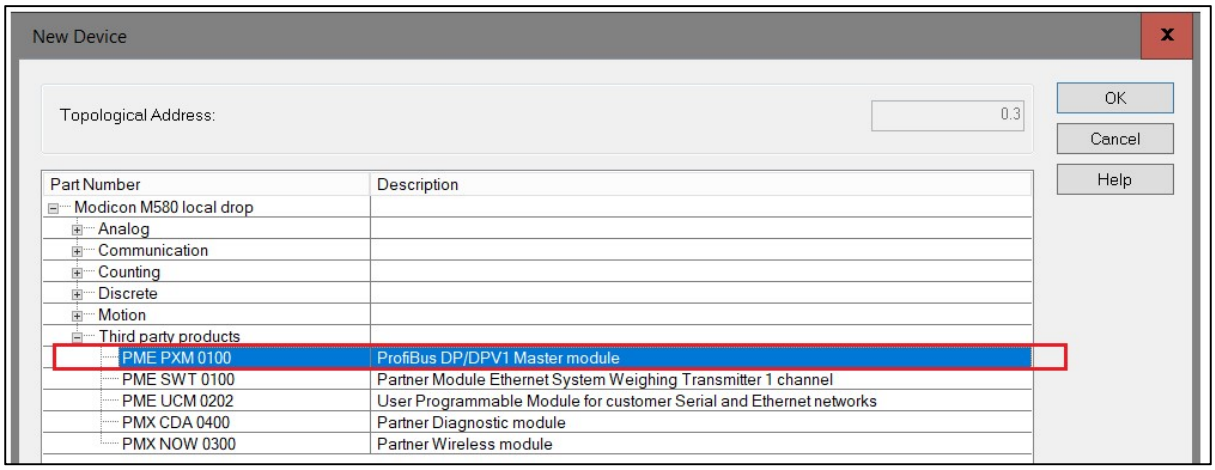


Figure 3.64 – Selecting PXM in Control Expert

Confirm that the module has been added to the rack.

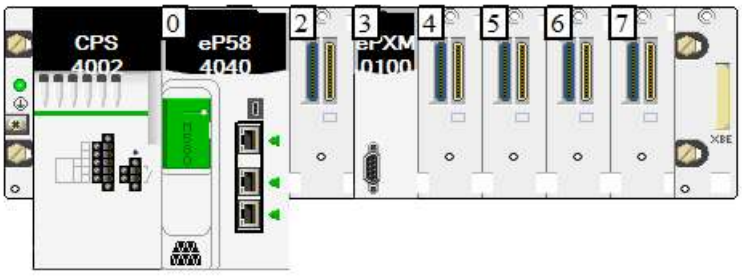


Figure 3.65 – Confirm PXM has been added to M580 backplane.

3.13.3. INSTANTIATE PXM DTM (GENERIC EDS)

Within the ***DTM Browser***, right-click on the CPU and select ***Add***.

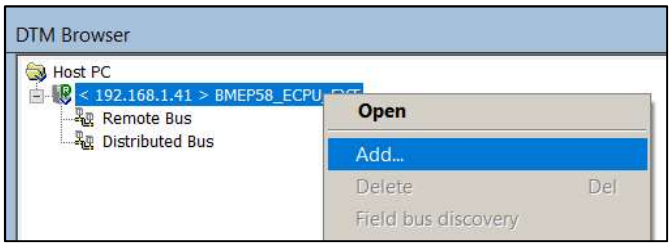


Figure 3.66 – Add DTM

Select the PME_PXM (from EDS) DTM.

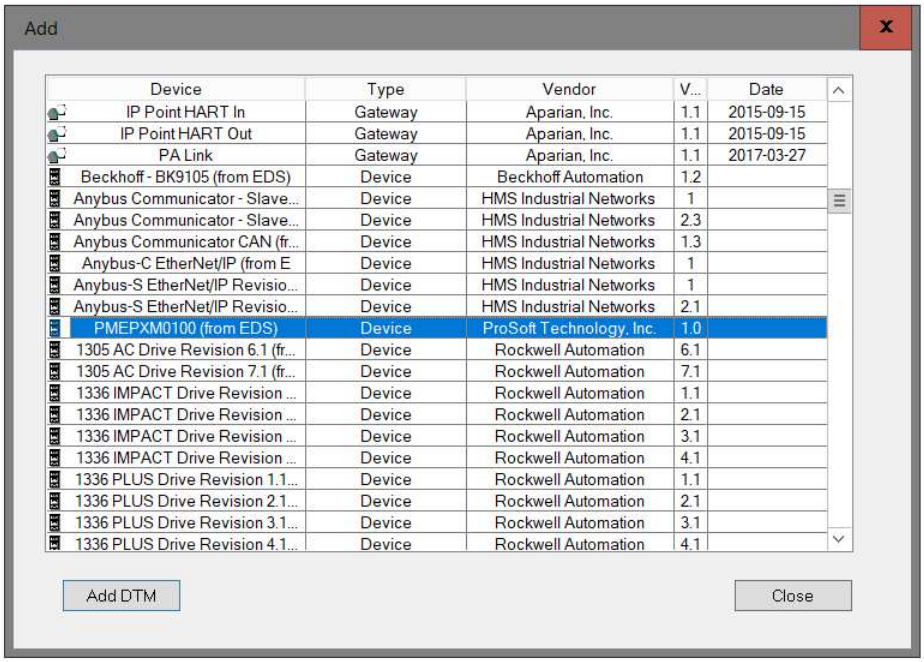


Figure 3.67 – Select PXM DTM

In the device properties, modify the DTM Name of the module to comply with the following rules:

Architecture	Name Rule	Name Example
Standalone Local rack	**Mx80_{Slot}_PXM0100**	Mx80_03_PXM0100
Standalone Remote rack	**Cyyy_{Slot}_PXM0100** (*Where yyy indicates the CRA rotary switch.*)	C002_03_PXM0100
HSBY Local rack A	**M58A_{Slot}_PXM0100**	M58A_03_PXM0100
HSBY Local rack B	**M58B_{Slot}_PXM0100**	M58B_03_PXM0100

Table 3.14 – PXM Control Expert Name Rules

NOTICE
THE PXM WILL NOT BE ABLE TO COMMUNICATE WITH THE M580 CONTROLLER The configured Instance Name will need to match the name given in Control Expert for the PXM DTM or the PXM will not communicate with the M580 controller. **Failure to follow these instructions may result in an unexpected behaviour.**

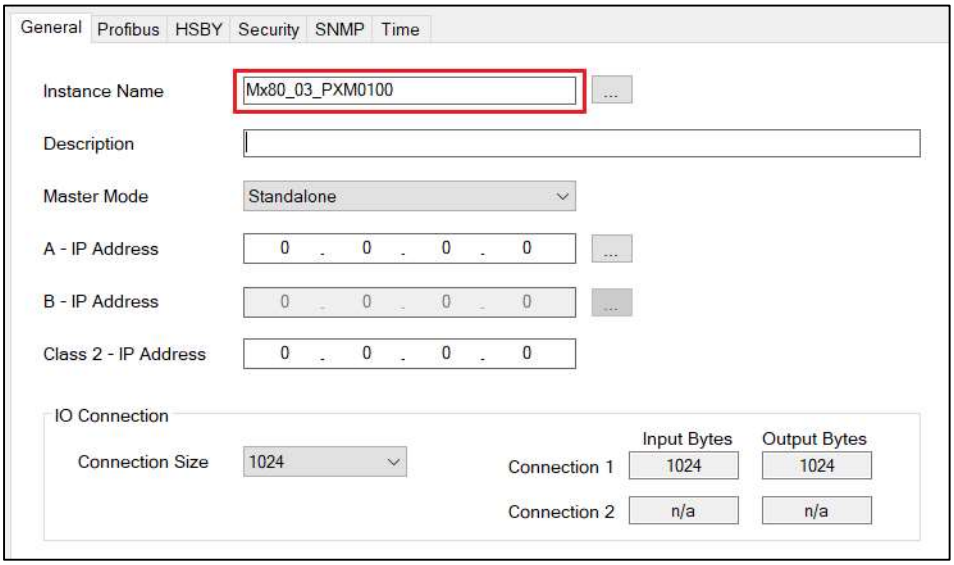


Figure 3.68 – Confirm matching name in Configuration Utility

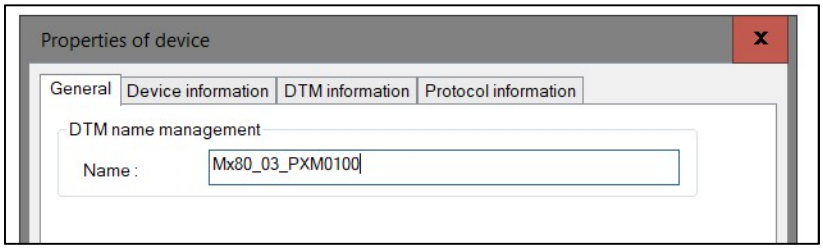


Figure 3.69 – Enter Control Expert PXM DTM Name

Confirm the DTM has been added in the DTM Browser tree.

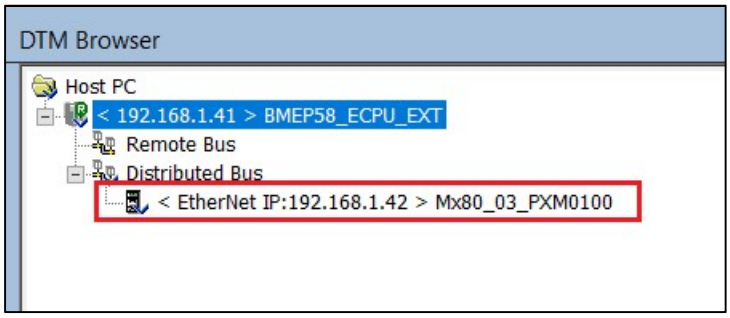


Figure 3.70 – Enter PXM DTM Name

3.13.4. MODIFYING PXM IP ADDRESS

To edit the PXM's address settings, in the DTM Browser, right-click on the CPU and select ***Device menu***, and then ***Configuration***.

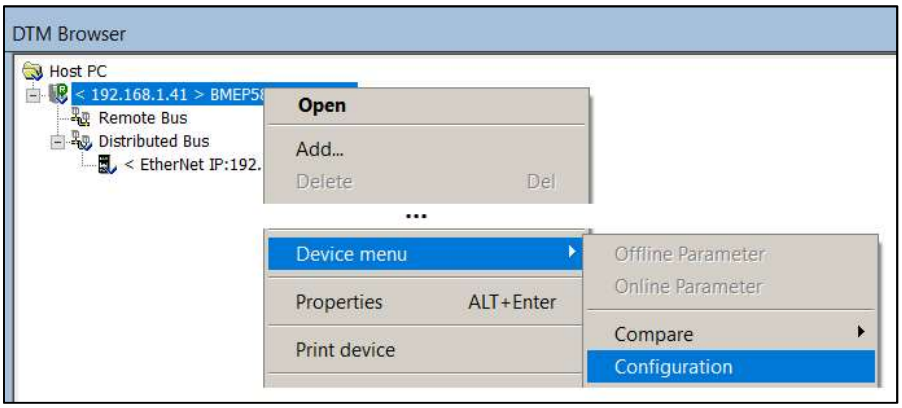


Figure 3.71 – CPU Device Configuration

On the tree on the left, under the Device List, select the PXM. On the *Address Setting* tab set the following parameters (as shown below):

- Configure the preferred *IP Address.*
- Enable the *DHCP* for this device.

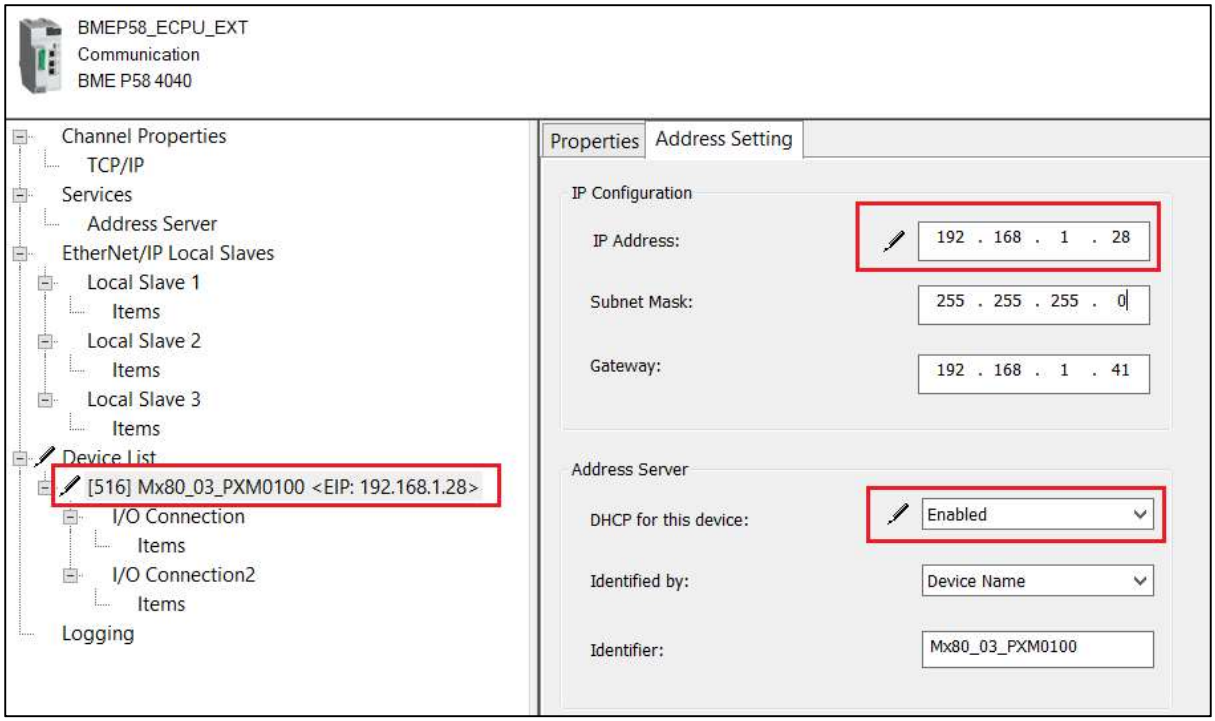


Figure 3.72 Update PXM network address settings

Once done click the *Ok* button.

NOTICE
THE PXM WILL NOT BE ABLE TO COMMUNICATE WITH THE M580 CONTROLLER Ensure that the IP address configured for the PXM in the ProSoft Configurator for Modicon matches that in the DTM configuration. **Failure to follow these instructions may result in an unexpected behaviour.**

The PXM does not support the DHCP assignment by MAC address and therefor the ***Identified By*** parameter in the ***Address Server*** section must be set to ***Device Name***. Selecting any other option will prevent the PXM from being assigned an IP address.

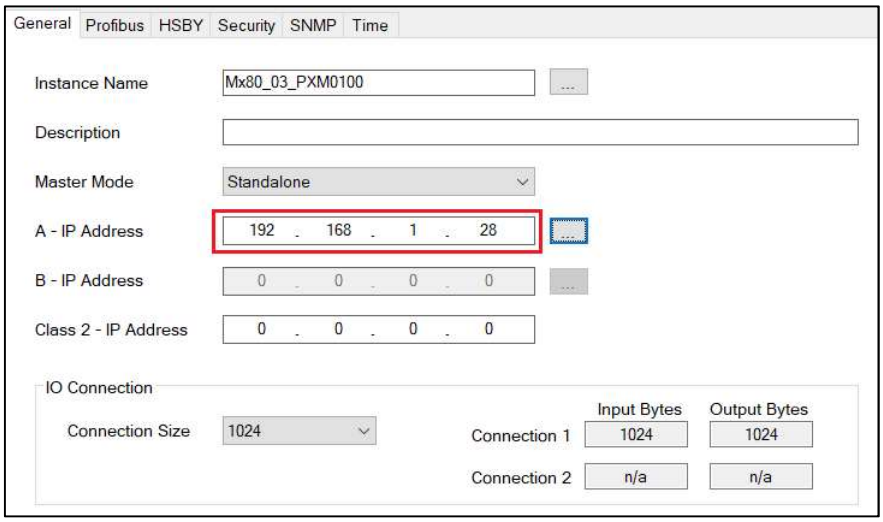


Figure 3.73 – Confirm matching IP address in Configuration Utility

3.13.5. MODIFYING PXM CONNECTION SETTINGS

To edit the PXM's connection settings, in the DTM Browser, right-click on the PXM and select ***Device menu***, and then ***Configuration***.

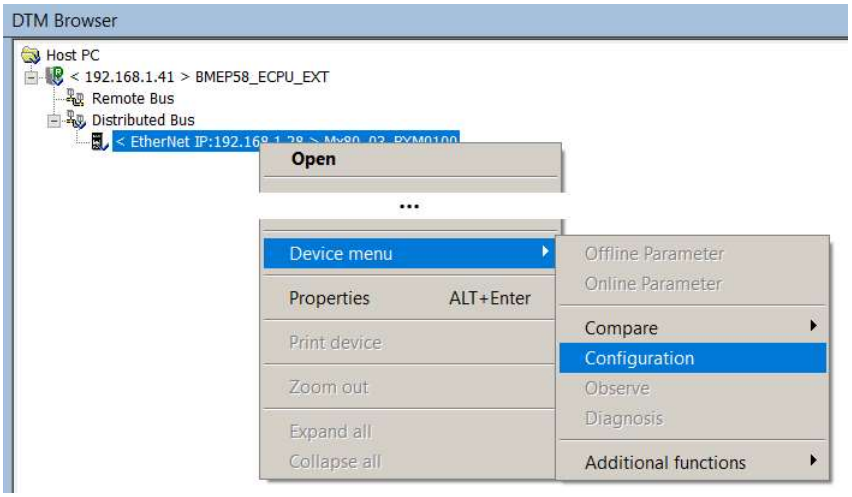


Figure 3.74 – PXM Configuration

Within this configuration: IO connections can be added and removed. For each connection, the RPI and Input and Output size can be configured. The same RPI value must be used for the ***Input T->O*** and the ***Output O->T*** configurations. The RPIs must also be the same for multiple connections. See the next section on RPI recommendations.

⚠ ***CAUTION***
UNINTENDED EQUIPMENT OPERATION When adding a second connection ensure that ***I/O Connection2*** is selected. **Failure to follow these instructions can result in injury or equipment damage.**

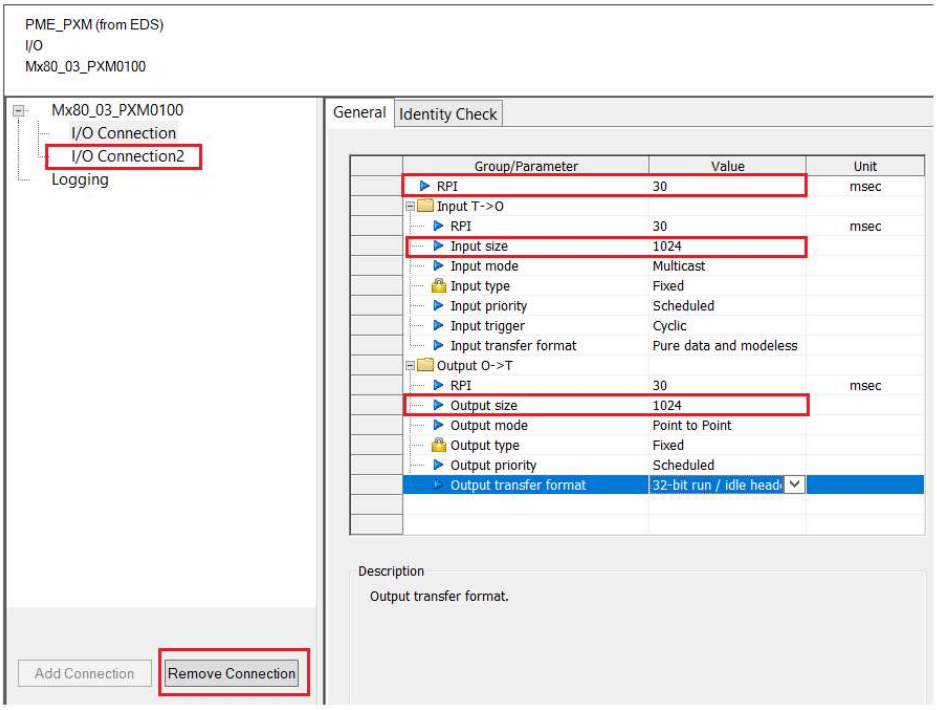


Figure 3.75 – PXM Connection Configuration

Once the selecting has been made click the *Ok* button.

NOTICE
THE PXM WILL NOT BE ABLE TO COMMUNICATE WITH THE M580 CONTROLLER Ensure that the connection size and connection count configured for the PXM in the ProSoft Configurator for Modicon matches that in the DTM configuration. **Failure to follow these instructions may result in an unexpected behaviour.**

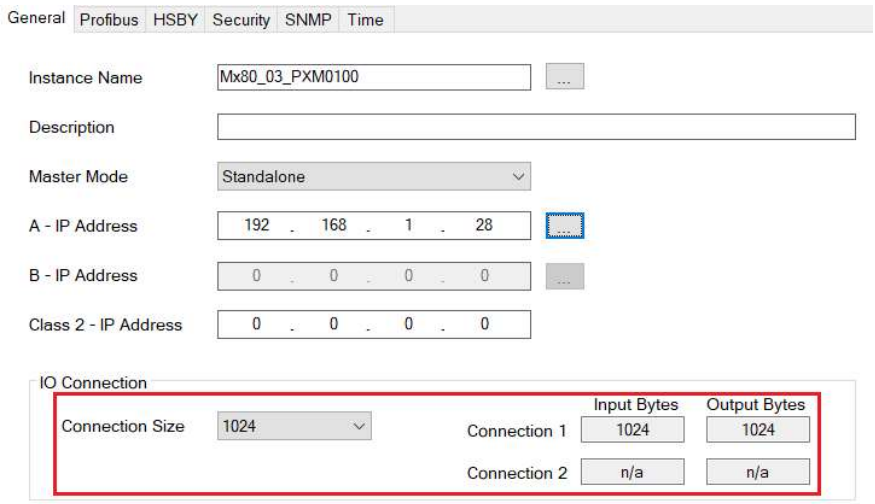


Figure 3.76 – Confirm matching connection parameters in Configuration Utility

3.13.6. PXM RPI RECOMMENDATIONS

The Requested Packet Interval (RPI) defines how often cyclic data is exchanged between the PXM and the M580 CPU. The exchanged data is processed in the MAST task, and thus the effective update rate also depends on the period of the MAST task.

Ideally the RPI and MAST task period should be less than the configured PROFIBUS Cycle to ensure the EtherNet/IP communication is not impeding performance. Where possible, the RPI should be set to half of the PROFIBUS Cycle.

The RPI must comply with the following constraints:

- RPI must be greater than or equal to 5ms
- RPI must be less than or equal to 1000ms
- RPI must be the same for the ***Input T->O*** and the ***Output O->T*** connections.
- RPI must be the same for both connections (where a second connection is used).

In an HSBY system, the choice of MAST task period and RPI will affect certain HSBY parameters. See chapter 5 for more information.

3.13.7. PXM Mapping Export/Import for Control Expert

Once the PXM and its PROFIBUS slave devices have been configured, the mapping configuration can be exported for later import into Control Expert. Right-click on the PXM module and select the *Export Control Expert Mapping* menu item.

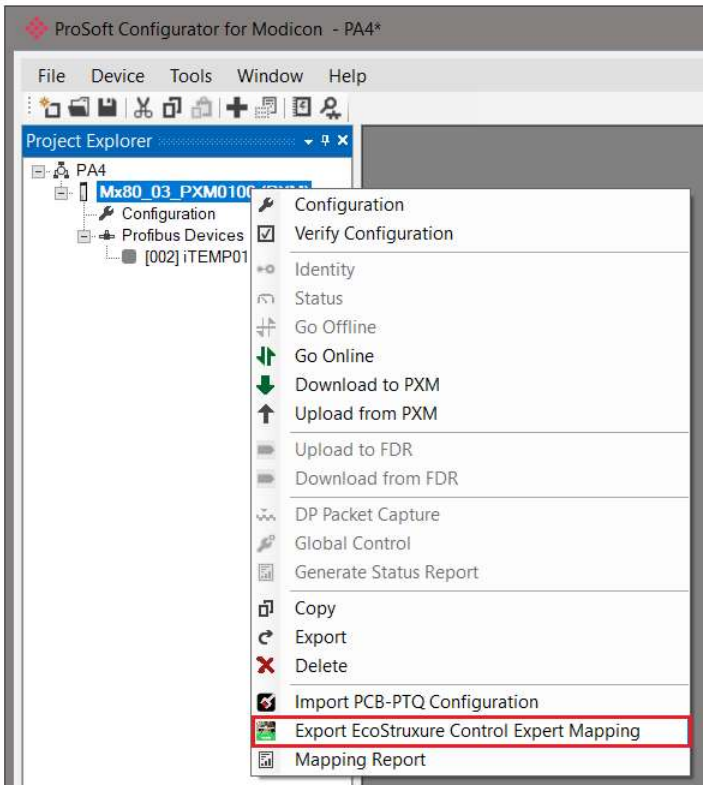


Figure 3.77 – Export Control Expert Mapping

In the Export options, select the Instantiation Format (Function Block or Structured Text) and select the Destination File followed by clicking *Export*.

The ***I/O Scan Mode*** defaults to ***Legacy***. This can be optionally changed to ***Enhanced*** mode for EcoStruxure Hybrid DCS mode. Note that the mode must match that configured in the Control Expert Project Settings.

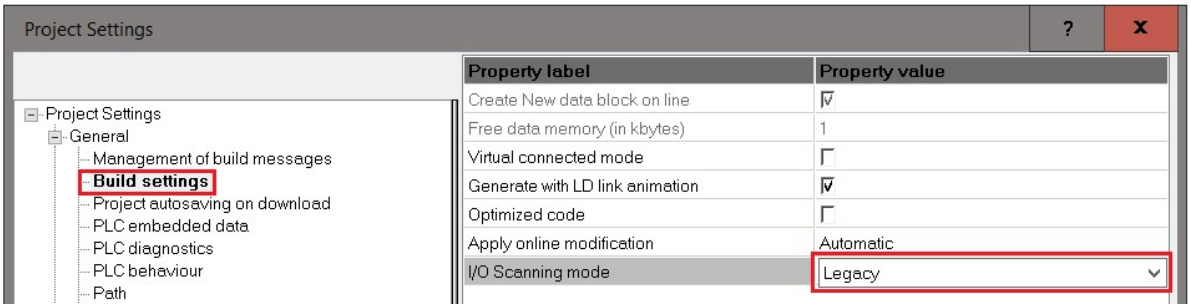


Figure 3.78 – Project Settings – I/O Scanning Mode

Selecting the "***Set Slave Device Enables***" option will automatically set all the configured slave devices' ***DeviceEnable*** bits to 1.

NOTICE
CONTROL EXPERT WILL GENERATE BUILD ERRORS FOLLOWING AN IMPORT The selected I/O Scan mode will need to match that configured in the Control Expert project settings. **Failure to follow these instructions will result in errors being generated when attempting to build the Control Expert application.**

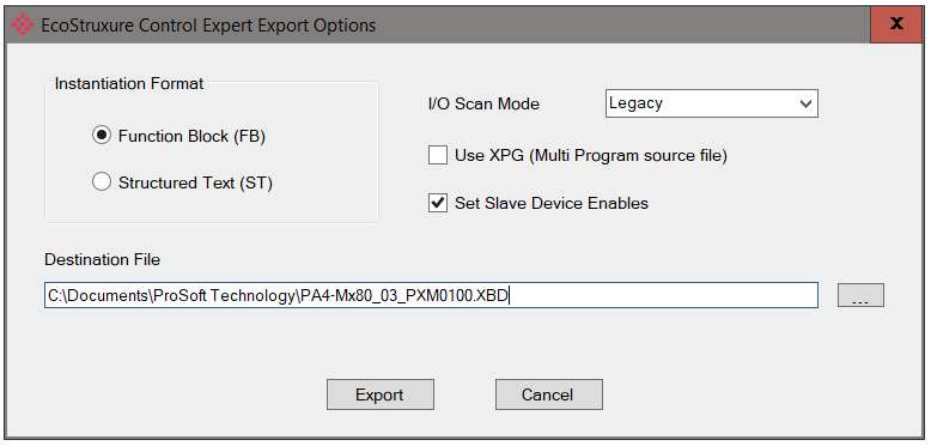


Figure 3.79 – Export Control Expert Mapping to File

In the Control Expert project navigate to *Program* – *Tasks* – *MAST* – *Sections*. Right-click on the *Sections* item and select *Import* as shown below.

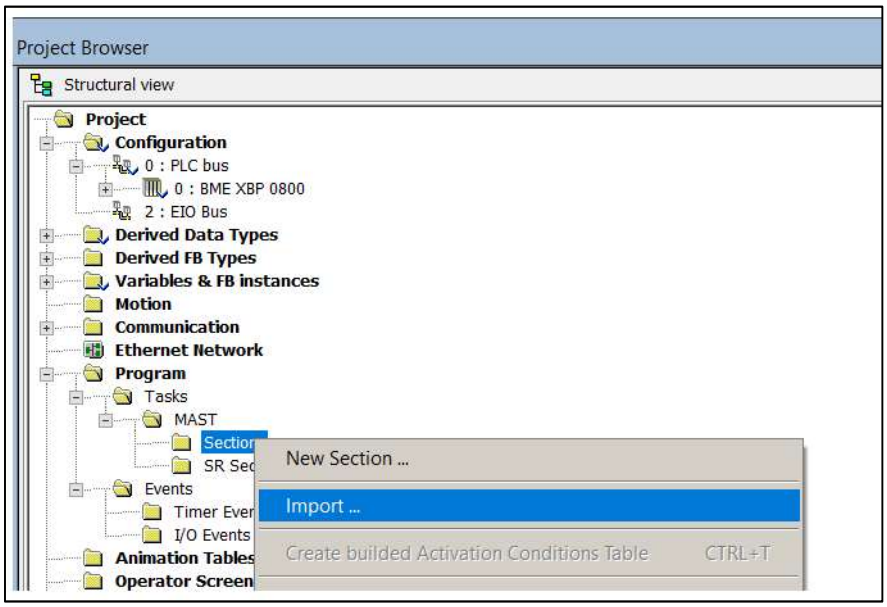


Figure 3.80 – Import Control Expert Mapping from File

In the Import file browser, select the Exported file created from the ProSoft Configurator for Modicon. Once the import has completed, confirm the new mapping section has been created.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

Subsequent file imports after changes have been made will result in object conflicts. It is important to select the ***Replace*** option to resolve such conflicts and continue with the import. Failing to do this will create a mismatch between the mapping code in the M580 and the PXM resulting in Slave devices not exchanging data correctly with the M580 CPU.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

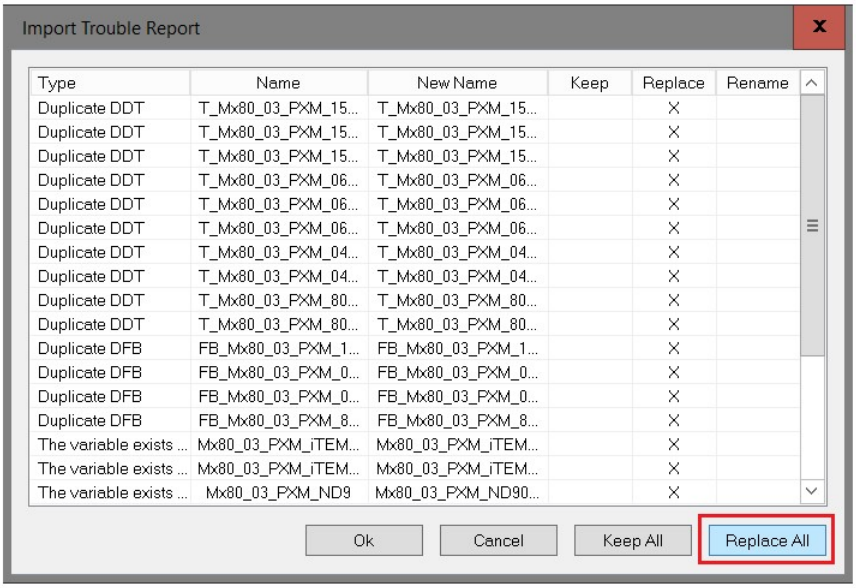


Figure 3.81 – Import Trouble Report

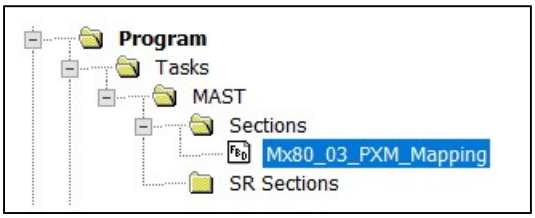


Figure 3.82 – Imported Control Expert Mapping

A number of specific mapping DDTs and DFBs are also created during the import process.

The device mapping DDTs are based on the instance name of the PXM. Incorrectly configuring the PXM's instance name can cause incorrect mapping DDTs and DFBs to be generated causing missing or erroneous device data and status in the M580 CPU.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

Ensure that the mapping configuration in the PXM matches that in Control Expert. Mismatched configurations can cause missing or erroneous device data and status in the M580 CPU.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

In an HSBY system the first section in the MAST task executes in both the Primary and Standby PLC. For this reason, it is important that the imported mapping section is not the first section. It is recommended that the PXM mapping section execute before any application code to ensure the application code receives the up-to-date PROFIBUS data.

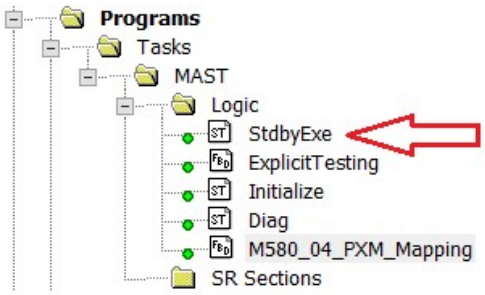


Figure 3.83 – First Section of MAST Task

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

Ensure that PXM mapping section is not the first section in an HSBY system.

The first section of the MAST Task executes in both the Primary and Standby PLCs. Executing the PXM mapping section in the Standby PLC will cause unexpected HSBY behaviour.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

3.13.8. Download to M580 Controller

Once the above configurations have been done the user will need to build and download the Control Expert project to the M580 controller.

In Control Expert, rebuild the project by selecting Rebuild All Project under the Build menu. Next download the project to the M580 controller by firstly selecting *Connect (*under the PLC menu) and once connected select *Transfer to PLC*.

After the project has downloaded, the M580 will assign the specified IP address to the PXM, at which stage the BS LED will flash green (see the Diagnostics section).

After downloading a new configuration to a PLC with a PXM in a remote (CRA) rack , the PLC will connect to the PXM before the CRA. When the PLC connects to the CRA, the CRA forces the PXM to disconnect and reconnect.

4. OPERATION

4.1. PROFIBUS DP

Once the PXM and M580 controller have been correctly configured the PXM will be ready to start exchanging data with PROFIBUS Slave Devices that have the ***DeviceEnable*** bit set. The user will need to set the PROFIBUS Operating mode from the PXM output assembly in the M580 controller (see the *Control Expert Mapping* section).

The PXM Operational State controls the PROFIBUS Fieldbus state as illustrated in the figure below.

		Field Bus State					
		0	1	2	3	5	6
PXM Operational State		Idle	No-Conf	Offline	Stop	Operate	Clear
0	Initialization						
1	Unconfigured		X				
2	Configured (Unconnected)			X			
3	Connected-Stop			X	X		
4	Connected-Run					X	
5	Connected-Clear (Fallback)						X

Table 4.1 – Relationship between PROFIBUS Fieldbus State and Operational State

The user will be able to see if there are any faults (e.g. configured device not found) by viewing the LEDs of the PXM (see the *Diagnostics* section for more details), by going online with the module in the ProSoft Configurator for Modicon and viewing the PXM and Device Diagnostics, or by viewing the input assembly of the PXM in Control Expert.

4.2. CONTROL EXPERT CONNECTION

When the PXM is connected to the M580 controller the *BS* LED will be solid green (see the *Diagnostics* section) and the module will report that it is owned in the ProSoft Configurator for Modicon. If the connection to the M580 controller is lost the *BS* LED will start flashing green.

The user can also verify that the PXM is connected to the controller by viewing the *Freshness* tag in the DDT that was created for the PXM during the mapping import. This value will be 1 if the PXM is connected and 0 if the PXM is **not** connected.

4.3. CONTROL EXPERT MAPPING

When the PXM mapping is imported into Control Expert a DDT is created for the PXM with the name:

Mx80_{xx}_PXM_Master

Where **xx** is the PXM slot number. Each PROFIBUS slave device that was configured in the ProSoft Configurator for Modicon will also have a DDT with the name:

Mx80_{xx}_PXM_{yy}

Where xx is the PXM slot number and the **yy** is the instance name of the device configured in the ProSoft Configurator for Modicon (see *PROFIBUS DP Device Parameters* section).

When operating in a HSBY architecture the DDT names will change as follows:

M580_{xx}_PXM_Master
M580_{xx}_PXM_{yy}

4.3.1. PXM MASTER DDT (T_PXM_MASTER)

The PXM Master DDT can be broken up into four sections; Status, Control, Input, and Output.

Name	Type	Comment
M580_04_PXM_Master	T_PXM_Master	
M580_04_PXM_Master.Status	T_PXM_MasterStatus	PXM Master Status
M580_04_PXM_Master.Status.Freshness	BOOL	PXM Connected and Updating. (0:Not updating, 1:Updating)
M580_04_PXM_Master.Status.DataValid	BOOL	Data Valid - (0:Not Valid, 1:Valid - PXM connected and mapping DFB executing)
M580_04_PXM_Master.Status.SourceIsB	BOOL	HSBY PXM Data Source (0:PXM A 1:PXM B)
M580_04_PXM_Master.Status.Connected	BOOL	PLC is Connected to PXM (0:Not Connected, 1:Connected)
M580_04_PXM_Master.Status.ConfigValid	BOOL	PXM Configuration is Valid. (0:Not Valid, 1:Valid)
M580_04_PXM_Master.Status.HSBYModeConfigured	BOOL	PXM Configured for HSBY. (0:Standalone, 1:HSBY)
M580_04_PXM_Master.Status.HSBYMastersConfigMismatch	BOOL	Configuration of PXM A and B do not match. (0:Ok, 1:Mismatch)
M580_04_PXM_Master.Status.HSBYActive	BOOL	HSBY mode is Active. (0:Standalone, 1:HSBY)
M580_04_PXM_Master.Status.HSBYPartnerOk	BOOL	PXM can communicate with partner PXM. (0:Fail, 1:Ok)
M580_04_PXM_Master.Status.ProfibusRunning	BOOL	Profibus Running. (0:Not running, 1:Running)
M580_04_PXM_Master.Status.PartnerConnected	BOOL	Partner PXM Connected to PLC. (0:Not Connected, 1:Connected)
M580_04_PXM_Master.Status.PartnerProfibusRunning	BOOL	Partner PXM's Profibus Running. (0:Not running, 1:Running)
M580_04_PXM_Master.Status.MultipleENIPConnection	BOOL	Indicates more than 1 ENIP connection used. (0:Single Connection, 1:Multiple Conne
M580_04_PXM_Master.Status.DuplicateDPStation	BOOL	Duplicate Profibus Station Address Detected. (0:No Duplicate, 1:Duplicate Detecte
M580_04_PXM_Master.Status.MasterCRC	UINT	Master Checksum of the PXM configuration.
M580_04_PXM_Master.Status.ActiveNodeCount	BYTE	Number of Profibus Slave Devices Online.
M580_04_PXM_Master.Status.LiveList	ARRAY[0..126] OF BOOL	Indicates Profibus Live List - Devices Online (Device 0 - 126)
M580_04_PXM_Master.Status.AlarmPending	ARRAY[0..126] OF BOOL	Indicates which Devices have Pending Alarms (Device 0 - 126)
M580_04_PXM_Master.Status.DiagPending	ARRAY[0..126] OF BOOL	Indicates which Devices have Pending Diagnostics (Device 0 - 126)
M580_04_PXM_Master.Status.DeviceName	STRING	PXM Instance Name.
M580_04_PXM_Master.Status.PXMOperatingState	BYTE	PXM Operating State. (0:Init, 1:Unconfigured, 2:Unconnected, 3:Stop, 4:Run, 5:Clear)
M580_04_PXM_Master.Status.PXMRedundantState	BYTE	PXM Redundant State. (1:Standalone, 2:PrimaryAlone, 3:PrimaryAssisted,4:Standby
M580_04_PXM_Master.Status.FieldbusState	BYTE	PXM Fieldbus State. (0:Idle, 1:No Configuration, 2:Offline, 3:Stop, 5:Operate, 6:Clear)
M580_04_PXM_Master.Status.FieldbusHealth	BYTE	PXM Fieldbus Health. (0:Idle, 1:Fieldbus Error, 2:Device/s Fault, 3:Device/s Error, 4:I
M580_04_PXM_Master.Status.EthernetServiceStatus	BYTE	PXM Ethernet Service Status.
M580_04_PXM_Master.Status.IPAddress	ARRAY[0..3] OF BYTE	PXM IP Address.
M580_04_PXM_Master.Control	T_PXM_MasterControl	PXM Master Control
M580_04_PXM_Master.Control.ProfibusStop	BOOL	Profibus STOP command.
M580_04_PXM_Master.Control.ProfibusRun	BOOL	Profibus RUN command.
M580_04_PXM_Master.Control.ProfibusClear	BOOL	Profibus CLEAR command.
M580_04_PXM_Master.Control.IsHSBYSystem	BOOL	HSBY Flag - Set by DFB.
M580_04_PXM_Master.Control.DeviceEnable	ARRAY[0..126] OF BOOL	Slave Device Enable array.
M580_04_PXM_Master.Input	ARRAY[0..2559] OF BYTE	PXM Master Raw Input Data
M580_04_PXM_Master.Output	ARRAY[0..2559] OF BYTE	PXM Master Raw Output Data

Figure 4.1 – PXM Master DDT Example (T_PXM_Master)

4.3.1.1. STATUS (T_PXM_MASTERSTATUS)

Tag	Description
Freshness	This indicates if the PXM is connected to the M580 controller and updating the DDT. 1 – PXM is connected and updating the input assembly 0 – PXM is not connected to the M580 controller.
DataValid	This indicates that the following data points are valid. It is based on the Freshness (above) and the execution of the mapping block. 1 – Data is valid 0 – Data is invalid.
SourceIsB	When operating in a HSBY system, this tag will indicate to the M580 controller if the PXM is in the same rack as M580 controller A or B. 1 – PXM is in the same rack as M580 controller B. 0 – PXM is in the same rack as M580 controller A.
Connected	Indicates if the PXM is connected to the M580 controller. 1 – PXM is connected. 0 – PXM is not connected.
ConfigValid	Configuration has been downloaded to the PXM and is being executed. 1 – PXM has been successfully configured. 0 – PXM is not configured.
HSBYModeConfigured	The PXM has been configured to operate in a HSBY system. 1 – PXM has been setup for HSBY. 0 – PXM has been setup for Standalone.
HSBYMastersConfigMismatch	In a HSBY system the configuration of the active and standby do not match. 1 – Active and Standby PXM configurations do **not** match. 0 – Active and Standby PXM configurations match.
HSBYActive	The PXM is running as a HSBY partner. 1 – PXM is running as a HSBY partner. 0 – PXM is running in Standalone mode.
HSBYPartnerOk	The PXM can communicate with its partner PXM in a HSBY system. 1 – PXM is communicating with partner PXM. 0 – PXM is **cannot** communicate with partner PXM.
PROFIBUSRunning	The PXM is exchanging data on the PROFIBUS network. 1 – PROFIBUS network is in OPERATE or CLEAR state. 0 – PROFIBUS network is in OFFLINE or STOP state.
PartnerConnected	Indicates if the partner PXM is also connected to the M580 controller. 1 – Partner PXM is connected to M580 controller. 0 – Partner PXM is **not** connected to M580 controller.

PartnerPROFIBUSRunning	The partner PXM is exchanging data on the PROFIBUS network. 1 – PROFIBUS network on the partner PXM is in OPERATE or CLEAR state. 0 – PROFIBUS network on the partner PXM is in OFFLINE or STOP state.
MultipleENIPConnection	Indicates that multiple EtherNet/IP connections are being used.
DuplicateDPStation	Indicates that the PXM has detected another PROFIBUS DP station with the same station address as itself and has entered a temporary Back-off mode. 1 – Duplicate detected (Back-off mode active) 0 – Normal (No duplicate detected). In this condition the PXM will not communicate on the PROFIBUS DP network. Although the back-off time is approximately 5 seconds, should the conflicting DP master remain active on the PROFIBUS network, the PXM will continuously re-enter the back-off mode. Because the PXM will never interrogate a slave device with the same station address as itself, this duplicate detection would be triggered only by the addition of another DP Master on the PROFIBUS network. The duplicate detection and subsequent invoking of the Back-off mode would occur if either the additional DP master has the same station address as the PXM, or it is interrogating another slave device with the same station address as the PXM.
MasterCRC	The checksum value of the PXM configuration.
ActiveNodeCount	Number of PROFIBUS slave device nodes online on local PXM.
LiveList	Indicates the nodes that are online on the local PROFIBUS network. Each bit represents a node. When the specific bit is set ‘1’ then the device is online and when the bit is off ‘0’ the device is not on the PROFIBUS network. Bit 0 – Node 0 Online Bit 1 – Node 1 Online ……. Bit 126 – Node 126 Online
AlarmPending	Indicates the nodes that have an alarm pending on the local PROFIBUS network. Each bit represents a node. When the specific bit is set ‘1’ then the device has an alarm pending that must be unloaded and when the bit is off ‘0’ the device does not have an alarm pending. Bit 0 – Node 0 has an alarm pending Bit 1 – Node 1 has an alarm pending ……. Bit 126 – Node 126 has an alarm pending
DiagPending	Indicates the nodes that have diagnostics pending on the local PROFIBUS network. Each bit represents a node. When the specific bit is set ‘1’ then the device has diagnostics pending that must be unloaded and when the bit is off ‘0’ the device does not have any diagnostics pending.

	Bit 0 – Node 0 has diagnostics pending Bit 1 – Node 1 has diagnostics pending ……. Bit 126 – Node 126 has diagnostics pending
DeviceName	A string indicating the instance name of the PXM.
PXMOperatingState	Indicates the current PXM Operating State: 0 – Initialization 1 – Unconfigured 2 – Configured (Unconnected) 3 – Connected – STOP 4 – Connected – RUN 5 – Connected – CLEAR (Fallback)
PXMRedundantState	Indicates the PXM's Redundant State: 1 – Standalone 2 – Primary Alone 3 – Primary Assisted (Standby is available) 4 – Standby Alone 5 – Standby Assisting (Primary is available)
FieldbusState	Indicates the current Fieldbus State: 0 – Idle 1 – No Configuration 2 – Offline 3 – Stop 5 – Operate 6 – Clear
FieldbusHealth	Indicates the current Fieldbus Health: 0 – Idle 1 – Fieldbus Error 2 – Device/s Fault 3 – Device/s Error 4 – Device/s Fault and Error
EthernetServiceStatus	Detailed Ethernet Services Status. One bit for each user-observable feature (0: Not OK, 1: OK/NA) Bit 0 - Reserved RSTP Service 0 - service is not operating normally 1 - service is operating normally or disabled or not implemented Bit 1 - SNTP Service 0 - service is not operating normally 1 - service is operating normally or Disabled Bit 2 - Reserved Bit 3 - SNMP 0 - service is not operating normally 1 - service is operating normally or Disabled Bit 4 - FDR regular client 0 - unable to download PRM file (during CCOTF or HSBY switchover) 1 - service is operating normally

	Bit 5 - Firmware Upgrade 0 - Firmware Upgrade unauthorized 1 - service is operating normally Bit 6 – Web Server 0 - no web page available 1 - service is operating normally or Disabled Bit 7 - Syslog Server 0 - Event logging disrupted 1 - Service is operating normally or Disabled
IPAddress	PXM IP Address (Represented as 4-byte array)

Table 4.2 – Status section of PXM DDT (T_PXM_MasterStatus)

4.3.1.2. CONTROL (T_PXM_MASTERCONTROL)

Tag	Description
PROFIBUSStop	This bit sets the PROFIBUS state to STOP.
PROFIBUSRun	This bit sets the PROFIBUS state to OPERATE.
PROFIBUSClear	This bit sets the PROFIBUS state to CLEAR.
IsHSBYSystem	When the PXM has been configured to operate in a HSBY system this bit must be set from the M580 controller to enable the HSBY operating in the PXM. Note: This bit is controlled by the Master Mapping DFB.
DeviceEnable	These bits enable nodes on the PROFIBUS network for data exchange. Each bit represents a node. When the specific bit is set '1' then the device (if configured) will exchange data with the PXM and when the bit is off '0' the device does exchange data with the PXM. Bit 0 – Node 0 is enabled for data exchange Bit 1 – Node 1 is enabled for data exchange ……. Bit 126 – Node 126 is enabled for data exchange

Table 4.3 – Control section of PXM DDT (T_PXM_MasterControl)

Check that the correct PROFIBUS control bits are set.
The user must avoid setting more than one PROFIBUS state bit at a time (Stop, Run, Clear). Below is the order of priority for the state bits.

PROFIBUS Stop	PROFIBUS Run	PROFIBUS Clear	Description
1	x	x	PROFIBUS network is in STOP state.

0	1	x	PROFIBUS network is in OPERATE state.
0	0	1	PROFIBUS network is in CLEAR state.
0	0	0	PROFIBUS network is in OFFLINE state.

Table 4.4 – PROFIBUS network state control

A configuration mismatch can cause missing or erroneous device data.

⚠ *WARNING*

UNINTENDED EQUIPMENT OPERATION

Before placing the PROFIBUS network in RUN mode, ensure that the device mapping in the M580 CPU matches that downloaded to the PXM.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

4.3.1.3. INPUT / OUTPUT

The Input and Output arrays in the PXM DDT is used to store the data from the various slave devices. The data in these arrays are unformatted and thus not used by the user. The data in these arrays are copied to and from the Device DDTs described below (which are formatted in engineering units).

4.3.2. DEVICE DDT

The Device DDT can be broken up into four sections; Status, Control, Input, and Output.

4.3.2.1. STATUS

Tag	Description
DataValid	This indicates that the following data points are valid. It is based on the Freshness of the master data and the execution of the mapping block. 1 – Data is valid 0 – Data is invalid.
Online	This bit indicates if the device is online on the PROFIBUS network. 1 – Device is online 0 – Device is not online
DataExchangeActive	This bit indicates if the device is configured and exchanging data on the PROFIBUS network. 1 – Device is active and exchanging data 0 – Device is not exchanging data The user must ensure that all application code making use of data from a slave device first checks that the ***DataExchangeActive*** bit is 1.

IdentMismatch	The device configured in the ProSoft Configurator for Modicon and the device at the configured node address do not match because they have different ident numbers. 1 – Online device Ident does not match configured device 0 – Online device and configured device ident match
Disabled	This bit indicates if the device has not been enabled for data exchange in the PXM device enable control bits. 1 – Device has **not** been enabled for data exchange 0 – Device has been enabled for data exchange
ErrorFlag	This bit indicates an error with the device. 1 – Device has an error. 0 – Device has no error. The error flag will be set when one of the following conditions occur: • If there is an ident mismatch during slave parameterization, • When receiving any form of FDL fault (data link layer fault). For example: SAP Not Activated or Resource Not Available. • When the data size of the DPV0 data exchange does not match what has been configured in the PCM. This Error flag is transient and will clear once a valid response is received.
AlarmPending	Indicates the device has an alarm pending on the local PROFIBUS network. When the specific bit is set '1' then the device has an alarm pending that must be unloaded and when the bit is off '0' the device does not have an alarm pending. 0 – The node has no alarm pending 1 – The node has an alarm pending
DiagnosticsPending	Indicates the device has diagnostics pending on the local PROFIBUS network. When the specific bit is set '1' then the device has diagnostics pending that must be unloaded and when the bit is off '0' the device does not have any diagnostics pending. 0 – The node has no diagnostics pending 1 – The node has diagnostics pending
ToggleActionRequired	Indicates that a device has been add online (CCOTF) and the DeviceEnable bit was already on. This DeviceEnable must be toggled off and then back on before the device will go online. 0 – No action required 1 – Toggle (DeviceEnable) action required
InputMappingMismatch	This bit indicates that there is a mismatch between the device and Control Expert mapping. If there is a mismatch in the mapping between Control Expert and the PXM it can result in data appearing in the incorrect location which means the user can be looking at data which is not accurate, or the actual data expected.

	0 – The mapping for the input data is correct. 1 – There is a mapping mismatch in the input data.
InputStationMismatch	This bit indicates that there is a mismatch between the actual device station address and the expected Control Expert mapping station address. 0 – Station address matches 1 – Station address mismatch
OutputMappingMismatch	This bit indicates that there is a mismatch between the device and Control Expert mapping. If there is a mismatch in the mapping between Control Expert and the PXM it can result in data appearing in the incorrect location which means the user can be sending incorrect data to a device which can have unpredicted results. 0 – The mapping for the output data is correct. 1 – There is a mapping mismatch in the output data.
OutputStationMismatch	This bit indicates that there is a mismatch between the actual device station address and the expected Control Expert mapping station address. 0 – Station address matches 1 – Station address mismatch
DPFresh	This bit indicates that the PXM has received the first Profibus Cyclic data since RUN. 0 – Cyclic data has not been received (data invalid) 1 – Cyclic data has been received
StationNumber	The station address of the device.
DeviceState	Indicate the current State of the Slave device: 0 – Reserved 1 – Inoperative 2 – Idle 3 – Offline 4 – Stopped 6 – Operational 7 – Clear

Table 4.5 – Status section of Device DDT (*T_PXM_DeviceStatus*)

⚠ *WARNING*

UNINTENDED EQUIPMENT OPERATION

Ensure that all application code making use of data from a slave device first checks that the ***DataExchangeActive*** bit is 1.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

4.3.2.2. CONTROL

This section (***T_PXM_DeviceControl***) is reserved and should not be changed by the user.

4.3.2.3. INPUT / OUTPUT

The input and output for each slave device DDT will be custom made depending on the slot configuration made in the ProSoft Configurator for Modicon. Below is an example of the slot configuration and the slave device DDT input and output. The different colour dots represent each configured process variable in both the configuration as well as the DDT.

Should a slave device be disconnected, the input data will retain the last updated value. The user must ensure that all application code making use of data from a slave device first checks that the ***DataExchangeActive*** bit is 1.

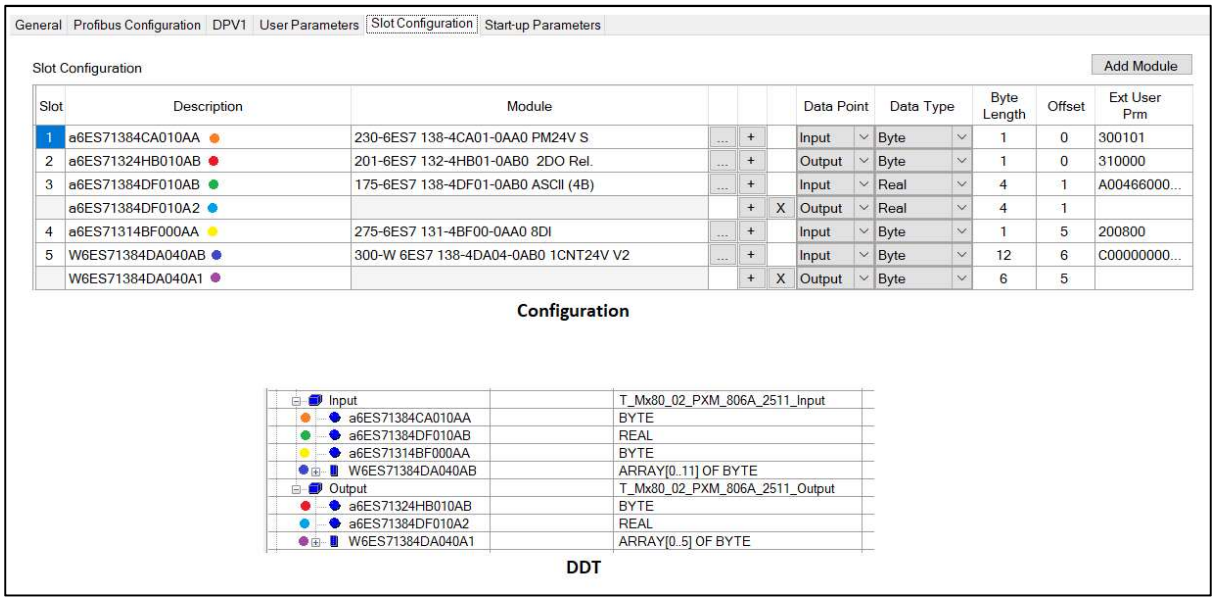


Figure 4.2 – Device DDT input and output

The mapping of implicit data from the PXM to the individual device DDTs is achieved using the mapping DFBs generated by the ProSoft Configurator for Modicon.

⚠ *WARNING*

ERRONEOUS DATA AND STATUS INFORMATION

Do not manually modify the mapping DDTs or DFBs.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

4.3.3. MASTER MAPPING DFB

The master mapping DFBs are responsible for the following functions:

1. Check connection to PXM is valid.
2. Combine PXM data from multiple connections (multiple connection only)
3. Map the Master Status information to the Master Status DDT sub-element ("Master.Status").
4. Map the Master Control information from the Master Control DDT sub-element ("Master.Control").
5. Map the PXM Input Data to the Master Input array ("Master.Input").
6. Map the PXM Output Data from the Master Output array ("Master.Output").
7. Set the HSBY enable bit.
8. Select data between PXM A and B (HSBY only)
9. Map the CPU HSBY status to the PXM (HSBY only)

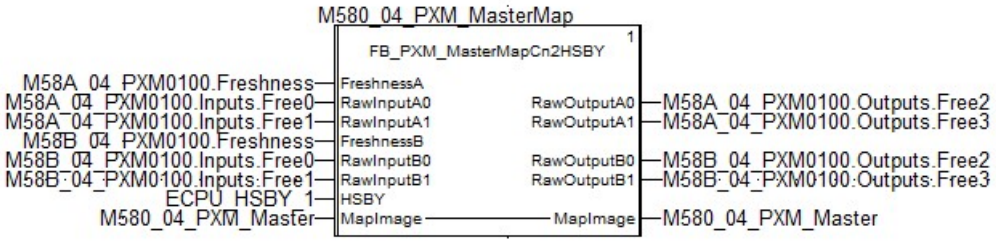


Figure 4.3 – Master Mapping DFB example

Depending on the PXM configuration one of the four possible Master Mapping DFBs will be instantiated. The Master Mapping DFBs are as follows:

DFB	EtherNet/IP Connection Count	CPU Architecture (Standalone / HSBY)
FB_PXM_MasterMapCn1	1	Standalone
FB_PXM_MasterMapCn2	2	Standalone
FB_PXM_MasterMapCn1HSBY	1	HSBY
FB_PXM_MasterMapCn2HSBY	2	HSBY

Table 4.6 – Master Mapping DFBs

The pins for the Master Mapping DFBs are defined as follows:

Pin Type	Pin	Data Type	Description
Input	**Freshness**	BOOL	Indicates the connection is active.

			(Standalone Only)
Input	**FreshnessA**	BOOL	Indicates the connection with PXM A is active. (HSBY Only)
Input	**FreshnessB**	BOOL	Indicates the connection with PXM B is active. (HSBY Only)
Input	**RawInput0**	ANY_ARRAY_BYTE	Input Data from first connection. (Standalone Only)
Input	**RawInput1**	ANY_ARRAY_BYTE	Input Data from second connection. (Standalone and dual-connection Only)
Input	**RawInputA0**	ANY_ARRAY_BYTE	Input Data from PXM A's first connection. (HSBY Only)
Input	**RawInputA1**	ANY_ARRAY_BYTE	Input Data from PXM A's second connection. (HSBY and dual-connection Only)
Input	**RawInputB0**	ANY_ARRAY_BYTE	Input Data from PXM B's first connection. (HSBY Only)
Input	**RawInputB1**	ANY_ARRAY_BYTE	Input Data from PXM B's second connection. (HSBY and dual-connection Only)
Input	**HSBY**	T_M_ECPU_HSBY_EXT	Status of system's HSBY. (HSBY Only)
Output	**RawOutput0**	ANY_ARRAY_BYTE	Output Data for first connection. (Standalone Only)
Output	**RawOutput1**	ANY_ARRAY_BYTE	Output Data for second connection. (Standalone and dual-connection Only)
Output	**RawOutputA0**	ANY_ARRAY_BYTE	Output Data for PXM A's first connection. (HSBY Only)
Output	**RawOutputA1**	ANY_ARRAY_BYTE	Output Data for PXM A's second connection. (HSBY and dual-connection Only)
Output	**RawOutputB0**	ANY_ARRAY_BYTE	Output Data for PXM B's first connection. (HSBY Only)
Output	**RawOutputB1**	ANY_ARRAY_BYTE	Output Data for PXM B's second connection. (HSBY and dual-connection Only)
In/Out	**MapImage**	T_PXM_Master	PXM Status, Control, Input and Output DDTs

Table 4.7 – Master Mapping DFB Configuration

Variable	Data Type	Description
ExtractErrIn0	INT	First Input Array Extraction Error (Standalone Only)
ExtractErrIn1	INT	Second Input Array Extraction Error (Standalone and dual-connection Only)
ExtractErrAIn0	INT	PXM A First Input Array Extraction Error (HSBY Only)
ExtractErrAIn1	INT	PXM A Second Input Array Extraction Error (HSBY and dual-connection Only)
ExtractErrBIn0	INT	PXM B First Input Array Extraction Error (HSBY Only)
ExtractErrBIn1	INT	PXM B Second Input Array Extraction Error (HSBY and dual-connection Only)
ExtractErrOut0	INT	First Output Array Extraction Error (Standalone Only)

ExtractErrOut1	INT	Second Output Array Extraction Error (Standalone and dual-connection Only)
ExtractErrAOut0	INT	PXM A First Output Array Extraction Error (HSBY Only)
ExtractErrAOut1	INT	PXM A Second Output Array Extraction Error (HSBY and dual-connection Only)
ExtractErrBOut0	INT	PXM B First Output Array Extraction Error (HSBY Only)
ExtractErrBOut1	INT	PXM B Second Output Array Extraction Error (HSBY and dual-connection Only)
ExtractNameIntErr	INT	PXM Device Name Array Extraction Error

Table 4.8 – Master Mapping DFB Public Variables

4.3.4. Device Mapping DFB

The device mapping DFBs are responsible for mapping the relevant sections of the PXM's Master Input and Output assembly data to the specific device DDTs. A Device Mapping DFB is automatically created by the PCM (Control Expert Mapping Export) for each unique device configuration.

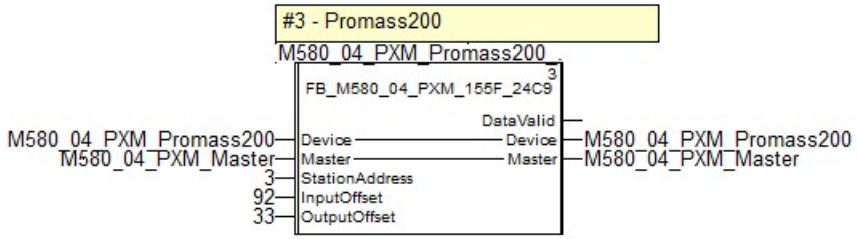


Figure 4.4 – Device Mapping DFB example

The pins for the Device Mapping DFBs are defined as follows:

Pin Type	Pin	Data Type	Description
In/Out	**Device**	[Device Specific DDT]	Device specific DDT (Status, Control, Input and Output Data)
In/Out	**Master**	T_PXM_Master	PXM Master Status, Control, Input and Output DDTs
Input	**StationAddress**	BYTE	Station Address of Device
Input	**InputOffset**	UINT	Offset of the start of the slave device's input data in the master input assembly.
Input	**OutputOffset**	UINT	Offset of the start of the slave device's output data in the master output assembly.
Output	**DataValid**	BOOL	This indicates that the data points are valid. 1 – Data is valid 0 – Data is invalid.

Table 4.9 – Device Mapping DFB Configuration

4.4. CHANGE CONFIGURATION ON THE FLY (CCOTF)

The PXM supports a number of CCOTF (Change Configuration on the Fly) functions, i.e. changing the PXM configuration while the M580 CPU and the PROFIBUS network are running.

These changes are only supported when the ProSoft Configurator for Modicon is online (connected) to the PXM. If the ProSoft Configurator for Modicon is not online then any subsequent changes will require an offline download to the PXM, which in turn will require either the PROFIBUS network to be in STOP mode or the M580 to be in STOP mode.

The allowed CCOTF changes can be summarised as follows:

- Master:
  - HSBY Switch-over parameters
  - Security settings (Service Enable, Access Control List)
  - SNMP, SysLog and Time Synchronisation settings
- Slave Devices
  - Add new device
  - Group membership
  - Watchdog settings
  - DPV1 settings
  - PROFIBUS User Parameters
  - Start-up parameters

Changes to the settings of an existing slave device, including DPV1 settings, and User Parameters may cause the slave device to go offline or stop exchanging data.

⚠ *CAUTION*
UNINTENDED EQUIPMENT OPERATION Ensure that any change to the settings of an existing slave device, including DPV1 settings, and User Parameters, is conducted when safe to do so.. **Failure to follow these instructions can result in injury or equipment damage.**

When adding a new device online, new mapping must be generated and imported into Control Expert to ensure there is no mismatch between the configurations of the M580 CPU and the PXM. Once the import into Control Expert is complete, the Build Changes options must be selected.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

Subsequent file imports after changes have been made will result in object conflicts. It is important to select the ***Replace*** option to resolve such conflicts and continue with the import. Failing to do this will create a mismatch between the mapping code in the M580 and the PXM resulting in Slave devices not exchanging data correctly with the M580 CPU.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

Ensure that the mapping configuration in the PXM matches that in Control Expert. Mismatched configurations can cause missing or erroneous device data and status in the M580 CPU.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

Check that HSBY settings are valid. Changing the HSBY settings with invalid parameter values may cause the PROFIBUS master to stop or become unstable. It is important to ensure valid parameters are entered especially during CCOTF.

4.5. EXPLICIT MESSAGING FUNCTION BLOCKS

The PXM supports DPV1 Class 1 (MS1) and Class 2 (MS2) messaging which can be used to read / write parameters in a slave device. The PXM DPV1 communication is achieved by using EtherNet/IP unconnected messaging (UCMM) or Class 3 connected messaging. The PXM can buffer up to 10 DPV1 messages at a time. By default, only 1 PROFIBUS Poll per cycle for such messaging is allowed. The user must ensure that the ***Extra DPV1 Poll / Cycle*** parameter is equal or greater to the maximum number of simultaneous explicit DPV1 messages that may be sent.

Check that the slave device supports DPV1 messaging before attempting it. The user must also set the ***DPV1 Enable*** bit in the user parameters of the slave device in the ProSoft Configurator for Modicon.

In order to simplify the execution of explicit DPV1 messaging a number of specific PXM DFBs are available:

- ID DFB – Maps connection parameters and optional Class 2 Initialization.
- RDRec DFB – DPV1 Read Record (Class 1 or Class 2)
- WRRec DFB – DPV1 Read Record (Class 1 or Class 2)

- RDDiag DFB – Read PROFIBUS Diagnostics
- RDAlarm DFB – Extract PROFIBUS Alarm
- GlobalControl DFB – Provide PROFIBUS Global Controls (Sync, Freeze)

These DFBs make use of one or more of the DPV1 Explicit (EtherNet/IP) messaging services described later in this chapter.

The availability of the explicit message types will depend on the current PROFIBUS DP field bus state as indicated in the table below.

Function / Fieldbus State	**Idle** (0)	**No Config** (1)	**Offline** (2)	**Stop** (3)	**Operate** (5)	**Clear** (6)
Class 2 Init	-	-	-	Yes	Yes	Yes
Class 2 Abort	-	-	-	Yes	Yes	Yes
DPV1 Read Class 1	-	-	-	-	Yes	Yes
DPV1 Read Class 2	-	-	-	Yes	Yes	Yes
DPV1 Write Class 1	-	-	-	-	Yes	Yes
DPV1 Write Class 2	-	-	-	Yes	Yes	Yes
Get Diagnostics	-	-	-	Yes	Yes	Yes
Alarm Extraction	-	-	-	-	Yes	Yes
Global Control	-	-	-	-	Yes	Yes

Table 4.10 – Explicit Function Availability

Excluding the PLC's "***IP Address A***" in the ACL configuration will prevent explicit messaging from operating.

⚠ *NOTICE*

UNINTENDED EQUIPMENT OPERATION

Configuring any PXM explicit messaging DFBs in the first section of the MAST Task in an HSBY system will cause erroneous and unexpected results. The first section of the MAST Task executes in both the Primary and Standby PLCs causing duplicate explicit messages.

Failure to follow these instructions may result in an unexpected behaviour.

4.5.1. ID DFB

The FB_PXM_ID block is required for all explicit messaging and provides 2 basic functions:

1. Maps the PXM connection parameters (IP address etc.)
2. Open a Class 2 connection to the slave device, when Class 2 is selected.

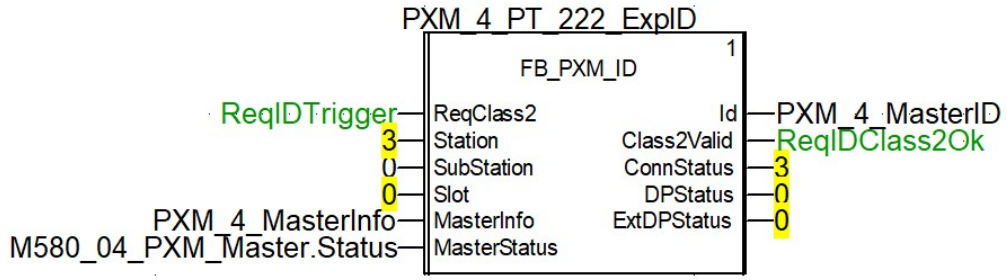


Figure 4.5 – ID DFB

The pins for the ID DFB are defined as follows:

Pin Type	Pin	Data Type	Description
Input	**ReqClass2**	BOOL	Used to Select Class 2 communication. On a 0 to 1 transition the Class 2 Initialization request is sent to the slave device. On a 1 to 0 transition the Class 2 Abort request is sent to the slave device.
Input	**Station**	BYTE	The Station Address of the slave device.
Input	**SubStation**	BYTE	The target sub-station address, if required.
Input	**Slot**	BYTE	The target slot number of the slave device.
Input	**MasterInfo**	T_PXM_Info	Master Information defining the path for the explicit message. The structure of the T_PXM_Info is shown below.
Input	**MasterStatus**	T_PXM_MasterStatus	The Master Status information which is produced by the standard PXM master mapping DFB.
Output	**Id**	T_PXM_Id	The resulting ID structure, which can be used by other explicit DFBs e.g. Class1Read, Class1Write etc.
Output	**Class2Valid**	BOOL	Output indicating whether the Class 2 connection is valid.
Output	**DPStatus**	BYTE	The PROFIBUS DP status of the response to the Class 2 Initialization request. See section 12.1 for the DP Status.
Output	**ExtDPStatus**	DINT	The Extended PROFIBUS DP status of the response to the Class 2 Initialization request. See section 12.2 for the Extended DP Status.

Table 4.11 – ID DFB Configuration

The required Info structure is shown below. The RackNumber, SlotNumber and ChannelNumber relate to the Ethernet port being used for the explicit message. For local and remote PXM applications the address is the CPU's backplane port (0,0,3).

PXM_4_MasterInfo		T_PXM_Info	
PXM_4_MasterInfo.RackNumber	0	BYTE	PLC Rack Number
PXM_4_MasterInfo.ModuleNumber	0	BYTE	PLC Module Number
PXM_4_MasterInfo.ChannelNumber	3	BYTE	PLC Channel Number

Figure 4.6 – ID DFB Info Structure

The primary output of the ID DFB is the ID variable which has the following structure:

PXM_4_MasterID		T_PXM_Id	
PXM_4_MasterID.ConnectionReference	0	BYTE	Class 2 Connection Reference.
PXM_4_MasterID.ConnectionStatus	3	BYTE	Connection Status.
PXM_4_MasterID.Station	3	BYTE	Device Station Address
PXM_4_MasterID.Slot	0	BYTE	Device Slot
PXM_4_MasterID.MasterInfo		T_PXM_Info	Communication Path to the Master

Figure 4.7 – ID DFB ID Output Structure

Connection Status	Description
0	Initializing
1	Class 2 - Starting connection
2	Class 1 Ready
3	Class 2 Ready
4	Class 2 Abort – Starting Abort Sequence
10	Class 2 Initialization Failed - Data Exchange Failed to Start
11	Class 2 Initialization Failed - Data Exchange Failed to Complete
12	Class 2 Initialization Failed - Data Exchange Failed / Invalid CIP Length
13	Class 2 Initialization Failed – Rejected on DP Layer (See DP Status)
14	Class 2 Abort Failed to complete.
15	Class 2 Initialization Failed – Invalid CIP Reply
16	Class 2 Initialization Failed – Invalid CIP Status

Table 4.12 – ID - Connection Status

The ID DFB also provides multiple parameters for debugging via its public variables.

PXM_4_PT_222_ExpID		FB_PXM_ID	
<inputs>			
<outputs>			
<inputs/outputs>			
<public>			
PXM_4_PT_222_ExpID.PubState	5	INT	Operational State
PXM_4_PT_222_ExpID.PubStatus	3	BYTE	Operational Status
PXM_4_PT_222_ExpID.PubCIPStatus	0	BYTE	CIP Status
PXM_4_PT_222_ExpID.PubDPStatus	0	BYTE	Class 2 Profibus DP Status
PXM_4_PT_222_ExpID.PubExtDPStatus	0	DINT	Class 2 Extended Profibus DP Status
PXM_4_PT_222_ExpID.PubDXCommReport	0	BYTE	Data Exchange Communication Report
PXM_4_PT_222_ExpID.PubDXOpReport	0	BYTE	Data Exchange Operational Report
PXM_4_PT_222_ExpID.PubCIPReplyLength	17	INT	CIP Reply Length

Figure 4.8 – ID DFB Public Variables

Variable	Data Type	Description
PubState	INT	The internal state of the block.
PubStatus	BYTE	The current status of the block. See table 4.14.
PubCIPStatus	BYTE	The CIP Status of the last received EtherNet/IP response. A value of 0 indicate success.
PubDPStatus	BYTE	The PROFIBUS DP status of the last received DP response. See section 12.1 for the DP Status.
PubExtDPStatus	DINT	The Extended PROFIBUS DP status of the response to the Class 2 Initialization request. See section 12.2 for the Extended DP Status.
PubDXCommReport	BYTE	M580 Data Exchange Communication Report provided by the DFB's internal DATA_EXCH instruction. See table 4.15 for details.
PubDXOpReport	BYTE	M580 Data Exchange Operation Report provided by the DFB's internal DATA_EXCH instruction. See table 4.16 for details.
PubCIPReplyLength	INT	The CIP data length of the last received EtherNet/IP response.

Table 4.13 – ID DFB Public Variables

Status Code	Description
0	Disabled
1	Pending
2	Class 1 Ready
3	Success / Class 2 Ready
4	Denied
5	Aborting
6	Invalid Fieldbus State
8	Data Extract Failed
9	User Array too small
10	Data Exchange Failed to Start
11	Data Exchange Failed to Complete
12	Data Exchange Failed / Returned Length Invalid
13	DP Status Failed
14	Abort Failed
15	CIP Service Mismatch
16	CIP Transaction Failed

17	Invalid ID Connection Status (Class 1/2)
18	Insufficient CIP Data
19	Write Length Mismatch
20	Request Rejected (CIP Error)
21	Invalid Length specified

Table 4.14 – Explicit DFB Status Codes

DX Comm Report Code	Description
0x00	Correct exchange
0x01	Exchange stop on timeout
0x02	Exchange stop on user request (CANCEL)
0x03	Incorrect address format
0x04	Incorrect destination address
0x05	Incorrect management parameter format
0x06	Incorrect specific parameters
0x07	Problem in sending to the destination
0x08	Reserved
0x09	Insufficient receive buffer size
0x0A	Insufficient send buffer size
0x0B	No system resources.
0x0C	Incorrect exchange number
0x0D	No telegram received
0x0E	Incorrect length
0x0F	Telegram service not configured
0x10	Network module missing
0x11	Request missing
0x12	Application server already active
0x13	UNI-TE V2 transaction number incorrect
0xFF	Message refused (See Operation Report)

Table 4.15 – Data Exchange – Comm Report Codes

DX Operation Report Code	Description
0x00	Positive result
0x01	Request not processed
0x02	Incorrect response
0x03	Reserved
Request code + 0x30	Upon positive reply for certain requests
0xFB	Upon reply to minor request
0xFD	Operational error
0xFE	Upon positive reply for certain request
The following, if the Communication Report = 0xFF	
0x01	No resources towards the processor
0x02	No line resources
0x03	No device or device without resources (*)
0x04	Line error
0x05	Length error
0x06	Faulty communication channel
0x07	Addressing error
0x08	Application error
0x0B	No system resources.
0x0C	Communication function not active
0x0D	Destination missing
0x0F	Intra-station routing problem / channel not configured
0x11	Address format not managed
0x12	No destination resources
0x14	Non-operational connection
0x15	No resource on the local channel
0x16	Access not authorized
0x17	Inconsistent network configuration
0x18	Connection temporarily unavailable
0x21	Application server stopped

0x30	Transmission error

Table 4.16 – Data Exchange – Operation Report Codes

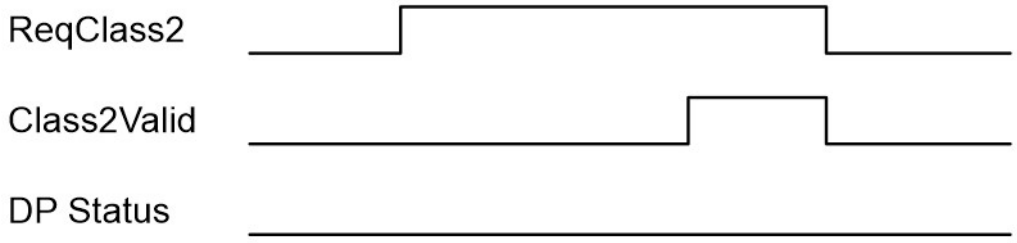


Figure 4.9 – ID DFB Timing Diagram - Success

Figure 4.10 – ID DFB Timing Diagram - Failure

4.5.2. RDREC DFB

The RDREC (Read Record) DFB provides DPV1 Read functionality using either Class 1 or Class 2 mechanisms. The selection whether Class 1 or Class 2 is used depends on the preceding ID DFB described in the previous section.

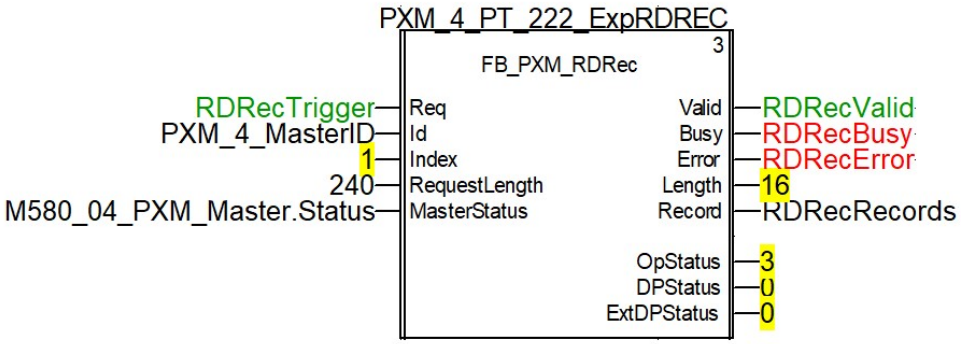


Figure 4.11 – RDRec DFB

The pins for the RDRec DFB are defined as follows:

Pin Type	Pin	Data Type	Description
Input	**Req**	BOOL	Used to Request the read action. A 0-1 transition triggers the read request.
Input	**Id**	T_PXM_Id	The Id structure received from the preceding ID DFB.
Input	**Index**	BYTE	The slave device's Index to be read.
Input	**RequestLength**	BYTE	The length (bytes) of the read request. (1-240 bytes)
Input	**MasterStatus**	T_PXM_MasterStatus	The Master Status information which is produced by the standard PXM master mapping DFB.
Output	**Valid**	BOOL	Read request was successful.
Output	**Busy**	BOOL	Read request is Busy.
Output	**Error**	BOOL	Read Request resulted in an error.
Output	**Length**	BOOL	The length (bytes) of the retuned record (0-240 bytes)
Output	**Record**	ANY_ARRAY_BYTE	The received data record as a byte array.
Output	**OpStatus**	BYTE	The Status of the DFB. See table 4.14
Output	**DPStatus**	BYTE	The PROFIBUS DP status of the last received DP response. See section 12.1 for the DP Status.
Output	**ExtDPStatus**	DINT	The Extended PROFIBUS DP status of the last received DP response. See section 12.2 for the Extended DP Status.

Table 4.17 – RDRec DFB Configuration

The RDRec DFB also provides multiple parameters for debugging via its Public variables.

PXM_4_PT_222_ExpRDREC		FB_PXM_RDRec	
<inputs>			
<outputs>			
<inputs/outputs>			
<public>			
PXM_4_PT_222_ExpRDREC.PubState	5	INT	Operational State
PXM_4_PT_222_ExpRDREC.PubStatus	3	BYTE	Operational Status
PXM_4_PT_222_ExpRDREC.PubCIPStatus	0	BYTE	CIP Status
PXM_4_PT_222_ExpRDREC.PubDPStatus	0	BYTE	Profibus DP Status
PXM_4_PT_222_ExpRDREC.PubExtDPStatus	0	DINT	Extended DP Status
PXM_4_PT_222_ExpRDREC.PubDXCommReport	0	BYTE	Data Exchange Communication Report
PXM_4_PT_222_ExpRDREC.PubDXOpReport	0	BYTE	Data Exchange Operational Report
PXM_4_PT_222_ExpRDREC.PubCIPReplyLength	26	INT	CIP Reply Length
PXM_4_PT_222_ExpRDREC.PubExtractErr	0	INT	Array Extraction Error

Figure 4.12 – RDRec DFB Public Variables

Variable	Data Type	Description

PubState	INT	The internal state of the block.
PubStatus	BYTE	The current status of the block. See table 4.14.
PubCIPStatus	BYTE	The CIP Status of the last received EtherNet/IP response. A value of 0 indicate success.
PubDPStatus	BYTE	The PROFIBUS DP status of the last received DP response. See section 12.1 for the DP Status.
PubExtDPStatus	DINT	The Extended PROFIBUS DP status of the last received DP response. See section 12.2 for the Extended DP Status.
PubDXCommReport	BYTE	M580 Data Exchange Communication Report provided by the DFB's internal DATA_EXCH instruction. See table 4.15 for details.
PubDXOpReport	BYTE	M580 Data Exchange Operation Report provided by the DFB's internal DATA_EXCH instruction. See table 4.16 for details.
PubCIPReplyLength	INT	The CIP data length of the last received EtherNet/IP response.
PubExtractError	INT	M580 Data Extract Error code generated by the DFB's internal EXTRACT instruction.

Table 4.18 – RDRec DFB Public Variables

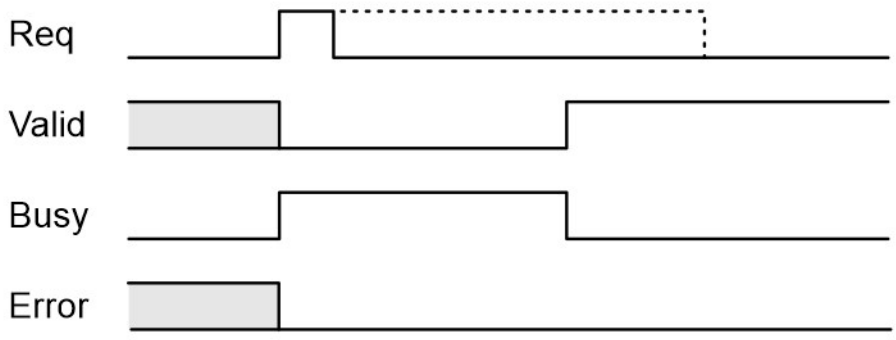


Figure 4.13 – RDRec DFB Timing Diagram - Success

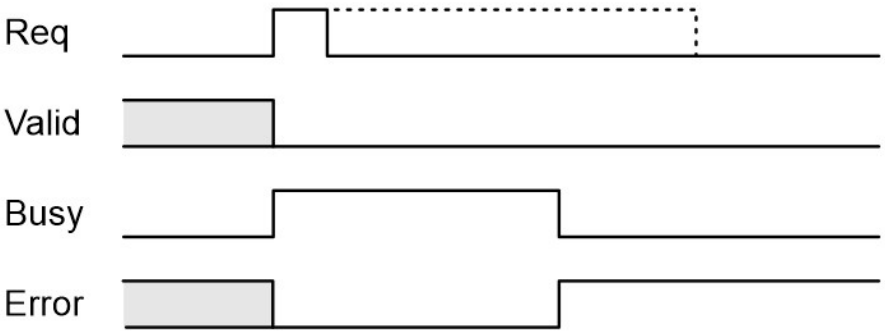


Figure 4.14 – RDRec DFB Timing Diagram - Failure

4.5.3. WRREC DFB

The WRRec (Write Record) DFB provides DPV1 Write functionality using either Class 1 or Class 2 mechanisms. The selection whether Class 1 or Class 2 is used depends on the preceding ID DFB described in the previous section.

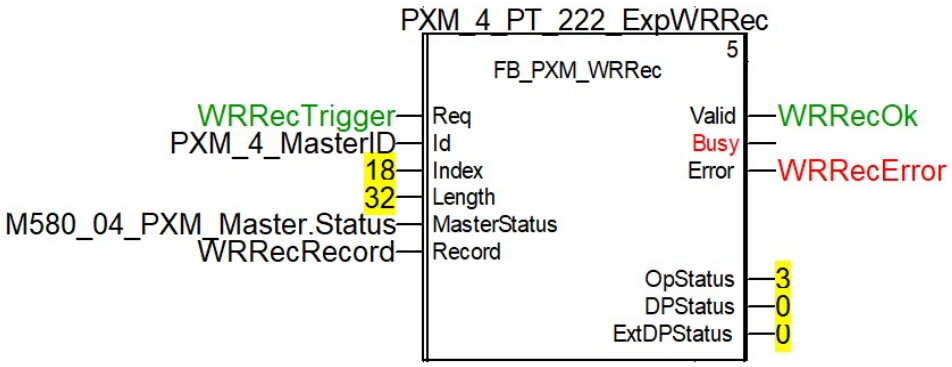


Figure 4.15 – WRRec DFB

The pins for the WRRec DFB are defined as follows:

Pin Type	Pin	Data Type	Description
Input	**Req**	BOOL	Used to Request the write action. A 0-1 transition triggers the write request.
Input	**Id**	T_PXM_Id	The Id structure received from the preceding ID DFB.
Input	**Index**	BYTE	The slave device's Index to be read.
Input	**Length**	BYTE	The length (bytes) of the write request. (1-240 bytes)
Input	**MasterStatus**	T_PXM_MasterStatus	The Master Status information which is produced by the standard PXM master mapping DFB.
Input	**Record**	ANY_ARRAY_BYTE	The data record (byte array) to be written.
Output	**Valid**	BOOL	Write request was successful.
Output	**Busy**	BOOL	Write request is Busy.
Output	**Error**	BOOL	Write Request resulted in an error.
Output	**OpStatus**	BYTE	The Status of the DFB. See table 4.14
Output	**DPStatus**	BYTE	The PROFIBUS DP status of the last received DP response. See section 12.1 for the DP Status.
Output	**ExtDPStatus**	DINT	The Extended PROFIBUS DP status of the last received DP response. See section 12.2 for the Extended DP Status.

Table 4.19 – WRRec DFB Configuration

The WRRec DFB also provides multiple parameters for debugging via its Public variables.

PXM_4_PT_222_ExpWRRec		FB_PXM_WRRec	
<inputs>			
<outputs>			
<inputs/outputs>			
<public>			
PXM_4_PT_222_ExpWRRec.PubState	5	INT	Operational State
PXM_4_PT_222_ExpWRRec.PubStatus	3	BYTE	Operational Status
PXM_4_PT_222_ExpWRRec.PubCIPStatus	0	BYTE	CIP Status
PXM_4_PT_222_ExpWRRec.PubDPStatus	0	BYTE	Profibus DP Status
PXM_4_PT_222_ExpWRRec.PubExtDPStatus	0	DINT	Extended Profibus DP Status
PXM_4_PT_222_ExpWRRec.PubDXCommReport	0	BYTE	Data Exchange Communication Report
PXM_4_PT_222_ExpWRRec.PubDXOpReport	0	BYTE	Data Exchange Operational Report
PXM_4_PT_222_ExpWRRec.PubCIPReplyLength	9	INT	CIP Reply Length
PXM_4_PT_222_ExpWRRec.PubWrittenLength	32	BYTE	Actual Written Length
PXM_4_PT_222_ExpWRRec.PubExtractErr	0	INT	Array Extraction Error

Figure 4.16 – WRRec DFB Public Variables

Variable	Data Type	Description
PubState	INT	The internal state of the block.
PubStatus	BYTE	The current status of the block. See table 4.14.
PubCIPStatus	BYTE	The CIP Status of the last received EtherNet/IP response. A value of 0 indicate success.
PubDPStatus	BYTE	The PROFIBUS DP status of the last received DP response. See section 12.1 for the DP Status.
PubExtDPStatus	DINT	The Extended PROFIBUS DP status of the last received DP response. See section 12.2 for the Extended DP Status.
PubDXCommReport	BYTE	M580 Data Exchange Communication Report provided by the DFB's internal DATA_EXCH instruction. See table 4.15 for details.
PubDXOpReport	BYTE	M580 Data Exchange Operation Report provided by the DFB's internal DATA_EXCH instruction. See table 4.16 for details.
PubCIPReplyLength	INT	The CIP data length of the last received EtherNet/IP response.
PubWrittenLength	BYTE	The actual number of bytes written.
PubExtractError	INT	M580 Data Extract Error code generated by the DFB's internal EXTRACT instruction.

Table 4.20 – WRRec DFB Public Variables

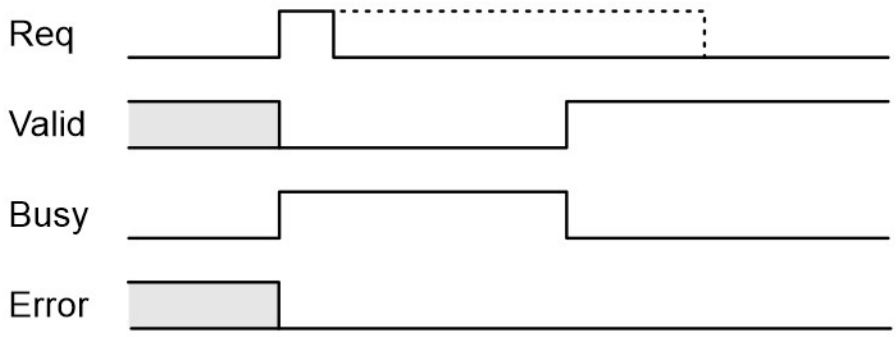


Figure 4.17 – WRRec DFB Timing Diagram - Success

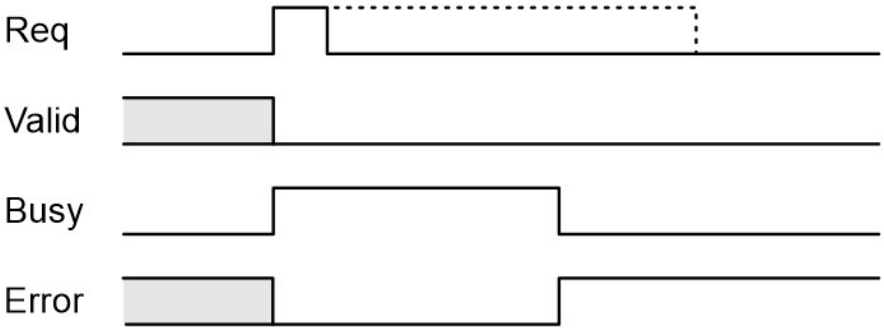


Figure 4.18 – WRRec DFB Timing Diagram - Failure

4.5.4. RDDIAG DFB

The RDDiag (Read Diagnostics) DFB provides the ability to read a slave device's diagnostics. The connection parameters are prescribed by the preceding ID DFB.

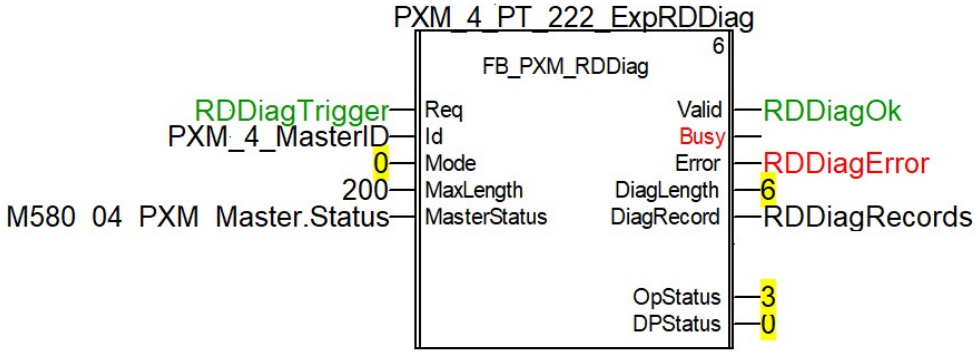


Figure 4.19 – RDDiag DFB

The pins for the RDDiag DFB are defined as follows:

Pin Type	Pin	Data Type	Description
Input	**Req**	BOOL	Used to Request the read diagnostics action. A 0-1 transition triggers the read diagnostic request.
Input	**Id**	T_PXM_Id	The Id structure received from the preceding ID DFB.
Input	**Mode**	BYTE	The mode employed to read the diagnostics. 0 – Read the slave device diagnostics that has been buffered in the PXM. 1 – Read the slave device diagnostics that has been buffered in the PXM and clear the Diagnostics Pending indication. 2 – Force the PXM to send a PROFIBUS Diagnostic Request to the specific slave device and return the diagnostics data received.
Input	**MaxLength**	BYTE	The maximum length (bytes) of the diagnostic read request. (1-244 bytes.)
Input	**MasterStatus**	T_PXM_MasterStatus	The Master Status information which is produced by the standard PXM master mapping DFB.
Output	**Valid**	BOOL	Read diagnostic request was successful.
Output	**Busy**	BOOL	Read diagnostic request is Busy.
Output	**Error**	BOOL	Read diagnostic request resulted in an error.
Output	**DiagLength**	BYTE	The length (bytes) of the retuned record. (0-244 bytes.)
Output	**DiagRecord**	ANY_ARRAY_BYTE	The received diagnostic record as a byte array.
Output	**OpStatus**	BYTE	The Status of the DFB. See table 4.14
Output	**DPStatus**	BYTE	The PROFIBUS DP status of the last received DP response. See section 12.1 for the DP Status.

Table 4.21 – RDDiag DFB Configuration

The RDDiag DFB also provides multiple parameters for debugging via its Public variables.

PXM_4_PT_222_ExpRDDiag		FB_PXM_RDDiag	
<inputs>			
<outputs>			
<inputs/outputs>			
<public>			
PXM_4_PT_222_ExpRDDiag.PubState	5	INT	Operational State
PXM_4_PT_222_ExpRDDiag.PubStatus	3	BYTE	Operational Status
PXM_4_PT_222_ExpRDDiag.PubCIPStatus	0	BYTE	CIP Status
PXM_4_PT_222_ExpRDDiag.PubDPStatus	0	BYTE	Profibus DP Status
PXM_4_PT_222_ExpRDDiag.PubDXCommReport	0	BYTE	Data Exchange Communication Report
PXM_4_PT_222_ExpRDDiag.PubDXOpReport	0	BYTE	Data Exchange Operational Report
PXM_4_PT_222_ExpRDDiag.PubCIPReplyLength	14	INT	CIP Reply Length
PXM_4_PT_222_ExpRDDiag.PubExtractErr	0	INT	Array Extraction Error

Figure 4.20 – RDDiag DFB Public Variables

Variable	Data Type	Description

PubState	INT	The internal state of the block.
PubStatus	BYTE	The current status of the block. See table 4.14.
PubCIPStatus	BYTE	The CIP Status of the last received EtherNet/IP response. A value of 0 indicate success.
PubDPStatus	BYTE	The PROFIBUS DP status of the last received DP response. See section 12.1 for the DP Status.
PubDXCommReport	BYTE	M580 Data Exchange Communication Report provided by the DFB's internal DATA_EXCH instruction. See table 4.15 for details.
PubDXOpReport	BYTE	M580 Data Exchange Operation Report provided by the DFB's internal DATA_EXCH instruction. See table 4.16 for details.
PubCIPReplyLength	INT	The CIP data length of the last received EtherNet/IP response.
PubExtractError	INT	M580 Data Extract Error code generated by the DFB's internal EXTRACT instruction.

Table 4.22 – RDDiag DFB Public Variables

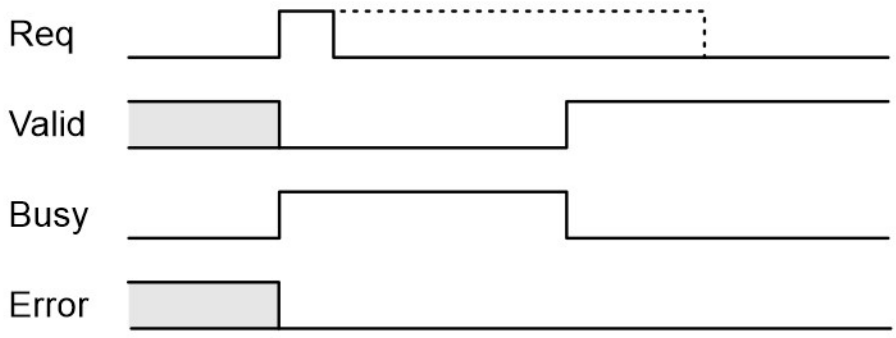


Figure 4.21 – RDDiag DFB Timing Diagram - Success

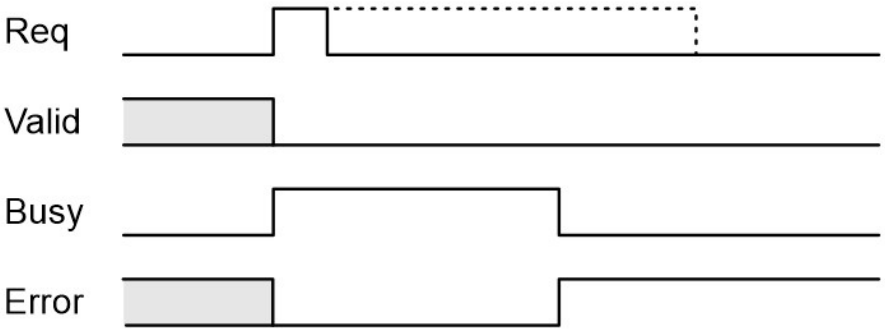


Figure 4.22 – RDDiag DFB Timing Diagram – Failure

4.5.5. RDALARM DFB

The RDAlarm (Read Alarm) DFB provides the ability to extract a slave device's alarm.
The connection parameters are prescribed by the preceding ID DFB.

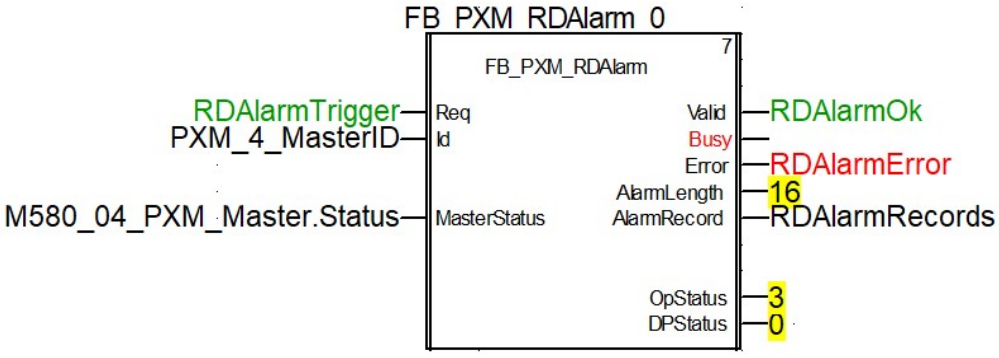


Figure 4.23 – RDAlarmDFB

The pins for the RDAlarm DFB are defined as follows:

Pin Type	Pin	Data Type	Description
Input	**Req**	BOOL	Used to Request the extract Alarm action. A 0-1 transition triggers the request.
Input	**Id**	T_PXM_Id	The Id structure received from the preceding ID DFB.
Input	**MasterStatus**	T_PXM_MasterStatus	The Master Status information which is produced by the standard PXM master mapping DFB.
Output	**Valid**	BOOL	Read alarm request was successful.
Output	**Busy**	BOOL	Read alarm request is Busy.
Output	**Error**	BOOL	Read alarm request resulted in an error.
Output	**AlarmLength**	BYTE	The length (bytes) of the returned record. (0-64 bytes.)
Output	**AlarmRecord**	ANY_ARRAY_BYTE	The received diagnostic record as a byte array.
Output	**OpStatus**	BYTE	The Status of the DFB. See table 4.14
Output	**DPStatus**	BYTE	The PROFIBUS DP status of the last received DP response. See section 12.1 for the DP Status.

Table 4.23 – RDAlarm DFB Configuration

The RDAlarm DFB also provides multiple parameters for debugging via its Public variables.

FB_PXM_RDAlarm_0		FB_PXM_RDAlarm	
<inputs>			
<outputs>			
<inputs/outputs>			
<public>			
FB_PXM_RDAlarm_0.PubState	5	INT	Operational State
FB_PXM_RDAlarm_0.PubStatus	3	BYTE	Operational Status
FB_PXM_RDAlarm_0.PubCIPStatus	0	BYTE	CIP Status
FB_PXM_RDAlarm_0.PubDPStatus	0	BYTE	Profibus DP Status
FB_PXM_RDAlarm_0.PubDXCommReport	0	BYTE	Data Exchange Communication Report
FB_PXM_RDAlarm_0.PubDXOpReport	0	BYTE	Data Exchange Operational Report
FB_PXM_RDAlarm_0.PubCIPReplyLength	26	INT	CIP Reply Length
FB_PXM_RDAlarm_0.PubExtractErr	0	INT	Array Extraction Error

Figure 4.24 – RDAlarm DFB Public Variables

Variable	Data Type	Description
PubState	INT	The internal state of the block.
PubStatus	BYTE	The current status of the block. See table 4.14.
PubCIPStatus	BYTE	The CIP Status of the last received EtherNet/IP response. A value of 0 indicate success.
PubDPStatus	BYTE	The PROFIBUS DP status of the last received DP response. See section 12.1 for the DP Status.
PubDXCommReport	BYTE	M580 Data Exchange Communication Report provided by the DFB's internal DATA_EXCH instruction. See table 4.15 for details.
PubDXOpReport	BYTE	M580 Data Exchange Operation Report provided by the DFB's internal DATA_EXCH instruction. See table 4.16 for details.
PubCIPReplyLength	INT	The CIP data length of the last received EtherNet/IP response.
PubExtractError	INT	M580 Data Extract Error code generated by the DFB's internal EXTRACT instruction.

Table 4.24 – RDAlarm DFB Public Variables

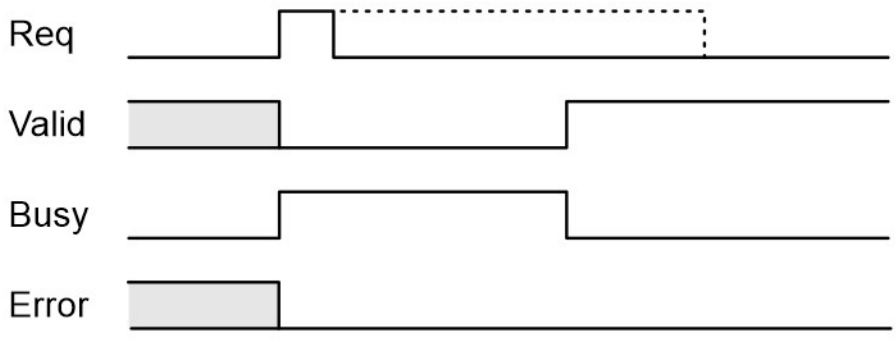


Figure 4.25 – RDAlarm DFB Timing Diagram - Success

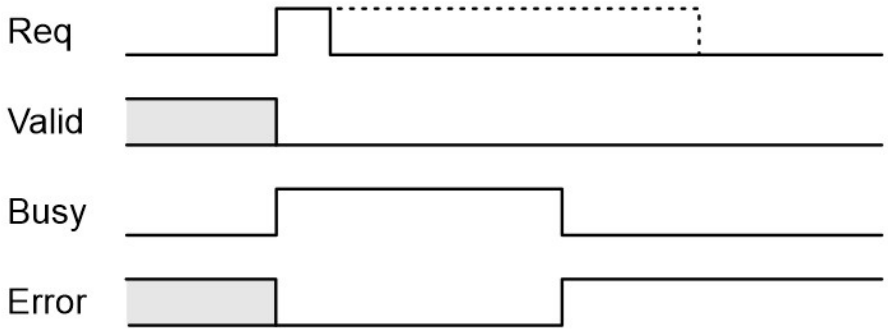


Figure 4.26 – RDAlarm DFB Timing Diagram - Failure

The ***RDAlarm*** instruction should only be executed when required, that is, when an alarm is pending for that device. It must also not be executed more than once per 3 PROFIBUS Cycles. The example logic below illustrates how this can be achieved. Note that the timer's PT is chosen to be at least the sum of the RPI and MAST Task and 3 x PROFIBUS Cycles.

PT > (***RPI*** + ***MASTTask*** + (3 x ***PROFIBUS Cycle)***)

Note that if the intention is to extract alarms from multiple devices then the input to the timer should be an AND function of all the **not**(***RDAlarmBusy)*** flags. This will ensure that no more than one RDAlarm request is executed per PROFIBUS Cycle.

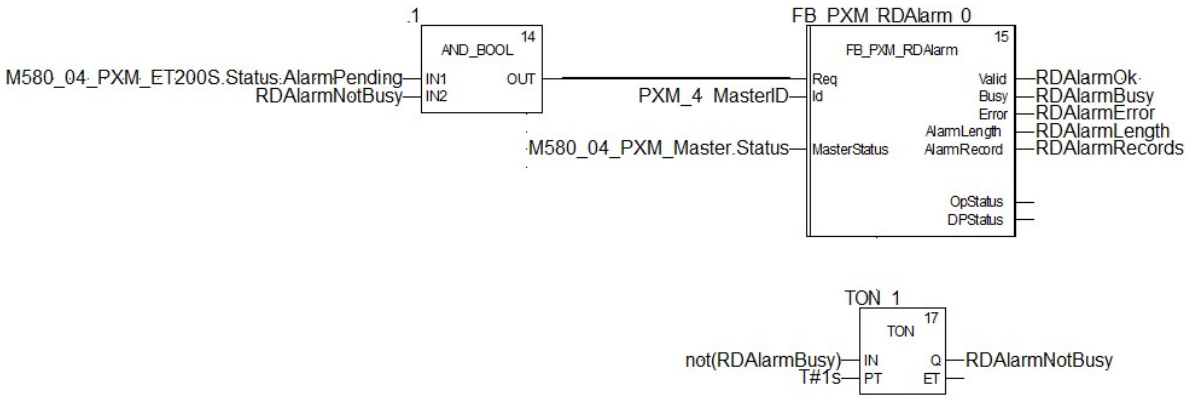


Figure 4.27 – RDAlarm Example

4.5.6. GLOBALCONTROL DFB

The GlobalControl DFB provides a mechanism to send PROFIBUS Global Control commands. The connection parameters are prescribed by the preceding ID DFB.

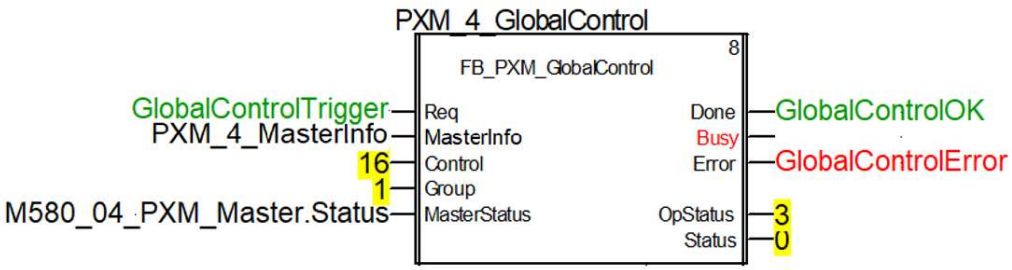


Figure 4.28 – GlobalControl DFB

The pins for the GlobalControl DFB are defined as follows:

Pin Type	Pin	Data Type	Description
Input	**Req**	BOOL	Used to Request the command action. A 0-1 transition triggers the command request.
Input	**MasterInfo**	T_PXM_Info	Master Information defining the path for the explicit message.
Input	**Control**	BYTE	Control Action 0 - Release the Clear mode for the devices 2 - Force the Clear Mode of devices 4 - Freeze 8 - UnFreeze 12 - UnFreeze + 16 - Sync + 32 - UnSync + 48 - UnSync
Input	**Group**	BYTE	Target group/s Each bit of the Group byte represents a Group. This allows the command to be sent to more than one group. Bit 0 – Group 1 Bit 1 – Group 2 Bit 2 – Group 3 Bit 3 – Group 4 Bit 4 – Group 5 Bit 5 – Group 6 Bit 6 – Group 7 Bit 7 – Group 8 For example, sending a command to Groups 2 and 3 would require a Group value of 6. Group Value = 2 (Bit 1) + 4 (Bit 2) = 6 A value of 0 (broadcast) is not supported.
Input	**MasterStatus**	T_PXM_MasterStatus	The Master Status information which is produced by the standard PXM master mapping DFB.
Output	**Done**	BOOL	Command sent successfully.
Output	**Busy**	BOOL	Command request is Busy.

Output	**Error**	BOOL	Command request resulted in an error.
Output	**OpStatus**	BYTE	The Status of the DFB. See table 4.14
Output	**Status**	BYTE	This is the status of the Global Control transmission: 0x00 – Success 0x13 – Failed

Table 4.25 – Global Control DFB Configuration

The GlobalControl DFB also provides multiple parameters for debugging via its Public variables.

PXM_4_GlobalControl		FB_PXM_GlobalControl	
<inputs>			
<outputs>			
<inputs/outputs>			
<public>			
PXM_4_GlobalControl.PubState	5	INT	Operational State
PXM_4_GlobalControl.PubStatus	3	BYTE	Operational Status
PXM_4_GlobalControl.PubCIPStatus	0	BYTE	CIP Status
PXM_4_GlobalControl.PubDXCommReport	0	BYTE	Data Exchange Communication Report
PXM_4_GlobalControl.PubDXOpReport	0	BYTE	Data Exchange Operational Report
PXM_4_GlobalControl.PubCIPReplyLength	5	INT	CIP Reply Length

Figure 4.29 – Global Control DFB Public Variables

Variable	Data Type	Description
PubState	INT	The internal state of the block.
PubStatus	BYTE	The current status of the block. See table 4.14.
PubCIPStatus	BYTE	The CIP Status of the last received EtherNet/IP response. A value of 0 indicate success.
PubDXCommReport	BYTE	M580 Data Exchange Communication Report provided by the DFB's internal DATA_EXCH instruction. See table 4.15 for details.
PubDXOpReport	BYTE	M580 Data Exchange Operation Report provided by the DFB's internal DATA_EXCH instruction. See table 4.16 for details.
PubCIPReplyLength	INT	The CIP data length of the last received EtherNet/IP response.

Table 4.26 – Global Control DFB Public Variables

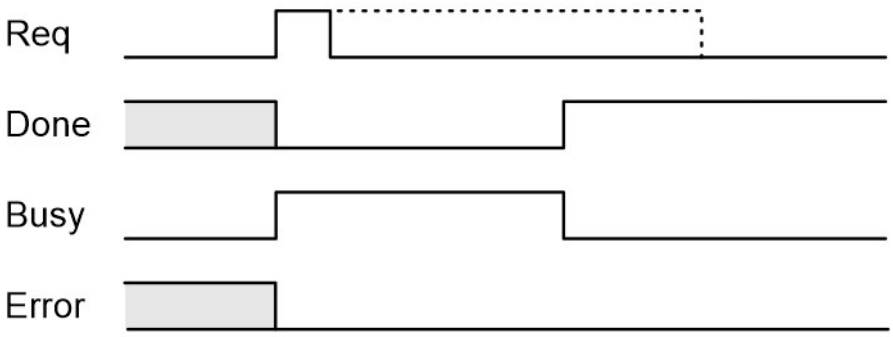


Figure 4.30 – GlobalControl DFB Timing Diagram - Success

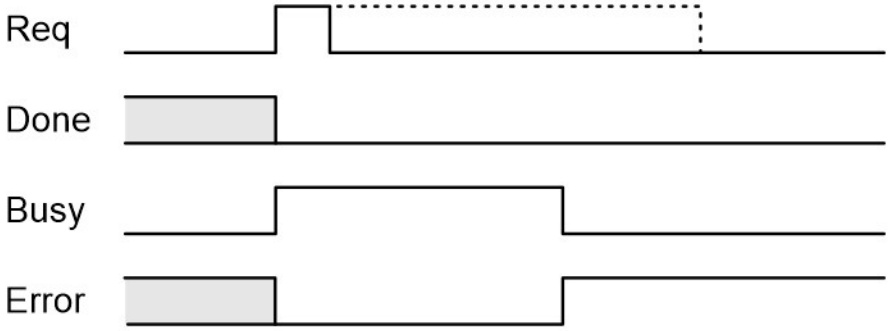


Figure 4.31 – GlobalControl DFB Timing Diagram - Failure

4.6. EXPLICIT MESSAGING UTILITY

The ProSoft Configurator for Modicon provides a utility to initiate explicit messages to the PROFIBUS devices via the PXM. The messaging options include the following:

- DPV1 Class 1 Read
- DPV1 Class 1 Write
- DPV1 Class 2 Read
- DPV1 Class 2 Write
- Read Diagnostics
- Read Alarms

To open this utility, right-click on a PROFIBUS device and select the ***Explicit Messaging*** option.

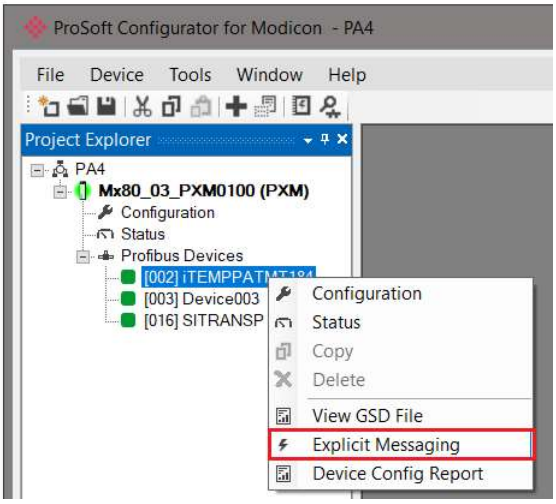


Figure 4.32 – Explicit Messaging Option

Use the ***Action*** combo-box to select the type of explicit message. Depending on the type selected, various other parameter controls will become available. Once the correct parameters have been entered select the ***Execute*** button to initiate the explicit exchange.

For Class 2 messages, if a class 2 connection has not already been established, then a Class 2 Initialization message will first be sent. The class 2 connection will then remain open until either the station address is changed, the manual Abort button is selected, or the utility is closed.

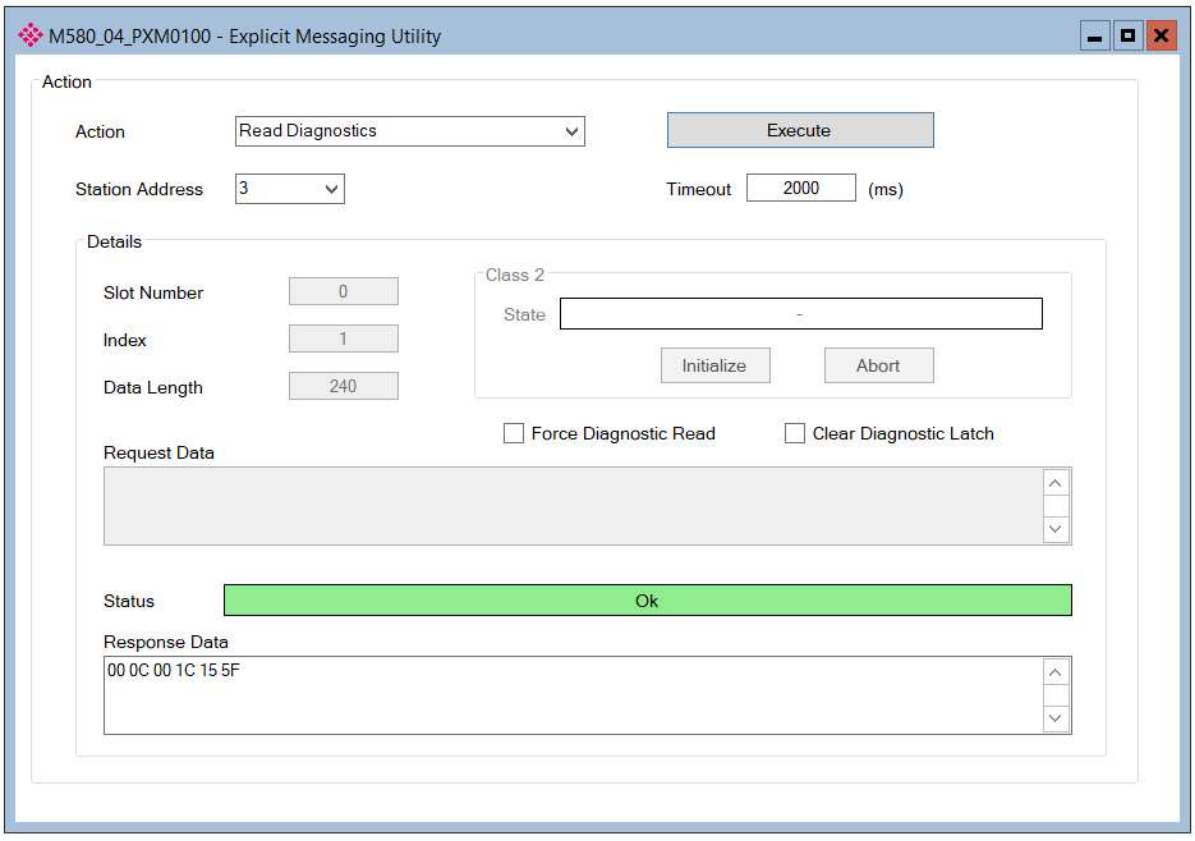


Figure 4.33 – Explicit Messaging Utility

The available parameter controls are as follows:

Parameter Control	Description
Action	The Action to be executed, select one of the following: • DPV1 Class 1 Read • DPV1 Class 1 Write • DPV1 Class 2 Read • DPV1 Class 2 Write • Read Diagnostics • Read Alarms
Execute	Command - Triggers the message.
Station Address	The target Station Address of the message.
Timeout	The CIP message timeout (milliseconds).
Slot Number	The slave slot number for the message. DPV1 Read and Write only.
Index	The slave index number for the message. DPV1 Read and Write only.
Data Length	The requested data length. DPV1 Read (Class 1 and Class 2) only. Note: For DPV1 Writes, the length is provided by the actual Request Data.
Class 2 - State	The current DPV1 Class 2 status: • Connected, displays the Connection Reference Id • Disconnected • Failed – Displays error message DPV1 Class 2 messages only.
Class 2 - Initialize	Command – Initializing a Class 2 connection. DPV1 Class 2 messages only.
Class 2 - Abort	Command – Aborting a Class 2 connection. DPV1 Class 2 messages only.
Force Diagnostic Read	Option to force a PROFIBUS DPV0 refresh of the slave device's diagnostics: • Unchecked – Retrieve cached diagnostic info. • Checked – Force fresh DPV0 diagnostic read on slave device. Read Diagnostics only. Note: When a Force Diagnostic Read is selected, it would inherently clear the Diagnostics latch.
Clear Diagnostic Latch	Option to clear a diagnostic flag after reading the diagnostics: • Unchecked – Flag unchanged. • Checked – Clear Flag

	Read Diagnostics only. Note: This option will be disabled when the Force Diagnostic Read option is selected, because the latter option would inherently clear the Diagnostics latch.
Request Data	Data payload to be used in request. (HEX array format) DPV1 Write messages only.
Status	Status of previous action. • Ok – Success • Failed – Message failed, Error information displayed.
Response Data	Data payload of response. (HEX array format) All messages except DPV1 Write messages.

Table 4.27 – Explicit Messaging Utility controls

4.7. GLOBAL CONTROL UTILITY

The ProSoft Configurator for Modicon provides a utility to send Global Controls to Groups of PROFIBUS devices via the PXM. To open this utility, right-click on the ***PROFIBUS Devices*** tree item and select the ***Global Control*** option.

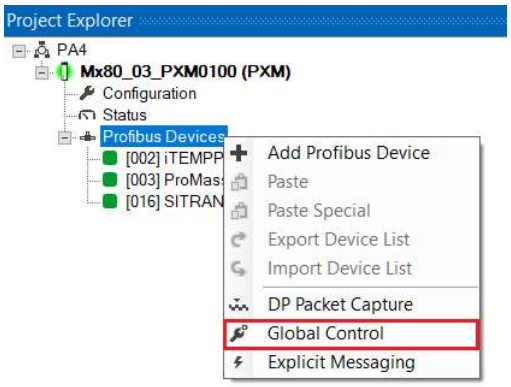


Figure 4.34 – Opening Global Control Utility

At least one PROFIBUS group must be selected.
The Global Control command options are categorised as follows:

Clear:

Select either ***Release Clear*** or ***Force Clear***

Sync:

Select ***Unchanged***, ***Sync***. or ***UnSync***.

Freeze:

Select ***Unchanged***, ***Freeze*** or ***UnFreeze***.

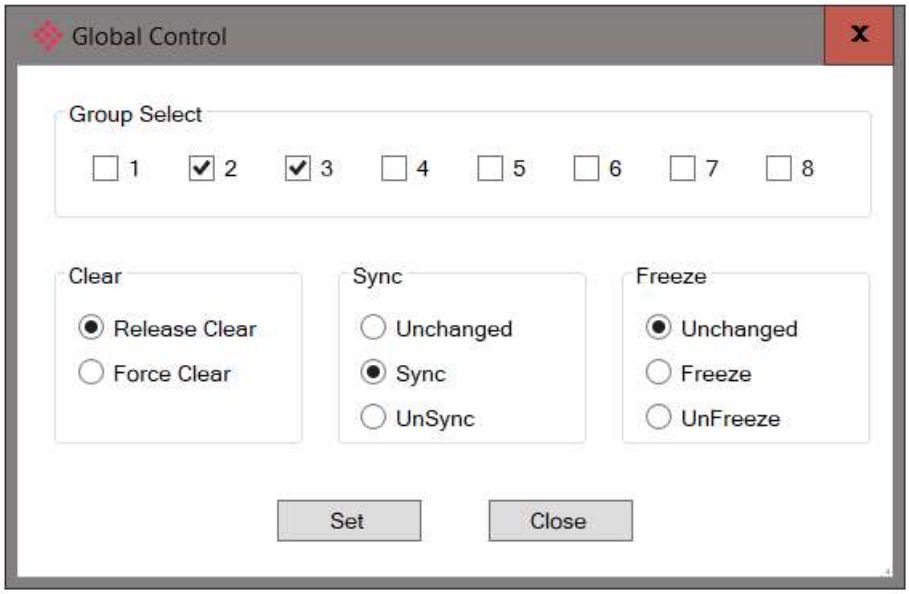


Figure 4.35 – Global Control Utility

4.8. DPV1 COMMUNICATION

4.8.1. CLASS 1 MESSAGING (MS1)

DPV1 Class 1 messaging will only be achievable if the slave device is in data exchange mode (i.e. the device is configured and exchanging cyclic data with the PXM). Only the DP Master exchanging data with the slave device can read and write parameters using DPV1 MS1. Below are the EtherNet/IP CIP message parameters as well as the request and response data structures.

4.8.1.1. READ

MESSAGE:

Parameter	Description
Service Code	0x52 (Hex)
Class	0x64 (Hex)
Instance	1
Attribute	N/A
Request Data Length	8

Table 4.28 – DPV1 Class 1 Read Message

REQUEST DATA:

Parameter	Data Type	Description
Timeout	Long	The amount of time (in milliseconds) the PXM waits for a DPV1 response before timing out and responding to the EtherNet/IP request with a Timeout Status.
Slave Address	Byte	The station number of the PROFIBUS device.
Slot Number	Byte	The DPV1 Slot number which must be read.
Index	Byte	The DPV1 Index number which must be read.
Data Length	Byte	The maximum number of bytes that must be read.

Table 4.29 – DPV1 Class 1 Read Request

RESPONSE DATA:

Parameter	Data Type	Description
DP Status	Byte	This is the status of the DPV1 data exchange. See section 12.1 for the DP Status.
Extended Status	Byte[3]	This is the extended status of the DPV1 data exchange. See appendix for the definitions of the returned extended status. See section 12.2 for the Extended DP Status.
Data Length	Byte	The length of the data returned.
Reserved	Byte	-
Data	ANY_ARRAY_BYTE	The data from the DPV1 Read request. The number of bytes will be equal to the Data Length in the response.

Table 4.30 – DPV1 Class 1 Read Response

4.8.1.2. WRITE

MESSAGE:

Parameter	Description
Service Code	0x53 (Hex)
Class	0x64 (Hex)
Instance	1

Attribute	N/A
Request Data Length	8 + Length of Data Payload

Table 4.31 – DPV1 Class 1 Write Message

REQUEST DATA:

Parameter	Data Type	Description
Timeout	Long	The amount of time (in milliseconds) the PXM waits for a DPV1 response before timing out and responding to the EtherNet/IP request with a Timeout Status.
Slave Address	Byte	The station number of the PROFIBUS device.
Slot Number	Byte	The DPV1 Slot number for the write request.
Index	Byte	The DPV1 Index number for the write request.
Data Length	Byte	The number of bytes that must be written.
Data	ANY_ARRAY_BYTE	The data that will be written to the specific address. The number of bytes will be equal to the Data Length in the request.

Table 4.32 – DPV1 Class 1 Write Request

RESPONSE DATA:

Parameter	Data Type	Description
DP Status	Byte	This is the status of the DPV1 data exchange. See appendix for the definitions of the returned status. See section 12.1 for the DP Status.
Extended Status	Byte[3]	This is the extended status of the DPV1 data exchange. See section 12.2 for the Extended DP Status.
Data Length	Byte	The length of the data that was written.

Table 4.33 – DPV1 Class 1 Write Response

4.8.2. CLASS 2 MESSAGING (MS2)

DPV1 Class 2 messaging is possible from several DP masters simultaneously, but the connection must be established explicitly by each DP Master. Below are the EtherNet/IP CIP message parameters as well as the request and response data structures.

4.8.2.1. INITIALIZE (ESTABLISH CONNECTION)

MESSAGE:

Parameter	Description
Service Code	0x54 (Hex)
Class	0x64 (Hex)
Instance	1
Attribute	N/A
Request Data Length	20 + (2 + Source Net Address Length + Source MAC Address Length) + (2 + Destination Net Address Length + Destination MAC Address Length)

Table 4.34 – DPV1 Class 2 Initialize Message

REQUEST DATA:

Parameter	Data Type	Description
Timeout	Long	The amount of time (in milliseconds) the PXM waits for a DPV1 response before timing out and responding to the EtherNet/IP request with a Timeout Status.
Slave Address	Byte	The station number of the PROFIBUS device.
Reserved	Byte[3]	-
Send Timeout	Short	Refer to the *PROFIBUS – DP Extensions to EN 50170 (DPV1)* for information regarding these parameters.
Features Supported	Short	
Profile Features Supported	Short	
Profile Ident Number	Short	
Source Type	Byte	
Source Address Length	Byte	
Destination Type	Byte	
Destination Address Length	Byte	

Source API	Byte	
Source SCL	Byte	
Source Net Address	ANY_ARRAY_BYTE	
Source MAC Address	ANY_ARRAY_BYTE	
Destination API	Byte	
Destination SCL	Byte	
Destination Net Address	ANY_ARRAY_BYTE	
Destination MAC Address	ANY_ARRAY_BYTE	

Table 4.35 – DPV1 Class 2 Initialize Request

RESPONSE DATA:

Parameter	Data Type	Description
DP Status	Byte	This is the status of the DPV1 data exchange. See appendix for the definitions of the returned status. See section 12.1 for the DP Status.
Extended Status	Byte[3]	This is the extended status of the DPV1 data exchange. See appendix for the definitions of the returned extended status. See section 12.2 for the Extended DP Status.
Features Supported	Short	Refer to the *PROFIBUS – DP Extensions to EN 50170 (DPV1)* for information regarding these parameters.
Profile Features Supported	Short	
Profile Ident Number	Short	
Connection Reference	Byte	The connection reference is a reference number that must be used for further communication on this connection (e.g. Read, Write, or Abort).

Table 4.36 – DPV1 Class 2 Initialize Response

4.8.2.2. ABORT

MESSAGE:

Parameter	Description
Service Code	0x55 (Hex)
Class	0x64 (Hex)
Instance	1

Attribute	N/A
Request Data Length	7

Table 4.37 – DPV1 Class 2 Abort Message

REQUEST DATA:

Parameter	Data Type	Description
Reserved	Long	-
Connection Reference	Byte	Connection Reference Received from the DPV1 Class 2 Initialize Response.
Subnet	Byte	Refer to the *PROFIBUS – DP Extensions to EN 50170 (DPV1)* for information regarding these parameters.
Instance Reason Code	Byte	

Table 4.38 – DPV1 Class 2 Abort Request

RESPONSE DATA:

Parameter	Data Type	Description
None	-	-

Table 4.39 – DPV1 Class 2 Abort Response

4.8.2.3. READ

MESSAGE:

Parameter	Description
Service Code	0x56 (Hex)
Class	0x64 (Hex)
Instance	1
Attribute	N/A
Request Data Length	8

Table 4.40 – DPV1 Class 2 Read Message

REQUEST DATA:

Parameter	Data Type	Description
Timeout	Long	The amount of time (in milliseconds) the PXM waits for a DPV1 response before timing out and responding to the EtherNet/IP request with a Timeout Status.
Connection Reference	Byte	Connection Reference Received from the DPV1 Class 2 Initialize Response.
Slot Number	Byte	The DPV1 Slot number which must be read.
Index	Byte	The DPV1 Index number which must be read.
Data Length	Byte	The maximum number of bytes that must be read.

Table 4.41 – DPV1 Class 2 Read Request

RESPONSE DATA:

Parameter	Data Type	Description
DP Status	Byte	This is the status of the DPV1 data exchange. See appendix for the definitions of the returned status. See section 12.1 for the DP Status.
Extended Status	Byte[3]	This is the extended status of the DPV1 data exchange. See appendix for the definitions of the returned extended status. See section 12.2 for the Extended DP Status.
Data Length	Byte	The length of the data returned.
Reserved	Byte	-
Data	ANY_ARRAY_BYTE	The data from the DPV1 Read request. The number of bytes will be equal to the Data Length in the response.

Table 4.42 – DPV1 Class 2 Read Response

4.8.2.4. WRITE

MESSAGE:

Parameter	Description
Service Code	0x57 (Hex)
Class	0x64 (Hex)
Instance	1
Attribute	N/A

Request Data Length	8 + Length of Data Payload

Table 4.43 – DPV1 Class 2 Write Message

REQUEST DATA:

Parameter	Data Type	Description
Timeout	Long	The amount of time (in milliseconds) the PXM waits for a DPV1 response before timing out and responding to the EtherNet/IP request with a Timeout Status.
Connection Reference	Byte	Connection Reference Received from the DPV1 Class 2 Initialize Response.
Slot Number	Byte	The DPV1 Slot number for the write request.
Index	Byte	The DPV1 Index number for the write request.
Data Length	Byte	The number of bytes that must be written.
Data	ANY_ARRAY_BYTE	The data that will be written to the specific address. The number of bytes will be equal to the Data Length in the request.

Table 4.44 – DPV1 Class 2 Write Request

RESPONSE DATA:

Parameter	Data Type	Description
DP Status	Byte	This is the status of the DPV1 data exchange. See appendix for the definitions of the returned status. See section 12.1 for the DP Status.
Extended Status	Byte[3]	This is the extended status of the DPV1 data exchange. See appendix for the definitions of the returned extended status. See section 12.2 for the Extended DP Status.
Data Length	Byte	The length of the data that was written.

Table 4.45 – DPV1 Class 2 Write Response

4.9. DIAGNOSTICS

The PXM will flag to the user when new diagnostics have been received. When new diagnostics have been flagged by the PXM the user can extract the diagnostics message from the PXM by using EtherNet/IP unconnected messaging (UCMM) or Class 3 connected messaging.

4.9.1. NOTIFICATION

The PXM will notify the user of pending diagnostics in four areas.

4.9.1.1. CONTROL EXPERT PXM DDT

In the Status part of the PXM DDT (see *PXM DDT* section) there is a tag DiagPending. This is an array of Boolean tags each of which represents a node on the network. Below is a description of the tag.

Tag	Description
DiagPending	Indicates the nodes that have diagnostics pending on the local PROFIBUS network. Each bit represents a node. When the specific bit is set '1' then the device has diagnostics pending that must be unloaded and when the bit is off '0' the device does not have any diagnostics pending. Bit 0 – Node 0 has diagnostics pending Bit 1 – Node 1 has diagnostics pending ……. Bit 126 – Node 126 has diagnostics pending

Table 4.46 – PXM DDT Diagnostics Pending Indications

4.9.1.2. CONTROL EXPERT FIELD DEVICE DDT

In the Status part of the Device DDT (see *Device DDT* section) there is a tag DiagnosticsPending. Below is a description of the tag.

Tag	Description
DiagnosticsPending	Indicates the device has diagnostics pending on the local PROFIBUS network. When the specific bit is set '1' then the device has diagnostics pending that must be unloaded and when the bit is off '0' the device does not have any diagnostics pending. 0 – The node has no diagnostics pending 1 – The node has diagnostics pending

Table 4.47 – PXM DDT Diagnostics Pending Indications

4.9.1.3. PROSOFT CONFIGURATOR FOR MODICON

If the user is online with the PXM in the ProSoft Configurator for Modicon, the user can open the status of the Device by right-clicking on the specific slave device and selecting *Status*. The General status will then indicate if there are diagnostics pending as shown below. See the *Diagnostics* section for more details.

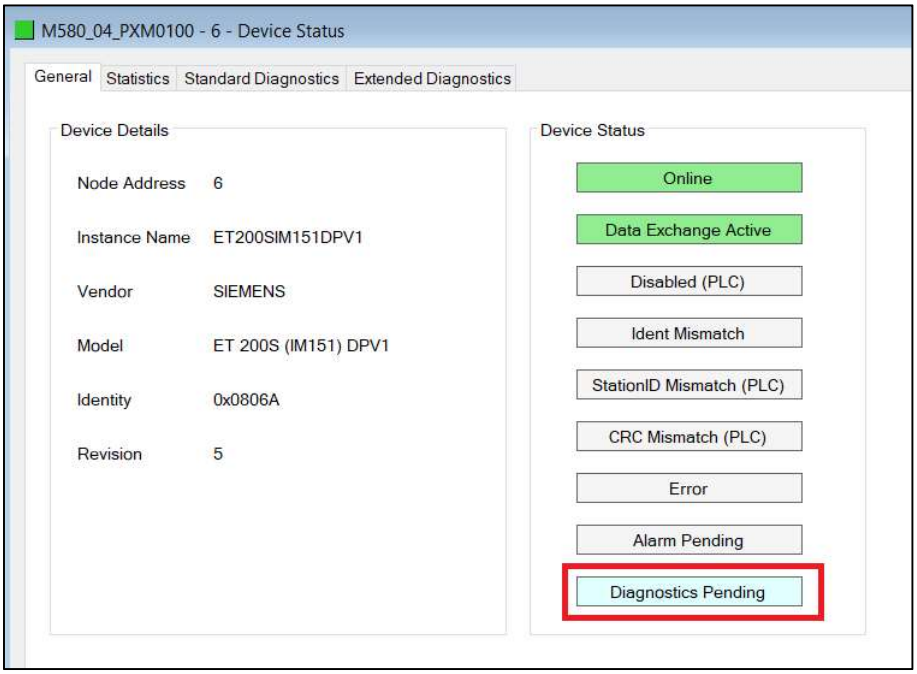


Figure 4.36 – Device Diagnostics Pending in ProSoft Configurator for Modicon

4.9.1.4. WEBSERVER

The user can also see an indication that the device has pending diagnostics using the module Webserver.

The user must have enabled HTTP in the PXM security settings for the webserver to work.

Once in the webserver, the user can click on the *Application* button and select *PROFIBUS DP Devices* in the Application Type. Next the user must select the *Device Index* to retrieve the slave device information (as shown below). See the *Diagnostics* section for more details regarding the PXM Webserver.

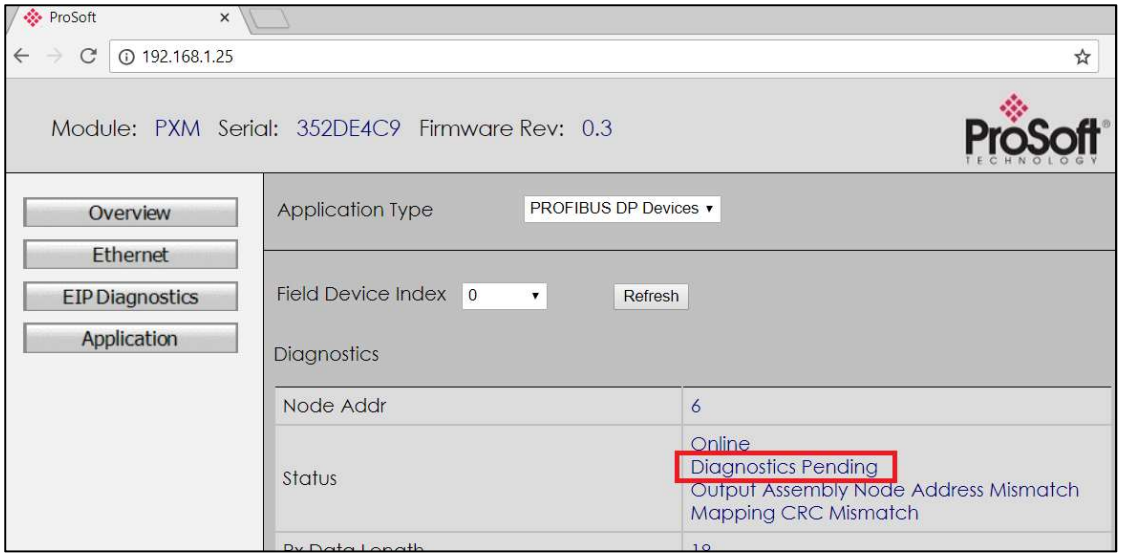


Figure 4.37 – Field device Diagnostics Pending in PXM Webserver

4.9.2. EXTRACTION

The user can extract diagnostics by using either the slave device node address or its node ID (which is in which position index it was configured). For example, if the user has three devices with node address 2,3, and 8, then the Node ID will be 0, 1, and 2.

The user can also decide how the diagnostics data must be extracted. This is changed by updating the mode in the Diagnostics Request message. There are one of three modes that can be selected:

Mode	Description
0	Read the slave device diagnostics that has been buffered in the PXM.
1	Read the slave device diagnostics that has been buffered in the PXM and clear the Diagnostics Pending indication.
2	Force the PXM to send a PROFIBUS Diagnostic Request to the specific slave device and return the diagnostics data received.

Table 4.48 – Diagnostics Extract Message

Below are the EtherNet/IP CIP message parameters as well as the request and response data structures.

MESSAGE:

Parameter	Description
Service Code	0x4B (Hex)
Class	0x65 (Hex)
Instance	1
Attribute	N/A
Request Data Length	8

Table 4.49 – Diagnostics Extract Message

REQUEST DATA:

Parameter	Data Type	Description
Timeout	Long	The amount of time (in milliseconds) the PXM waits for a DPV1 response before timing out and responding to the EtherNet/IP request with a Timeout Status.
Slave Address	Byte	The station number of the PROFIBUS device.
Node ID	Byte	The Node ID is position of the slave device in the configuration.
Reserved	Byte	-
Mode	Byte	0 – Read the slave device diagnostics that has been buffered in the PXM. 1 – Read the slave device diagnostics that has been buffered in the PXM and clear the Diagnostics Pending indication. 2 – Force the PXM to send a PROFIBUS Diagnostic Request to the specific slave device and return the diagnostics data received.

Table 4.50 – Diagnostics Extract Request

RESPONSE DATA:

Parameter	Data Type	Description
Status	Byte	This is the status of the DPV1 data exchange. See appendix for the definitions of the returned status.
Reserved	Byte	-
Diagnostics data length	Byte	The number of diagnostic bytes that have been returned.
Reserved	Byte	-

Diagnostics Data	ANY_ARRAY_BYTE	Refer to the *PROFIBUS Specification EN 50170* for information regarding the diagnostics.

Table 4.51 – Diagnostics Extract Response

4.10. GLOBAL CONTROL

Global control commands are multi-cast PROFIBUS commands which can be sent to a group of slave devices. Below are the EtherNet/IP CIP message parameters as well as the request and response data structures.

MESSAGE:

Parameter	Description
Service Code	0x4E (Hex)
Class	0x65 (Hex)
Instance	1
Attribute	N/A
Request Data Length	6

Table 4.52 – Global Control Message

REQUEST DATA:

Parameter	Data Type	Description
Timeout	Long	The amount of time (in milliseconds) the PXM waits for a response before timing out and responding to the EtherNet/IP request with a Timeout Status.
Control	Byte	The Global Control action: 0 - Release the Clear mode for the devices 2 - Force the Clear Mode of devices 4 - Freeze 8 - UnFreeze 12 - UnFreeze + 16 - Sync + 32 – UnSync + 48 - UnSync
Group	Byte	Target group/s Each bit of the Group byte represents a Group. This allows the command to be sent to more than one group. Bit 0 – Group 1 Bit 1 – Group 2 Bit 2 – Group 3

		Bit 3 – Group 4 Bit 4 – Group 5 Bit 5 – Group 6 Bit 6 – Group 7 Bit 7 – Group 8 For example, sending a command to Groups 2 and 3 would require a Group value of 6. Group Value = 2 (Bit 1) + 4 (Bit 2) = 6 A value of 0 (broadcast) is not supported.

Table 4.53 – Global Control Request

RESPONSE DATA:

Parameter	Data Type	Description
Status	Byte	This is the status of the Global Control transmission: 0x00 – Success 0x13 – Failed

Table 4.54 – Global Control Response

4.11. ALARMING

The PXM will flag to the user when a new alarm has been received. When a new alarm has been flagged by the PXM the user can extract the alarm from the PXM by using EtherNet/IP unconnected messaging (UCMM) or Class 3 connected messaging.

The Alarm extraction process requires one or more Profibus Cycles to exchange the alarm acknowledgment with the slave device. Attempting to execute a subsequent ***ReadAlarm*** instruction within the same Profibus Cycle, or before this acknowledgment has completed will result in an error.

If there is more than one alarm pending, then after extract the alarm the alarm pending will be set again to indicate there are more alarms to unload.

4.11.1. NOTIFICATION

The PXM will notify the user of a pending alarm in four areas.

4.11.1.1. CONTROL EXPERT PXM DDT

In the Status part of the PXM DDT (see *PXM DDT* section) there is a tag ***AlarmPending***. This is an array of Boolean tags each of which represents a node on the network. Below is a description of the tag.

Tag	Description
AlarmPending	Indicates the nodes that have an alarm pending on the local PROFIBUS network. Each bit represents a node. When the specific bit is set '1' then the device has an alarm pending that must be unloaded and when the bit is off '0' the device does not have an alarm pending. Bit 0 – Node 0 has an alarm pending Bit 1 – Node 1 has an alarm pending ……. Bit 126 – Node 126 has an alarm pending

Table 4.55 – PXM DDT Alarm Pending Indications

4.11.1.2. CONTROL EXPERT FIELD DEVICE DDT

In the Status part of the Device DDT (see *Device DDT* section) there is a tag AlarmPending. Below is a description of the tag.

Tag	Description
AlarmPending	Indicates the device has an alarm pending on the local PROFIBUS network. When the specific bit is set '1' then the device has an alarm pending that must be unloaded and when the bit is off '0' the device does not have an alarm pending. 0 – The node has no alarm pending 1 – The node has an alarm pending

Table 4.56 – PXM DDT Alarm Pending Indications

4.11.1.3. PROSOFT CONFIGURATOR FOR MODICON

If the user is online with the PXM in the ProSoft Configurator for Modicon, the user can open the status of the Device by right-clicking on the specific slave device and selecting *Status*. The General status will then indicate if there is an alarm pending as shown below. See the *Diagnostics* section for more details.

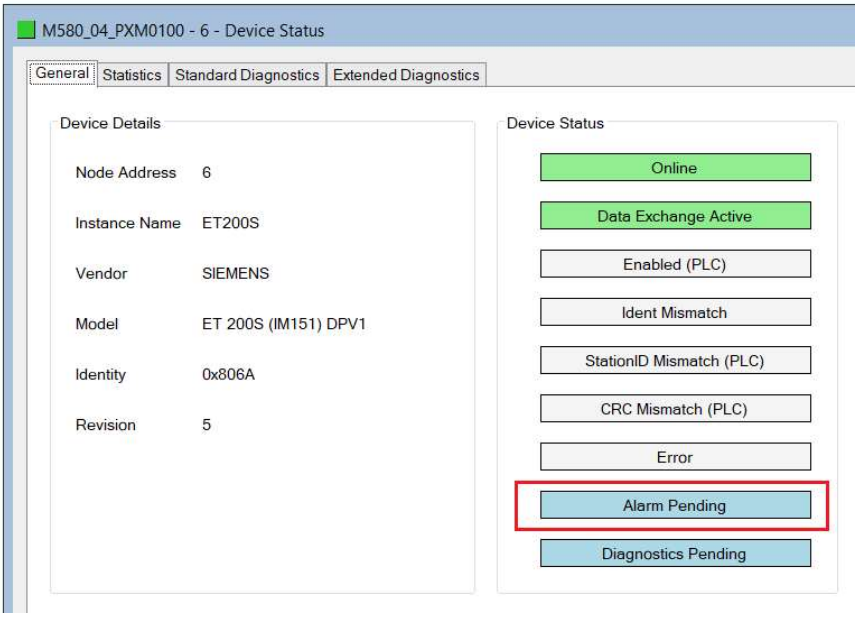


Figure 4.38 – Device Alarm Pending in ProSoft Configurator for Modicon

4.11.1.4. WEBSERVER

The user can also see an indication that the device has a pending alarm using the module Webserver.

NOTICE
UNEXPECTED BEHAVIOUR The user must have enabled HTTP in the PXM security settings for the webserver to work. **Failure to follow these instructions may result in an unexpected behaviour.**

Once in the webserver, the user can click on the *Application* button and select *PROFIBUS DP Devices* in the Application Type. Next the user must select the *Device Index* to retrieve the slave device information (as shown below). See the *Diagnostics* section for more details regarding the PXM Webserver.

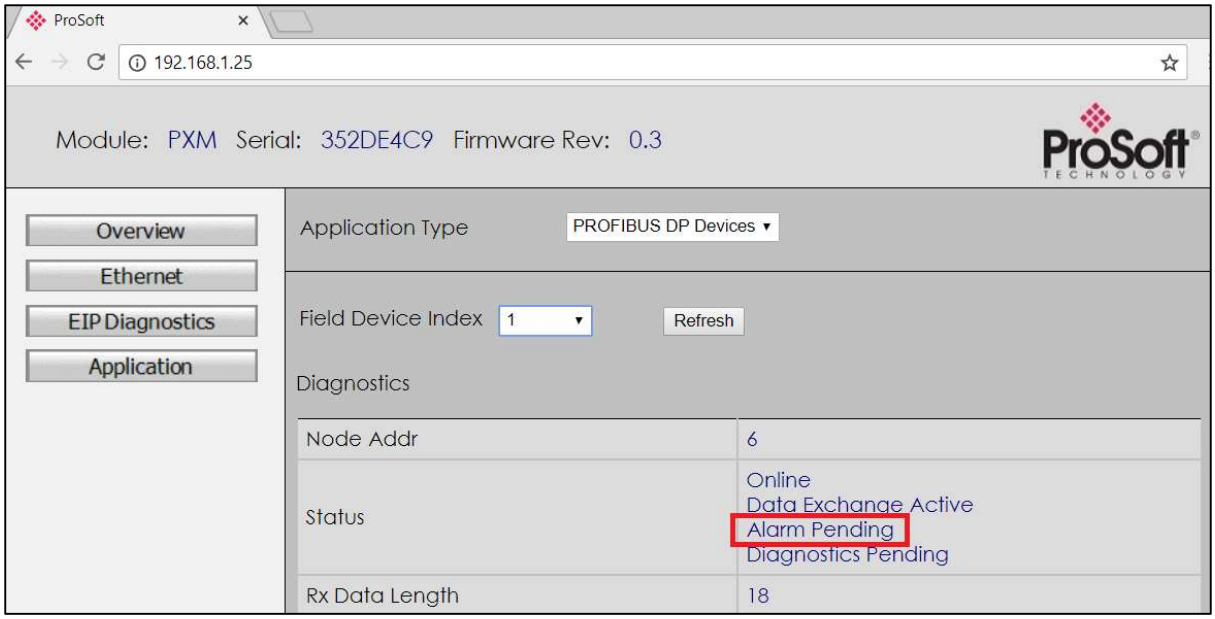


Figure 4.39 – Field device Alarm Pending in PXM Webserver

4.11.2. EXTRACTION

The user can extract an alarm by using the slave device node address. Below are the EtherNet/IP CIP message parameters as well as the request and response data structures.

MESSAGE:

Parameter	Description
Service Code	0x58 (Hex)
Class	0x64 (Hex)
Instance	1
Attribute	N/A
Request Data Length	5

Table 4.57 – Alarm Extract Message

REQUEST DATA:

Parameter	Data Type	Description
Timeout	Long	The amount of time (in milliseconds) the PXM waits for a DPV1 response before timing out and responding to the EtherNet/IP request with a Timeout Status.
Slave Address	Byte	The station number of the PROFIBUS device.

Table 4.58 – Alarm Extract Request

RESPONSE DATA:

Parameter	Data Type	Description
Status	Byte	This is the status of the DPV1 data exchange. See appendix for the definitions of the returned status.
Extended Status	Byte[3]	This is the extended status of the DPV1 data exchange. See appendix for the definitions of the returned extended status.
Alarm data length	Byte	The amount of alarm bytes that have been returned.
Alarm data	ANY_ARRAY_BYTE	Refer to the *PROFIBUS Specification EN 50170* for information regarding the diagnostics.

Table 4.59 – Alarm Extract Response

4.12. FAST DEVICE REPLACEMENT (FDR)

The PXM can retrieve the module configuration from the M580 head module. This will be initiated when the PXM is starting up and if the M580 head module has a copy of the PXM module configuration (the configuration downloaded from the ProSoft Configurator for Modicon). The user can view the status of the last configuration retrieval or storage from the PXM General Status in the ProSoft Configurator for Modicon (see the *Diagnostics* section).

The user can download the current configuration to the M580 head module by right-clicking on the PXM in the ProSoft Configurator for Modicon (when online with the module) and selecting Upload to FDR as shown below.

The user can stop the PXM from using the configuration in both the PXM non-volatile memory as well as the M580 head module by setting DIP 1. See the Installation section.

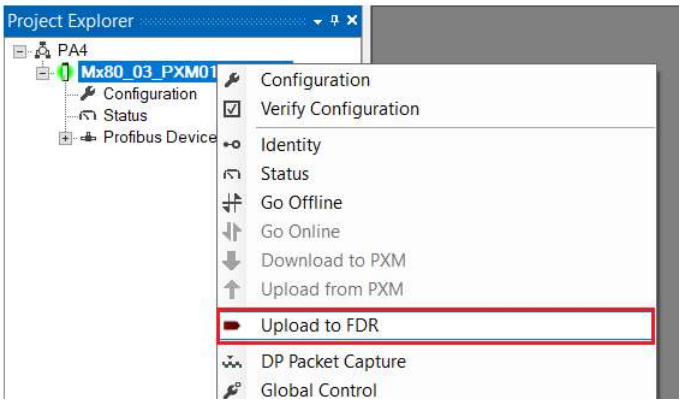


Figure 4.40 – Upload PXM Configuration to FDR

Once the module reboots it will retrieve the last configuration uploaded to the FDR and overwrite the current configuration in the PXM non-volatile memory.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

Ensure that the configuration in the M580 head module and the configuration in the ProSoft Configurator for Modicon (downloaded to the module) are the same.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

4.13. FIRMWARE UPGRADING

The PXM allows the user to upgrade the module firmware in the field. If the firmware needs to be updated the user will need to use the ProSoft Configurator for Modicon to update it.

Before the PXM's firmware can be upgraded, the PLC connected to it must be in STOP mode. In the case of an HSBY application, both the A and B PLCs must be in STOP mode.

The preferred connection method when upgrading the PXM firmware is to connect to the **Service Port** of the PLC.

In the ProSoft Configurator for Modicon go to the Tool menu and select the *DeviceFlash* option.

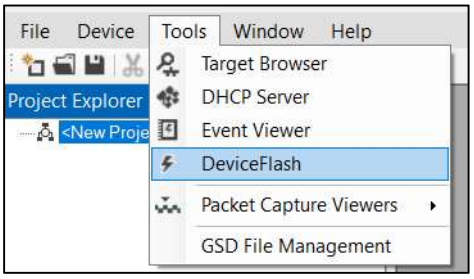


Figure 4.41 - DeviceFlash Tool

The user will need to select the appropriate AFB binary file which will be used to upgrade the PXM firmware.

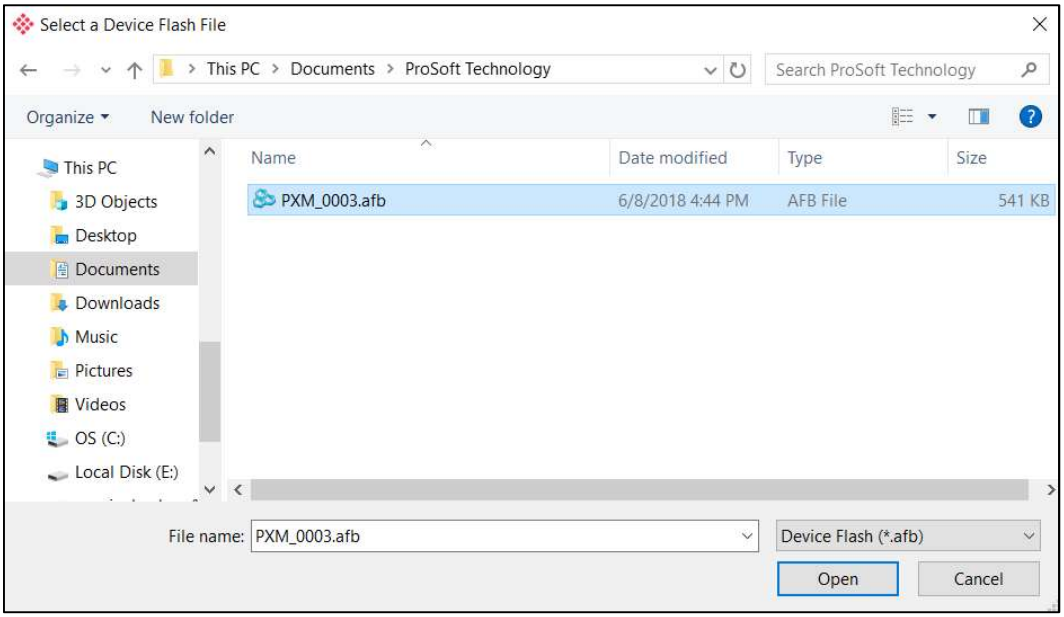


Figure 4.42 - Select the AFB binary

The Target Browser will then open allowing the user to browse to the PXM module to be upgraded.

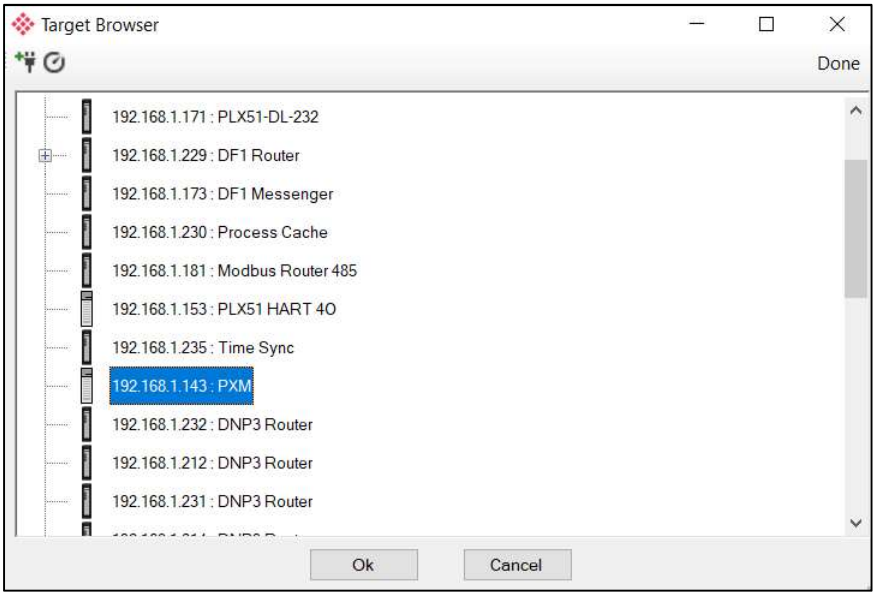


Figure 4.43 - Select the correct PXM module

When connecting to the M580 rack using a NOC module, then the PXM will not appear in the browse list automatically. To manually connect to the PXM, select ***Scan IP Node*** button at the top of the Target Browser and input the PXM's IP Address. The PXM will then appear and can be selected.

Figure 4.44 – Add PXM manually (Connecting via NOC)

Once the module is done upgrading the firmware the Device Flash tool will provide the user with the details of the updated module.

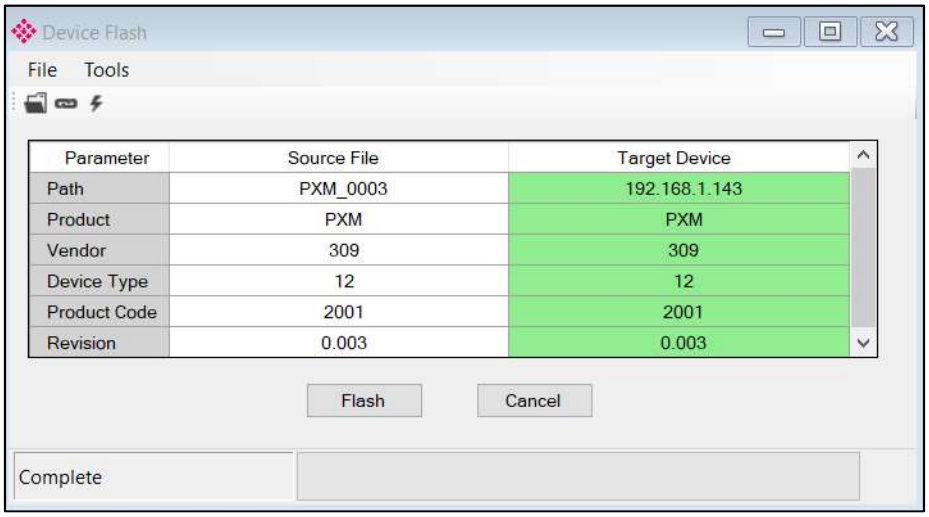


Figure 4.45 – PXM successfully updated.

The PXM firmware is digitally signed so the user will only be able to flash the PXM with authorized firmware.

5. HSBY

5.1. INTRODUCTION

The PXM can operate in an M580 Hot-Standby (HSBY) control architecture. In such a system there are two PLC CPUs, A and B. Provided they have been configured and connected correctly, one of the CPUs will be the Primary and the other the Standby. Should some failure occur on the Primary CPU (or rack) e.g. power failure, then the previous Standby PLC would take over and become the new Primary. The user is also able to trigger a SWAP using application code, which would see the two PLC CPUs swapping their Primary and Standby roles.

5.2. PXM IN HSBY

Using the PXM in an HSBY system requires a pair of identically configured PXMs placed in the same slot of both the PLC A and PLC B rack. The PXM's PROFIBUS DP ports must be connected to the same PROFIBUS network.

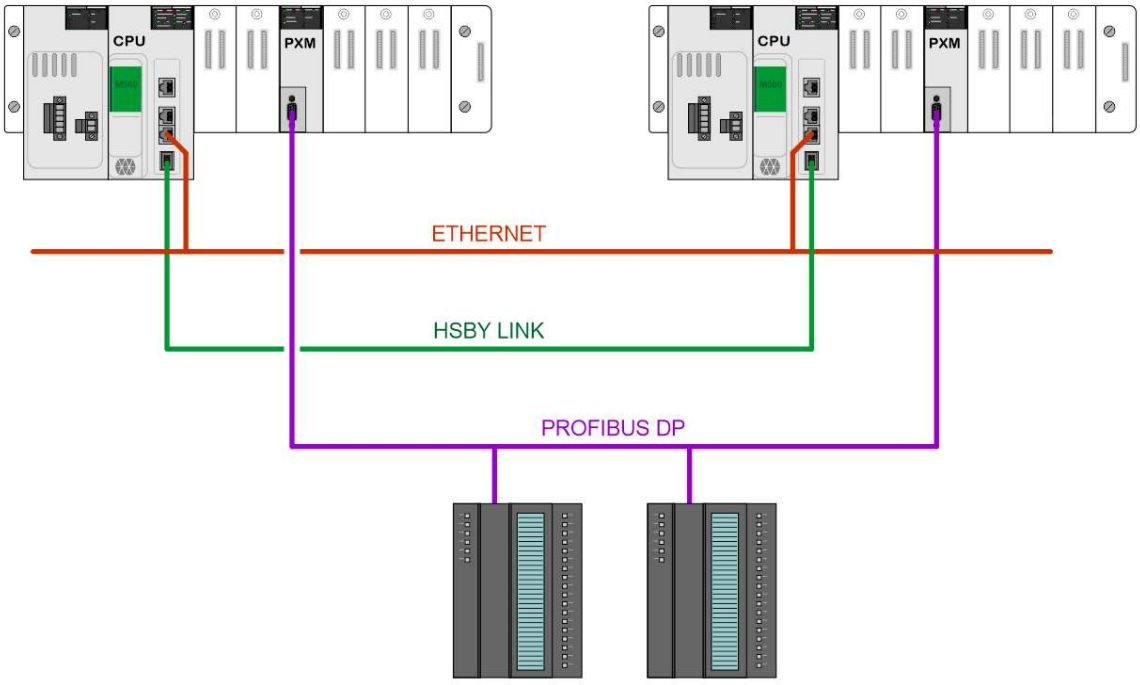


Figure 5.1 – PXM HSBY architecture

Ensure the HSBY system is connected correctly. In addition to the HSBY link, the M580 CPUs must also be connected on the Ethernet network to allow the communication between the two PXMs (A and B).

PXM A and PXM B will retain their specific IP addresses irrespective of which PLC is the Primary.

The Primary PLC CPU will establish a class 1 EtherNet/IP connection to both the PXMs (PXM and PXM B). The Standby CPU will not establish a connection to either of the PXMs. This is illustrated in the two figures below, when the red arrow implies connection establishment.

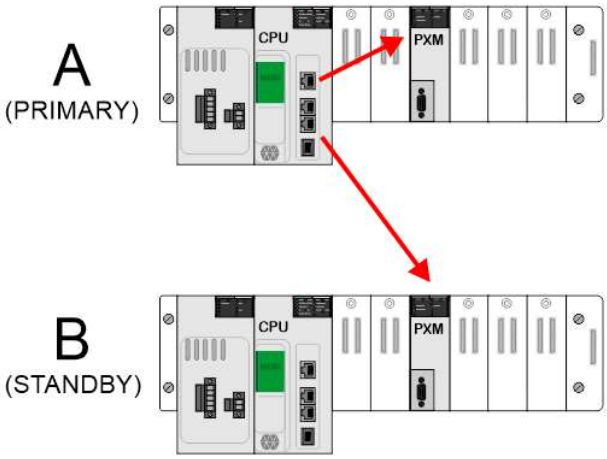


Figure 5.2 – HSBY – PLC A as Primary

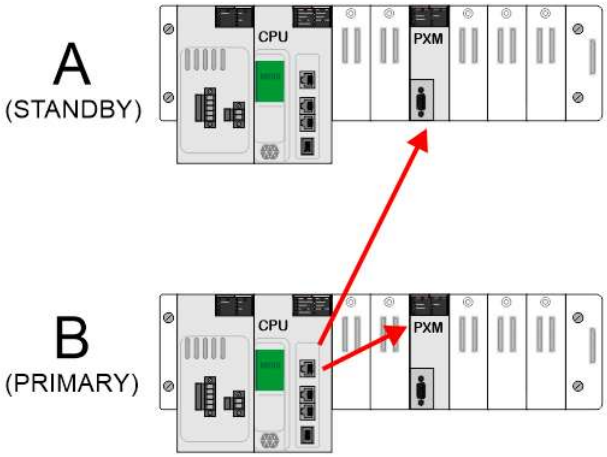


Figure 5.3 – HSBY – PLC B as Primary

During a SWAP, the previously Primary PLC will stop exchanging data with the PXM pair and the new Primary PLC will establish new class 1 connections (Forward Open) to both PXM's. The time taken for the inter-PLC SWAP to complete and the new connections to be established depends on the size and nature of the user application code.

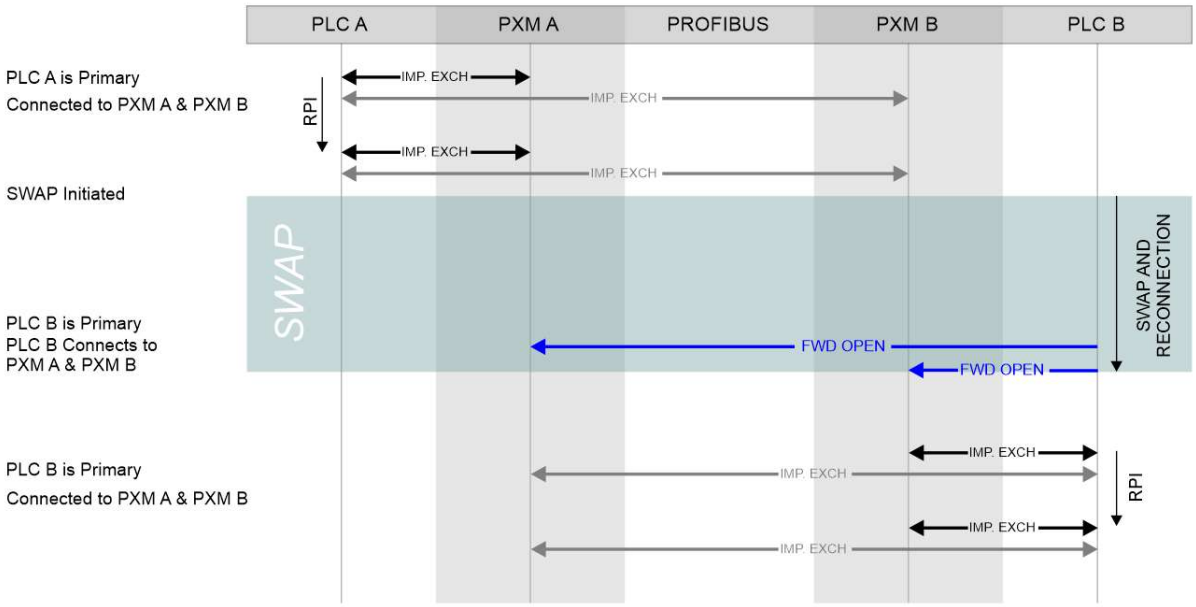


Figure 5.4 – HSBY – Example SWAP PLC A to PLC B

Once an EtherNet/IP class 1 connection has been established between a PLC and a PXM, they will exchange (implicit) input and output data respectively at the RPI (Recommended Packet Interval) rate.

The master mapping DFB adds the status of the PLC to the implicit (class 1) output data being sent to each PXM. This allows the PXM to determine which PLC (A or B) is connected to it and the status of the PLC (Run / Stop, Primary / Standby etc.).

Normally, each PXM will determine whether it should be the PROFIBUS master based on the status of the PLC that connects to it. That is, the “PXM follows the CPU in its local rack”.
So, if PLC A is connected to PXM A and it is in Run-Primary, then PXM A will assume the PROFIBUS master role and PXM B will be in a Standby role.
Conversely, if PLC B is connected to PXM A and it is in Run-Primary, then PXM A will be in a Standby role and PXM B will assume the PROFIBUS master role.

There are, however, some short-term exceptions to the above rule to ensure the PROFIBUS slave devices do not timeout during the SWAP (and thus remain in a Data Exchange state). This is required as the PROFIBUS device timeouts are typically significantly shorter than the SWAP and reconnections times. To achieve this, the standby PXM monitors the PROFIBUS DP traffic. If no new DP packets have been detected for more than the configured ***HSBY DP Deadtime***, then the Standby PXM will take-over as the PROFIBUS master. If no actual PLC SWAP is executed within the ***HSBY Holdover*** time, then the Standby PXM will relinquish its temporary master role. This behaviour will repeat every 10-20 seconds. This cyclic behaviour will be observed if the PROFIBUS DP cable between the Primary and Standby PXM is severed.

There are basically two conditions that can cause a PLC switch over (SWAP):

- Controlled SWAP - typically initiated by the user or user application code
- Uncontrolled SWAP - typically caused by hardware or power failure.

5.2.1. CONTROLLED SWAP

The controlled SWAP mechanism is illustrated in the timing diagram below.

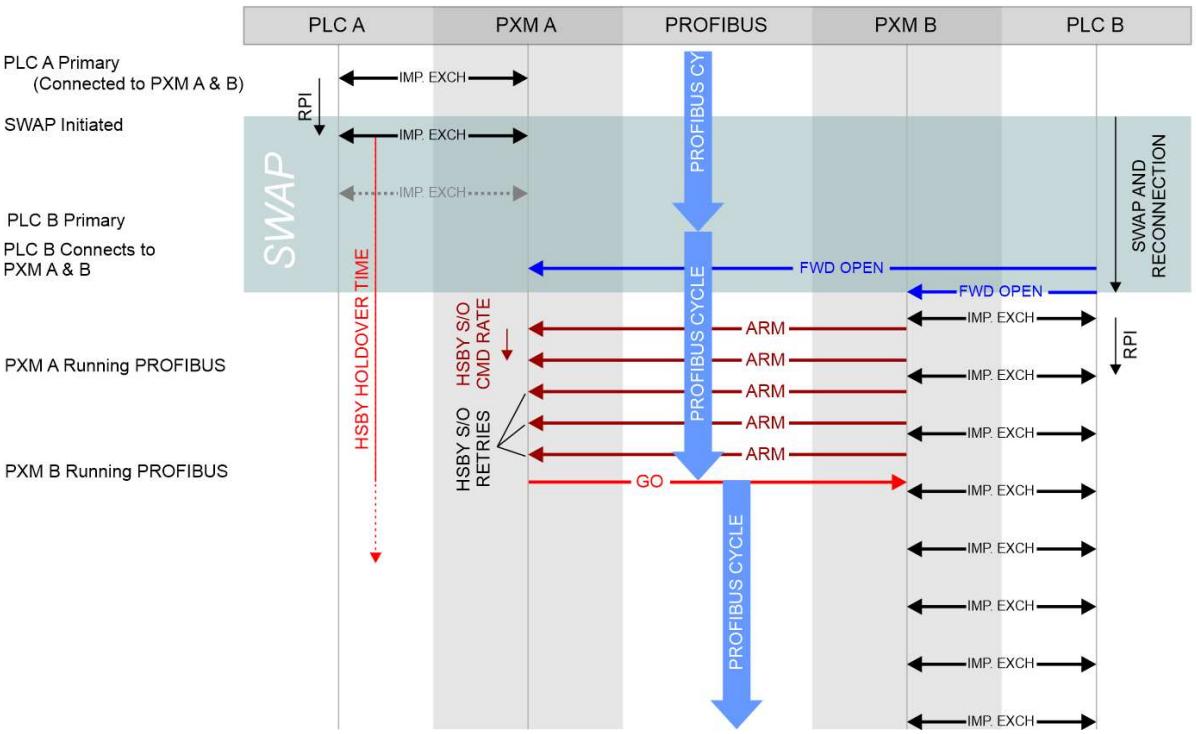


Figure 5.5 – HSBY – Example Controlled SWAP PLC A to PLC B

In this example the system starts with PLC A being the Primary and connected to both PXM A and PXM B. Implicit data (class 1) is exchanged between the PLC and the PXMs every RPI. Note that in the diagram the data exchanges with PXM B have been omitted for the sake of clarity.

At this point PXM A is the PROFIBUS master.

The SWAP command is then initiated, either by the user via Control Expert, or by user application code.

During the SWAP process, involving synchronization between PLC A and PLC B, PXM A remains the PROFIBUS master. Because PXM A is the PROFIBUS master, but no longer has a valid connection from a Primary PLC in its local rack, it will start the internal HSBY Holdover timer. Should this timer reach the user-configurable ***HSBY Holdover*** time, then the Profibus network would go Offline.

Once the inter-PLC SWAP has completed, the new Primary PLC (B) establishes connections to both PXM A and PXM B. (Note that exchanges between PLC B and PXM A have been omitted in the diagram for the sake of clarity.)

Although PLC B is now the Primary PLC, PXM A will remain the PROFIBUS master until the end of the current PROFIBUS cycle.

As soon as PXM B is ready to assume the PROFIBUS master role, by virtue of being connected to a Primary running PLC in the local rack, it will notify the other PXM through a series of "ARM" commands. These inter-PXM EtherNet/IP commands are sent at the user-configurable ***HSBY Switch Over Command Rate***, and continue until either the ***HSBY Switch Over Retry Limit*** has been exhausted or a "GO" command is received from the other PXM.

As soon as PXM A completes the current PROFIBUS cycle, it relinquishes the master role by sending the "GO" command to PXM B.

PXM B immeditaly assumes the role of PROFIBUS master and starts a new PROFIBUS cycle.

5.2.2. UNCONTROLLED SWAP

The uncontrolled SWAP mechanism is typically triggered by hardware or power failure of the current Primary rack or PLC. It is not automatically triggered by PXM removal or failure, nor any form of PROFIBUS network or media failure.

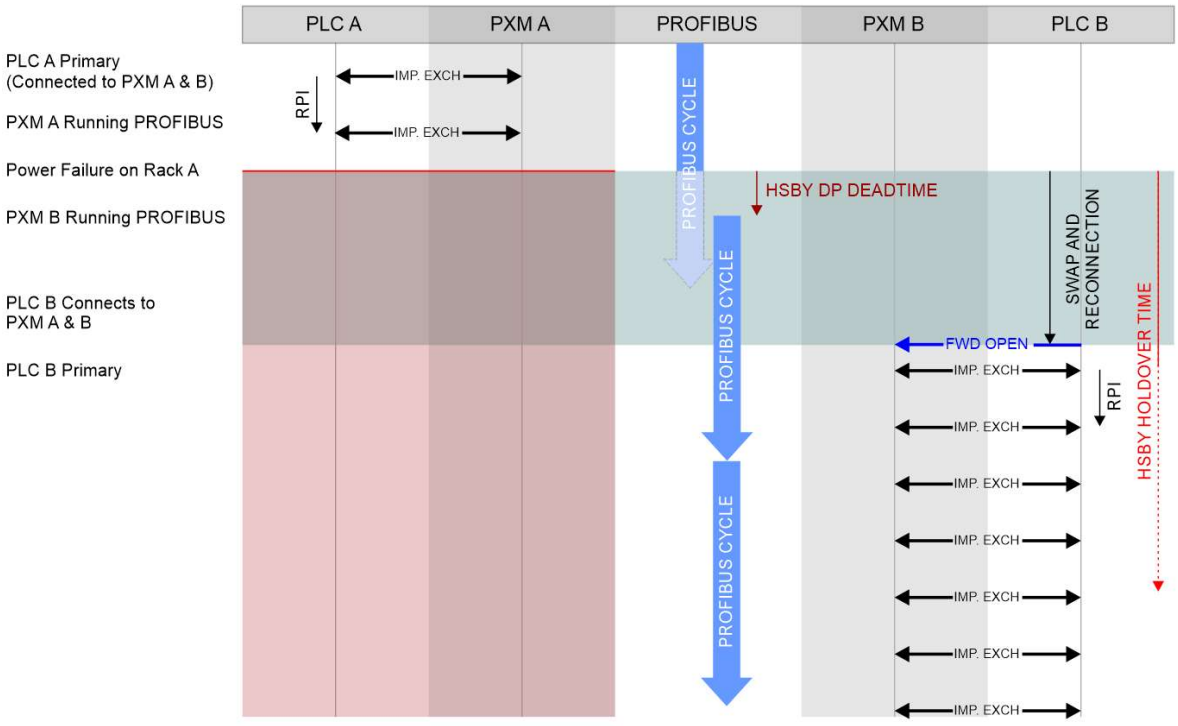


Figure 5.6 – HSBY – Example Uncontrolled SWAP PLC A to PLC B

In this example, similar to the previous one, the system starts with PLC A being the Primary and connected to both PXM A and PXM B. Implicit data (class 1) is exchanged between the PLC and the PXMs every RPI. (Note that in the diagram the data exchanges with PXM B have been omitted for the sake of clarity.)
At this point PXM A is the PROFIBUS master.

The power is then removed from rack A, instantly switching off both PLC A and PXM A.

The PXM B (in standby) continuously monitors the traffic on the PROFIBUS DP port. Should no valid packets be received for a period exceeding the user-configurable ***HSBY DP Deadtime*** then it would assume the PROFIBUS master role.

PXM B is now the PROFIBUS master and will start a new PROFIBUS cycle using output data previously received from PLC A before it disconnected.

Because PXM B is the PROFIBUS master but does not yet have a valid connection from a Primary PLC in its local rack, it will start the internal HSBY Holdover timer.
Should this timer reach the user-configurable ***HSBY Holdover*** time, then the Profibus would go Offline.

The new Primary (PLC B) then establishes a connection to PXM B and the system continues to run normally with PLC B exchanging data with PXM B.

5.3. CONFIGURATION

The user must configure the PXM Master Mode to support HSBY:

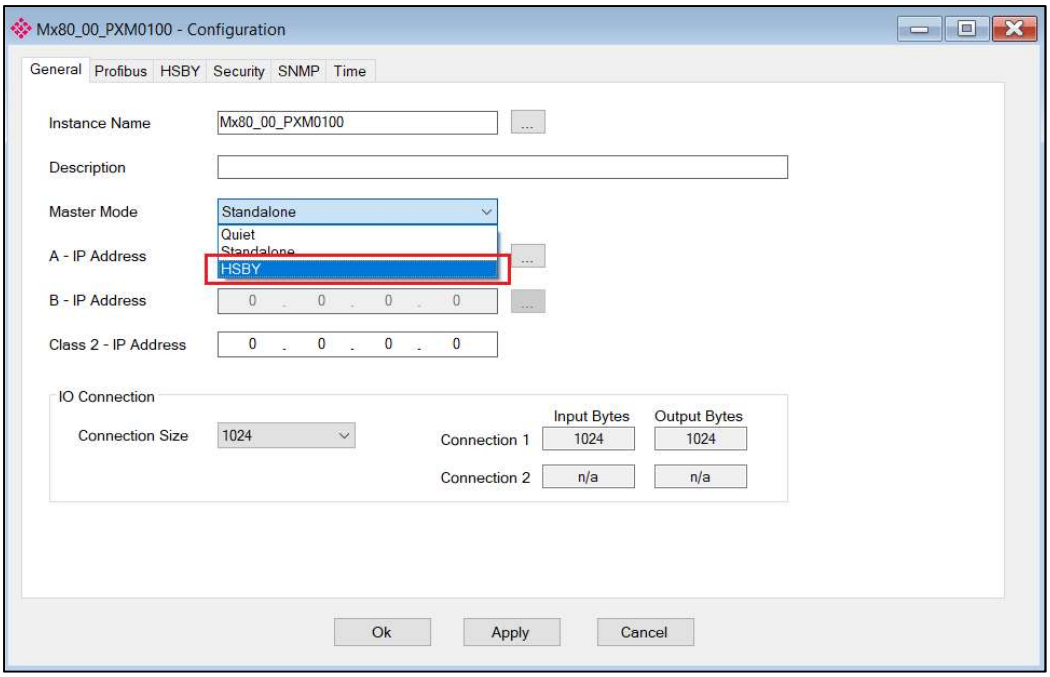


Figure 5.7 – Master Mode selection

Next the IP address of the two PXM modules must be entered as shown below:

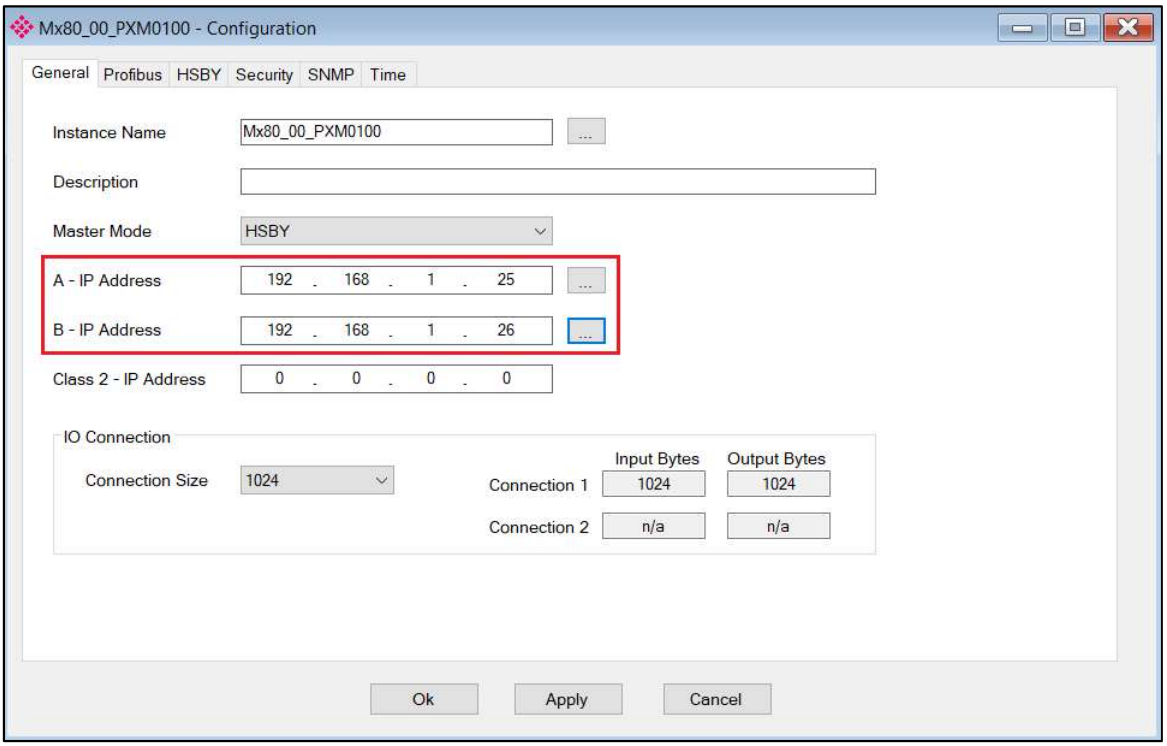


Figure 5.8 – HSBY PXM IP addresses

Once the user applies the changes the ProSoft Configurator for Modicon will indicate that there is a PXM HSBY pair.

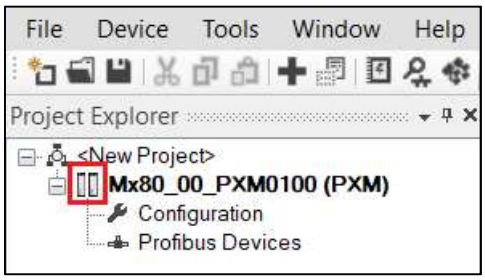


Figure 5.9 – HSBY PXM pair

A number of HSBY parameters must be configured to ensure the correct HSBY behaviour.

In most cases the recommended values (***Recommend*** button option) will be adequate, but in some more complex systems, some parameters may need to be adjusted. The ***Recommend*** function relies on the correct PXM RPI and MAST Task Period parameters being entered correctly. These parameters must match those configured in the Control Expert project.

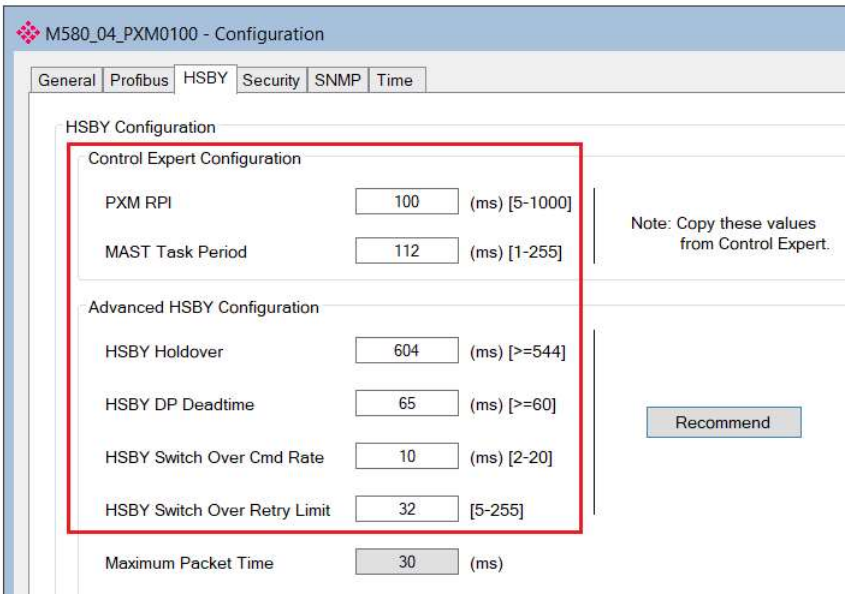


Figure 5.10 – HSBY Parameters

5.3.1. HSBY HOLDOVER

The ***HSBY Holdover*** is the amount of time the active PXM will keep running the PROFIBUS DP network without an EtherNet/IP Class 1 connection from a Running and Primary M580 PLC located in its local rack.

The Holdover time is required to allow uninterrupted operation of the PROFIBUS network during a system SWAP. During this time the PROFIBUS devices will hold their last state.

If this value is too low, then during a SWAP, the PROFIBUS network would enter an Offline state, and all the slave devices would need to be reconfigured and re-parameterized.
If this value is too large, the PROFIBUS network continues to operate with outdated output data for a prolonged period.

The HSBY Holdover value must, at a minimum, be equal to the sum of the following:

- 1 x SWAP and Reconnection Time (depends on application code, typically 100-200ms)
- 1 x MAST Task period
- 1 x RPI (5-1000ms)
- 1 x PROFIBUS Cycle Time (depends on BAUD rate and number of devices, etc.)
- 2 x HSBY Switch Over Command Rate (2-20ms)

Due to the asynchronous nature of these parameters it is also recommended to add a small margin to the value. An example of a calculation is shown below:

Parameter	Value (ms)	Multiplier	Total (ms)
PLC SWAP and Reconnection Time	200	1	200
MAST Task Period	10	1	10
RPI	10	1	10
PROFIBUS Cycle	112	1	112
Switch Over Command Rate	10	2	20
Margin	60	1	60
Recommended Value (Total)			**412**

Table 5.1 – HSBY Holdover Calculation Example

5.3.1. HSBY DP DEADTIME

The DP Deadtime is the amount of time the standby PXM will wait between the PROFIBUS DP network is quiet before taking over as the PROFIBUS DP Master. It is effectively the maximum time between successive valid PROFIBUS DP packets.

The DP Deadtime value must, at a minimum, be equal to two of the largest expected PROFIBUS packets. As an aid in this calculation the ***Maximum Packet Time*** is displayed on the configuration window. It takes into account the BAUD rate and the largest PROFIBUS DP DPV0 data exchange.

5.3.2. HSBY SWITCH OVER COMMAND RATE

During a controlled SWAP, the new Primary PXM will send an “ARM” command to the current PROFIBUS master PXM at this interval.
The ***Switch Over Command Rate*** value should generally not exceed half the PROFIBUS cycle, otherwise there may be a delay between the last PROFIBUS cycle of the previously Primary PXM and the first PROFIBUS cycle of the new Primary PXM. However, in systems where the RPI is very large in relation to the PROFIBUS Cycle then this value would need to be increased to more than the PROFIBUS Cycle.

Even in the case of relatively large PROFIBUS cycles, the ***Switch Over Command Rate*** should not exceed 20ms.

5.3.3. HSBY SWITCH OVER RETRY LIMIT

During a controlled SWAP, the new Primary PXM will send many “ARM” commands to the current PROFIBUS master PXM. If no valid “GO” command is received before this limit is

reached, then the new Primary PXM will assume the previous PXM is not available and take-over as PROFIBUS master anyway.
This condition should only be realised if a hardware or power failure occurred shortly after a SWAP command was issued.

The user must ensure that the ***HSBY Switch Over Retry Limit*** when multiplied by the ***HSBY Switch Over Command Rate*** and added to the best-case SWAP and Reconnection time is greater than the ***HSBY Holdover*** time.

⚠ ***CAUTION***
UNSUCCESSFUL SWAP OR INCORRECT PXM OPERATION In an HSBY system, ensure the HSBY parameters are configured correctly and match the configuration inside Control Expert. **Failure to follow these instructions can result in injury or equipment damage.**

5.4. DOWNLOAD CONFIGURATION

From here the user can configure the PXM as per normal. Once completed the user can download to the PXM pair. This is done by right-clicking on the PXM and selecting download. The ProSoft Configurator for Modicon will download the configuration to both PXM modules to ensure that the configuration in both PXM modules are the same. Once online the user will also see that there is status for PXM A and status for PXM B respectively.

If the user selects the device status or packet capture the ProSoft Configurator for Modicon will only show the values of the active PXM.

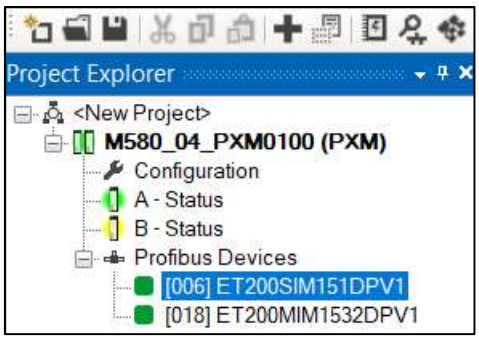


Figure 5.11 – Online with HSBY PXM pair

After downloading new configuration to the PXMs, ensure that the configuration has been uploaded to the FDR server.

⚠ CAUTION
UNEXPECTED RESULTS FOLLOWING AN HSBY SWAP Ensure that the FDR servers are up to date. **Failure to follow these instructions can result in injury or equipment damage.**

The configured Class 2 IP address can be used for DPV1 Class 2 messaging (e.g. DTM). When operating in HSBY mode only the active DP Master will have this IP address enabled. When a HSBY swap occurs, the new active DP Master will enable this IP address and the new standby DP Master will disable this IP address.

5.5. CONTROL EXPERT SETUP

The user must setup the two PXM modules in Control Expert similar to instantiating a PXM in standalone mode (see *Control Expert Configuration* section). The primary M580 controller will connect to both PXM modules.

The user will also need to export and import the Control Expert mapping similar to a PXM in standalone mode (see *Control Expert Configuration* section).

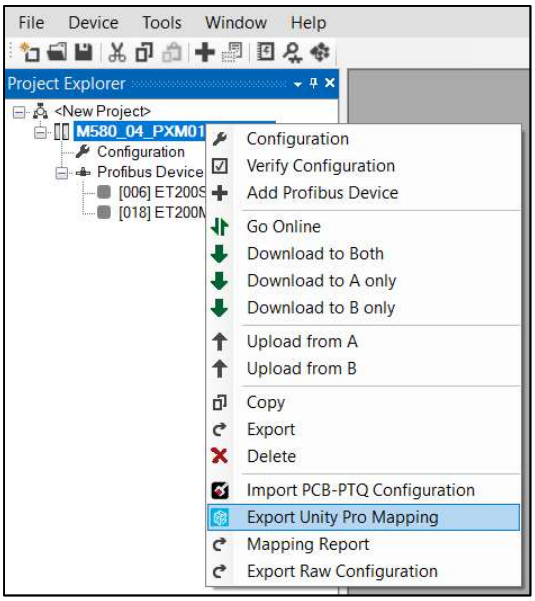


Figure 5.12 – Export Control Expert Mapping for HSBY

The Control Expert mapping will also include all the operational code required for the PXM operating in HSBY mode.

5.6. OPERATION

The active PXM will operate as a single DP Master on the PROFIBUS network under normal operation with the standby PXM acting as standby. Only when there is a M580 controller swap will the PXM pair swap as well.

5.6.1. CONTROL EXPERT OPERATION

In Control Expert the user will only look at a single DDT for the PXM which is used for operation in a HSBY system (similar to when operating in standalone mode). See the *Control Expert Mapping* section for mapping details. The mapping will determine which PXM is active and only update the PXM DDT with the active data.

5.7. DIAGNOSTICS

5.7.1. PROSOFT CONFIGURATOR FOR MODICON DIAGNOSTICS

The status for each PXM can be accessed by clicking on either *A – Status* or *B – Status.* This will show each PXM status individually. However, when viewing the status of the devices in HSBY mode only the data from the active PXM will be displayed.

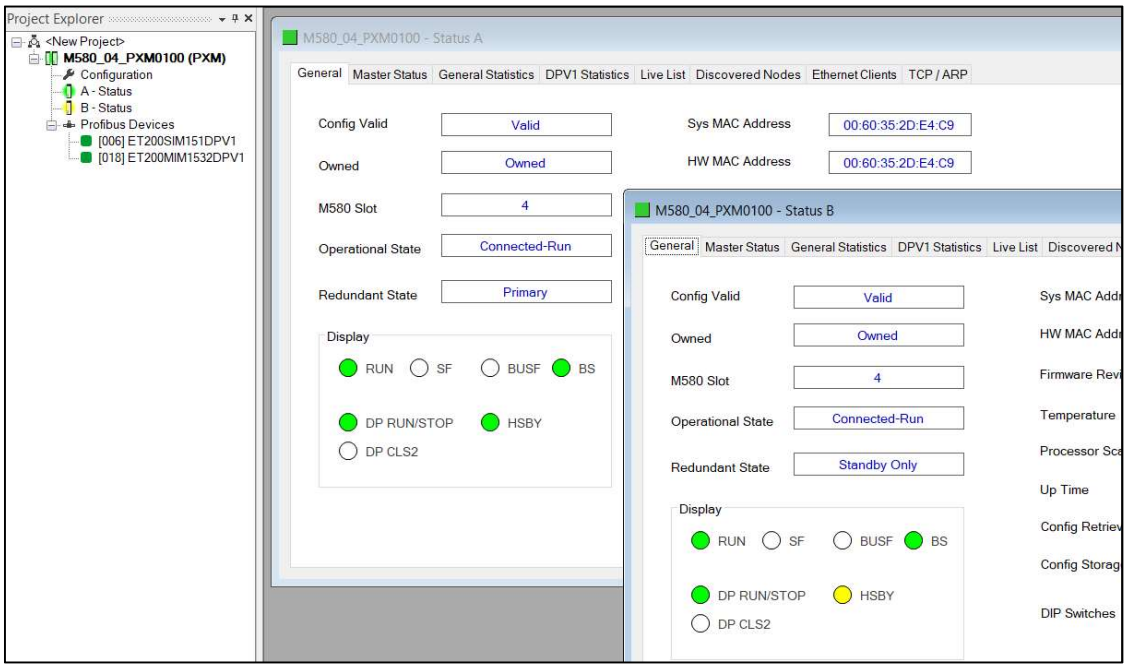


Figure 5.13 – ProSoft Configurator for Modicon PXM diagnostics in HSBY Mode

6. MIGRATING PTQ-PDPMV1 PROJECTS

The ProSoft Configurator for Modicon provides a method to simplify the migration from a PTQ-PDPMV1 application to a PXM application. The process involves first exporting the configuration from the PTQ-PDPMV1 using the ProSoft Configuration Builder (PCB) software and then importing it into the ProSoft Configurator for Modicon.

Before importing a PTQ-PDPMV1 application, ensure that all the necessary GSD files have first been registered in the ProSoft Configurator for Modicon's GSD Manager.

The PTQ-PDPMV1 import requires a ProSoft Configurator for Modicon project with no existing slave devices.

Open the ProSoft Configuration Builder (PCB) software and open the existing PTQ-PDPMV1 project.

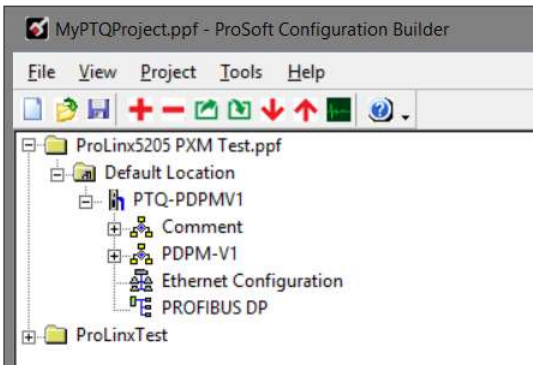


Figure 6.1 – PCB PTQ-PDPMV1 Project

Expand the PTQ-PDPMV1 configuration and right-click on the ***PROFIBUS DP*** menu item and select the ***Configure*** option.

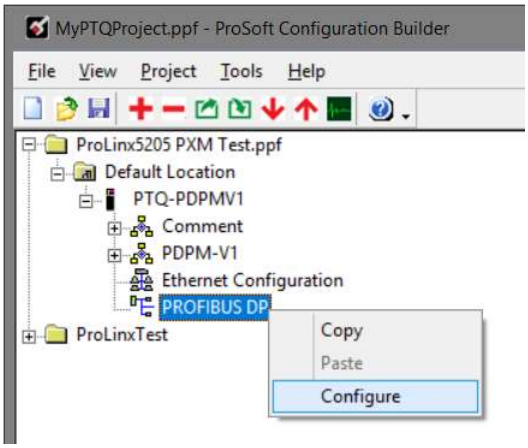


Figure 6.2 – PCB PTQ PROFIBUS DP Configuration

The PROFIBUS Master window will open. Select the ***Export Master Config*** button located at the bottom of the window.

If the Export Master Config button is disabled, then first select the Configure PROFIBUS button. This will open the PROFIBUS configuration tool, once this tool has been closed the Export button will then be enabled.

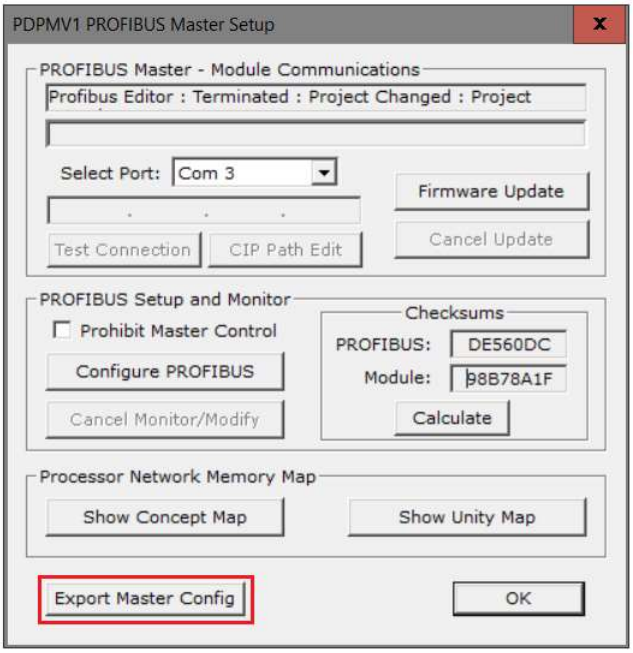


Figure 6.3 – PCB PTQ Export Master Configuration

The generated export (XML) file can then be saved.

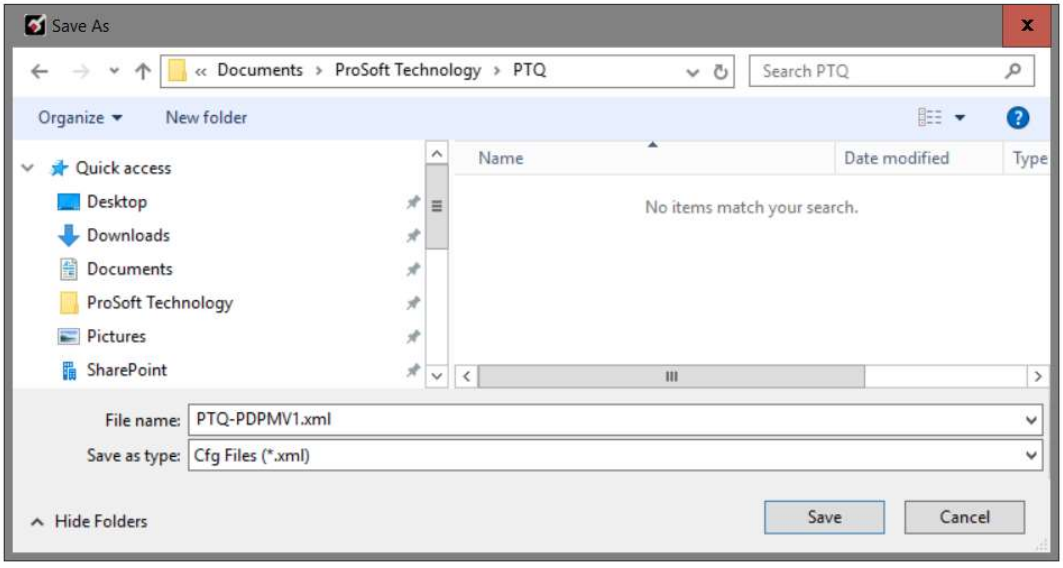


Figure 6.4 – PCB PTQ – Save Export XML File

In the ProSoft Configurator for Modicon, right-click on the PXM module and select the ***Import PCB-PTQ Configuration*** option.

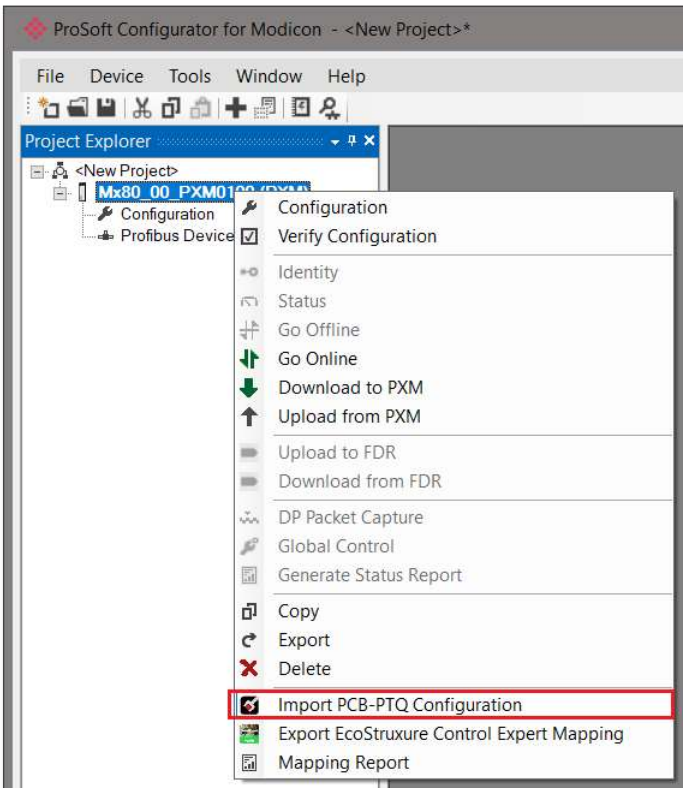


Figure 6.5 – Import PCB-PTQ Configuration

After the import is complete, a brief Import report will be shown, and may indicate any issues encountered during the import process.

If an exact GSD filename is not found, a suitable (matching) alternative will be sought. The details of this will be displayed in the import report.

The master and slave device configurations can then be modified if required.

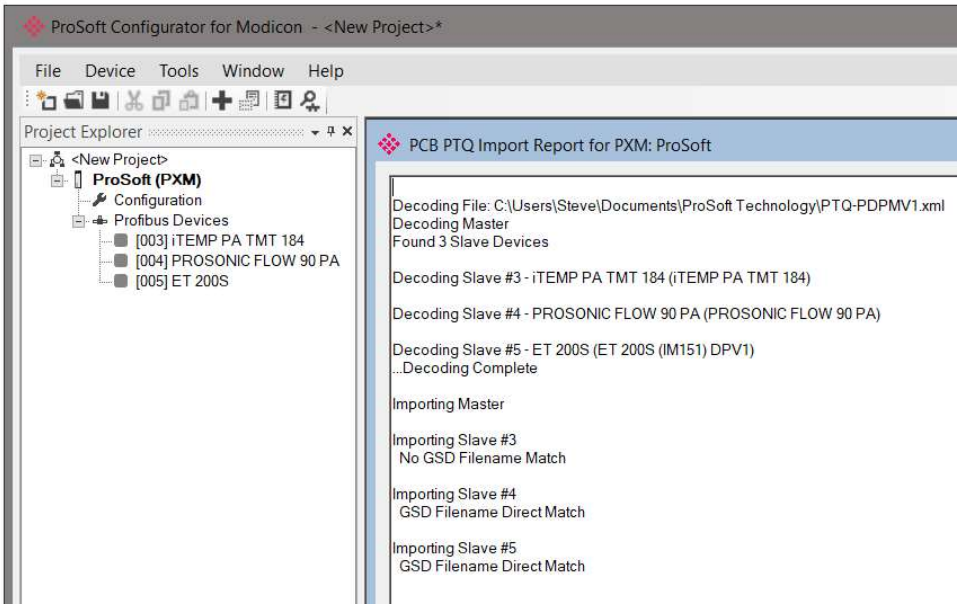


Figure 6.6 – PCB PTQ Import Report

7. DEVICE TYPE MANAGER (DTM)

The PXM supports FDT / DTM technology, allowing the user to configure any slave device using its DTM (Device Type Manager) in any standard FDT Frame (Field Device Tool). To use a device DTM with the PXM, the ProSoft PMEPXM DTM pack will first need to be installed.

7.1. INSTALLATION

Installation of the PMEPXM DTM pack is achieved by executing the following installer:
ProSoft Technology - PMEPXM DTM Pack 1.000 Setup.msi

The installation wizard will guide the user through the installation process.

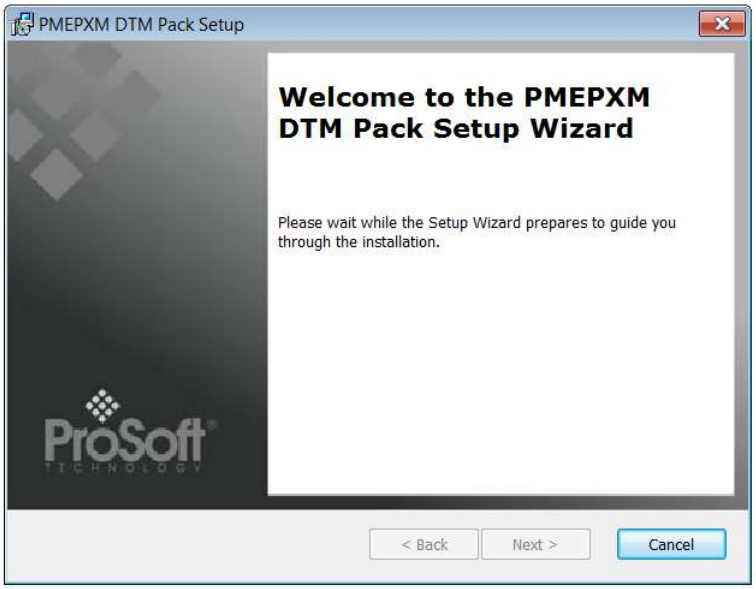


Figure 7.1 – PMEPXM DTM Pack Installation

7.2. CONFIGURATION

Once the DTM pack is installed, the selected FDT Frame would need to have its DTM Catalogue updated. The steps required for this action are slightly different for each FDT frame. Typically, one selects the DTM Catalogue or Device Catalogue and select Refresh or rebuild.

After the catalogue has been updated, the PXM device can then be added to a new project. This involves selecting the Add Device function and then selecting the PXM DTM.
The example below makes use of PACTware FDT frame.

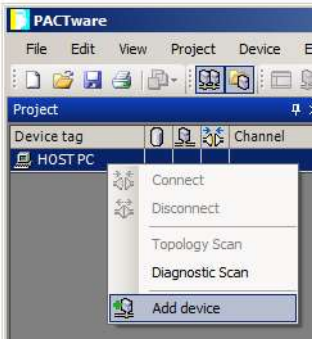


Figure 7.2 – Adding new device

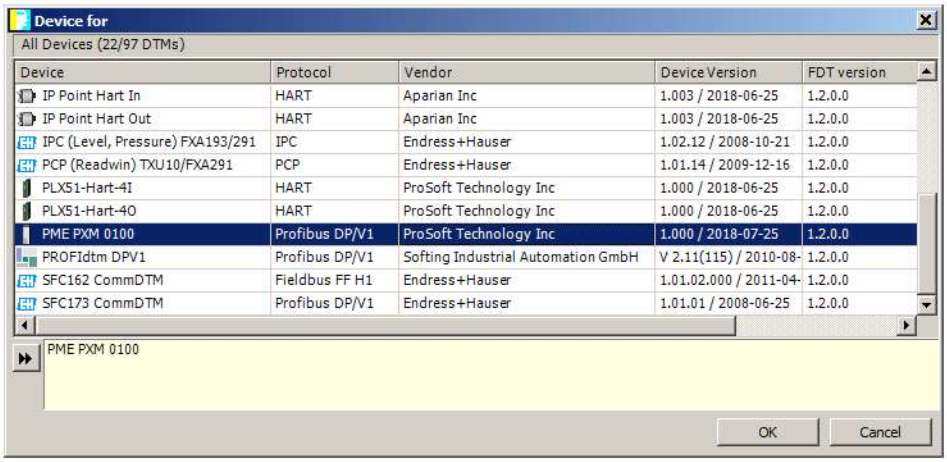


Figure 7.3 – Selecting PXM DTM

After instantiating the PXM DTM, select the Parameter option.

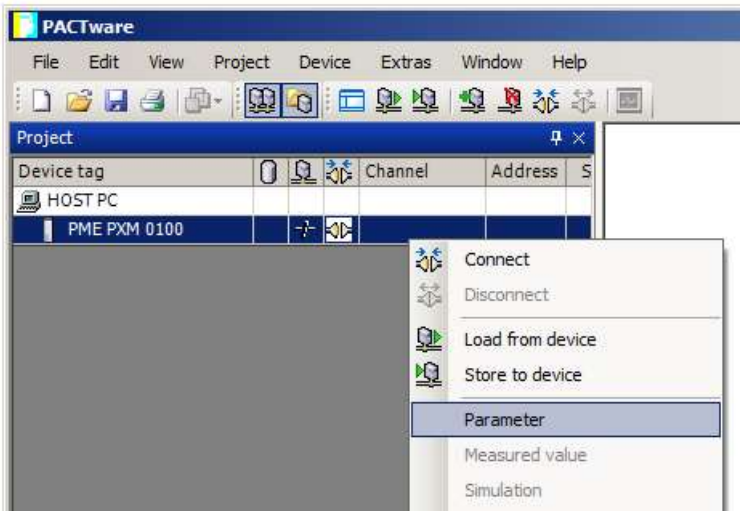


Figure 7.4 – Select Parameter option

The PXM DTM's configuration allows the CIP Path to the PXM to be configured. This is typically just the IP address of the PXM. In a redundant configuration the Class 2 IP address can be used.

Check that the PXMs and PLCs are in the correct state. The Class 2 IP address is only available when one of the PXM modules is in a Primary role. When both PXM are in a Standby role (e.g. disconnected from the PLC, or PLCs in STOP) then the Class 2 IP address will not be available.

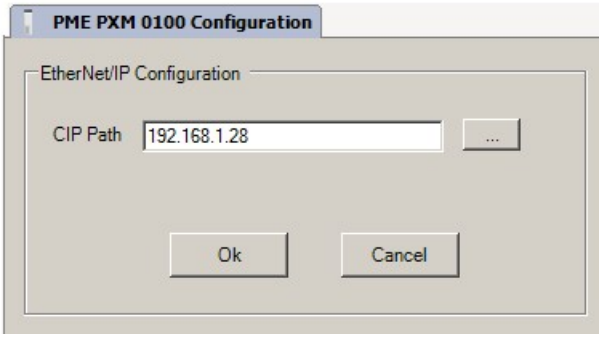


Figure 7.5 – PXM CIP Path

The path can either be entered manually or the Browse button can be used to open the Target Browser, and then the PXM can be selected.

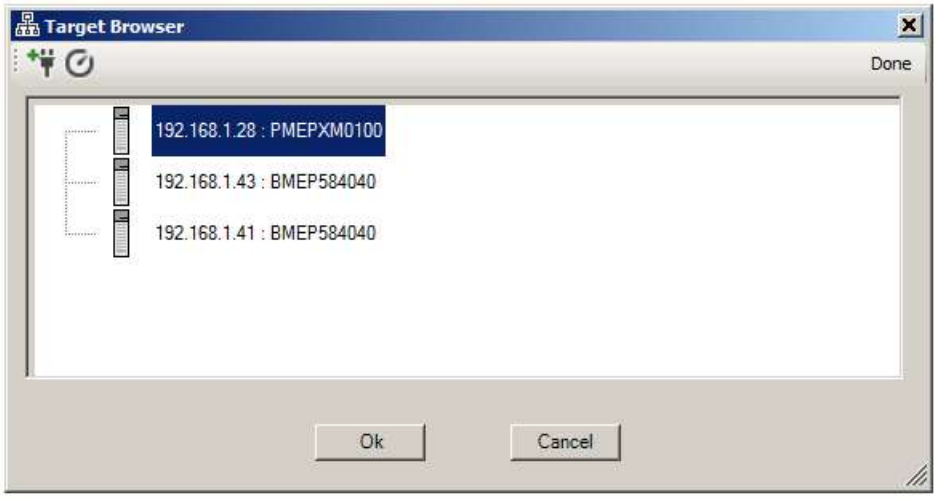


Figure 7.6 – Target Browser

Once the PXM DTM has been configured, the child Device DTMs can be added by right-clicking on the PXM DTM and selecting Add Device.

The user can then select the matching device DTM.

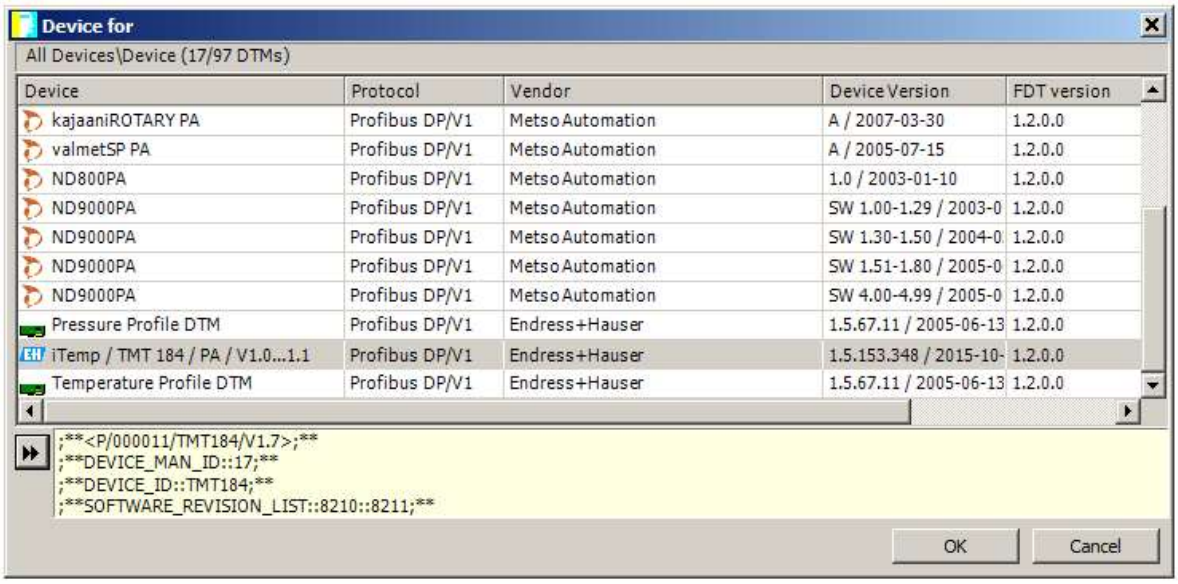


Figure 7.7 – Device DTM Selection

Once the child Device DTM has been added, a configuration window opens to set the Station Node address.

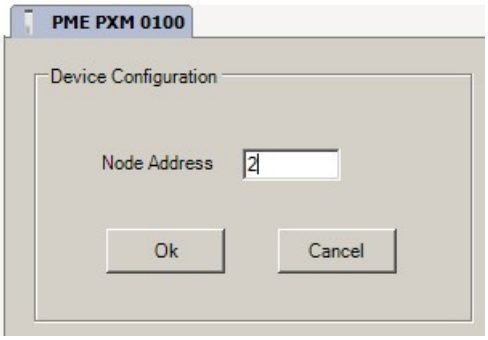


Figure 7.8 – Device DTM Node Address

7.1. OPERATION

After the FDT project has been configured, the DTMs can be place online by selecting the Online or Connect option.

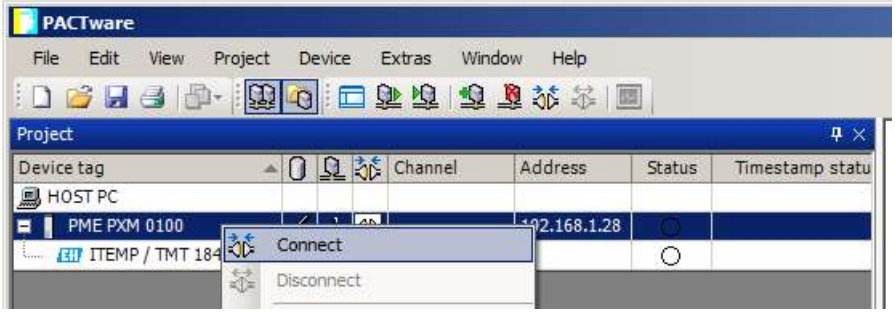


Figure 7.9 – DTM Connect

Once the PXM DTM is online (connected) a number of diagnostic pages can be opened by selecting the Measure Value.

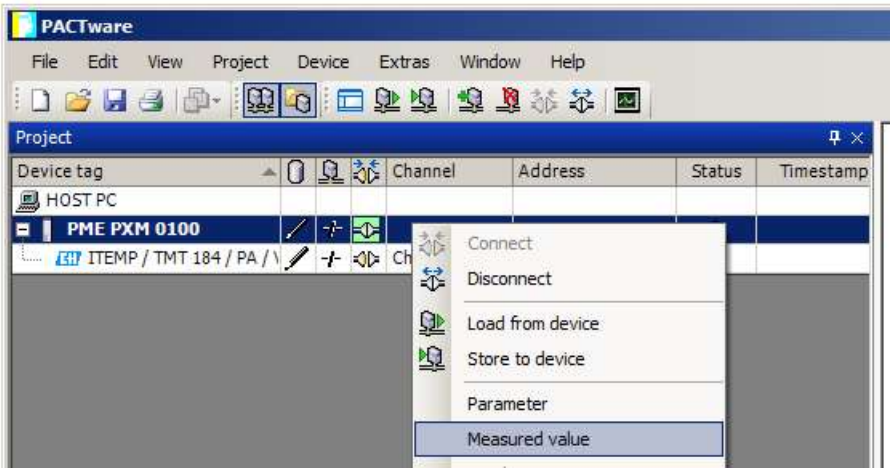


Figure 7.10 – Measured Value

The General page provides basic status information for the PXM module, including LED status and CPU status etc.

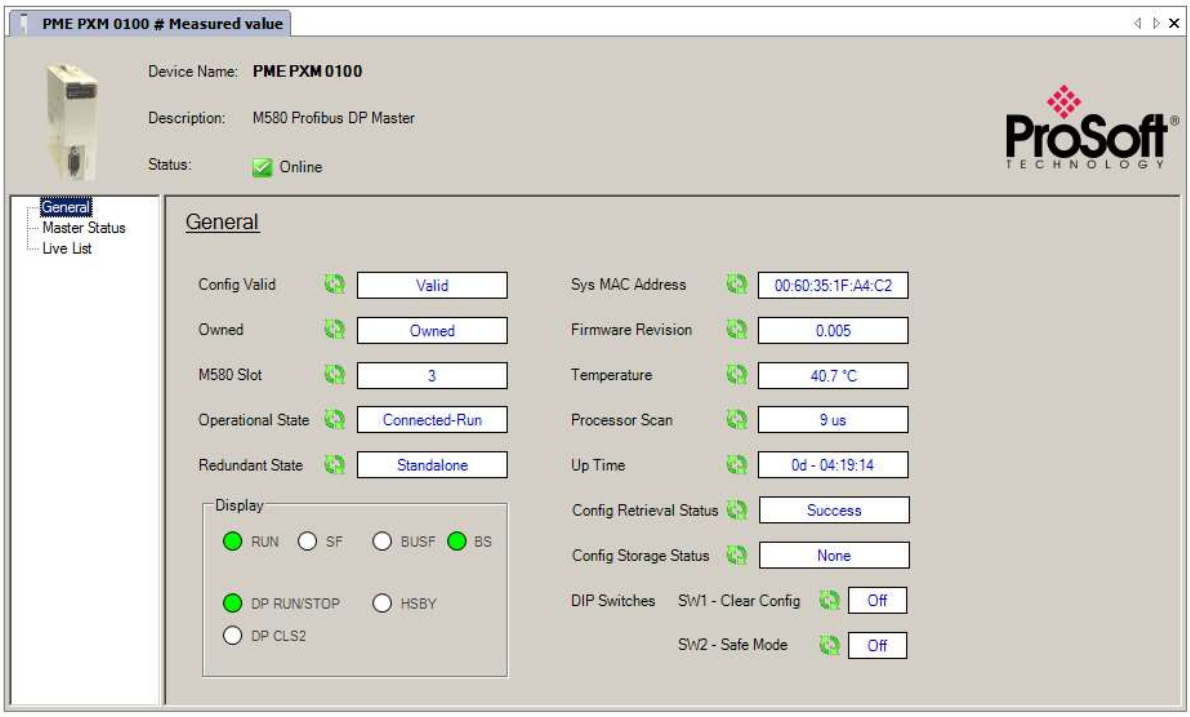


Figure 7.11 – PXM DTM - General Status Page

The Master Status page shows PROFIBUS Master statistics and status.

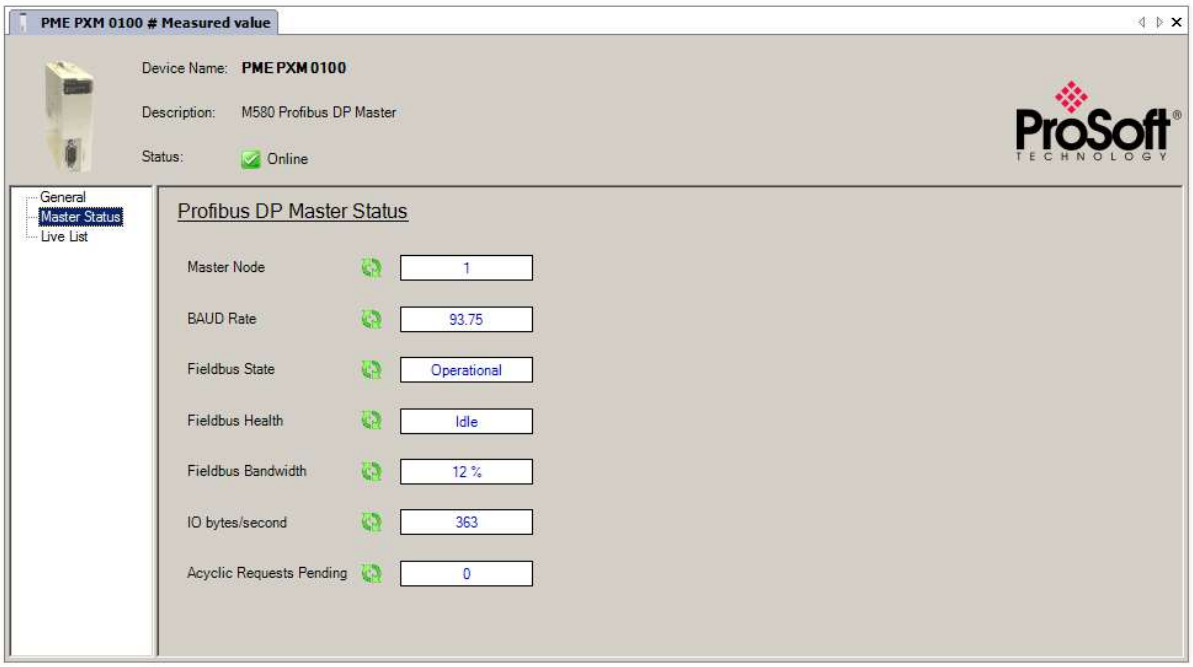


Figure 7.12 – PXM DTM – Master Status Page

The Live List page shows the state of the devices on the PROFIBUS network.

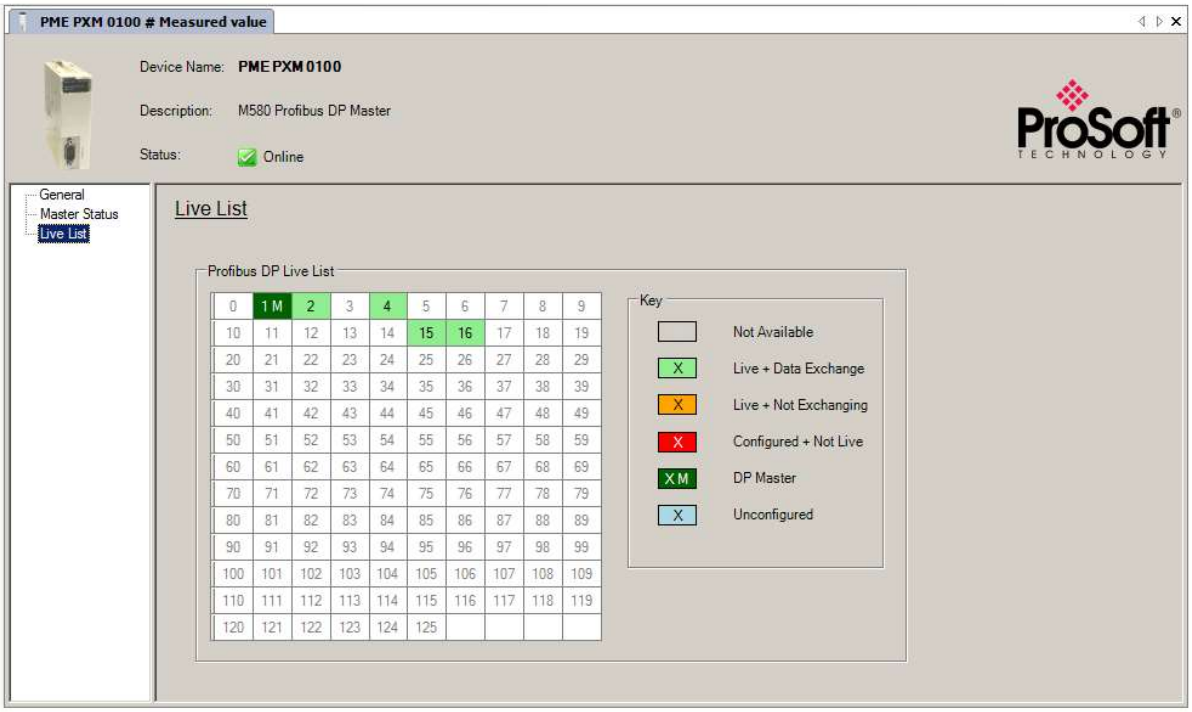


Figure 7.13 – PXM DTM - Live List Page

Slave Device DTM under the PXM DTM can also be brought online by selecting the Online or Connect option.

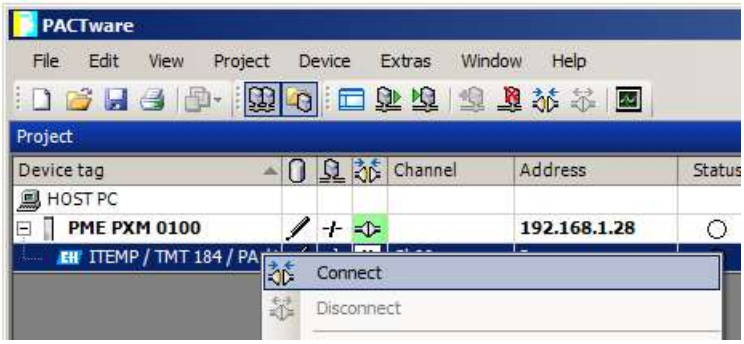


Figure 7.14 – Slave Device DTM Connect

Depending on the device DTM, a number of online parameters, diagnostics and measure variables can be displayed.

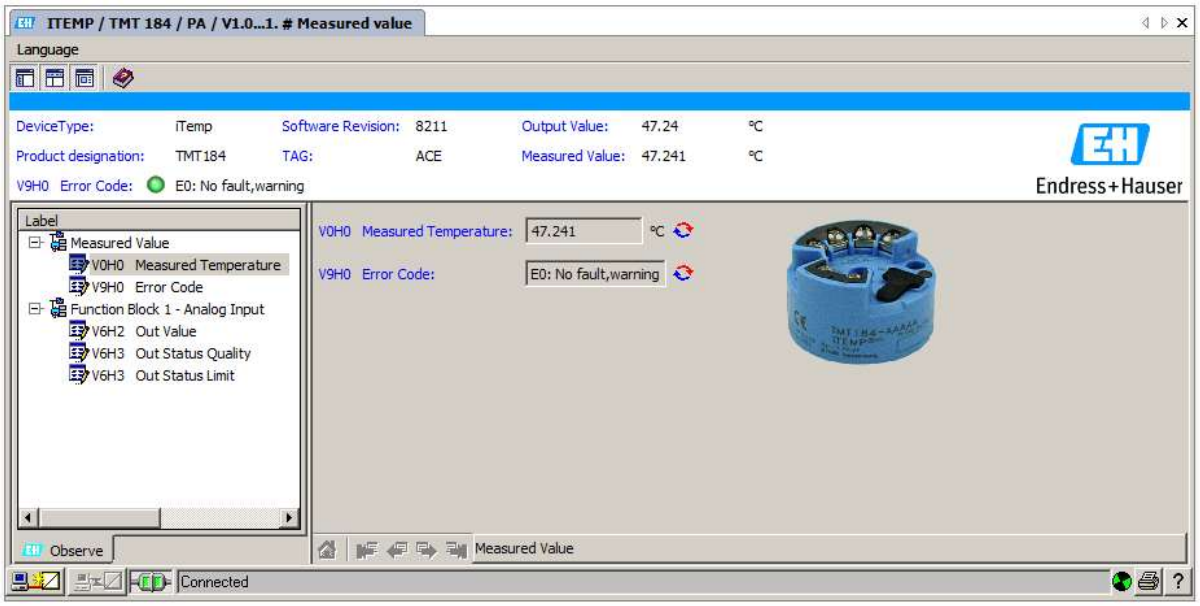


Figure 7.15 – Device DTM

8. DIAGNOSTICS

8.1. LEDS

The module provides seven LEDs for diagnostics purposes as shown below. A description of each LED is given in the table below.

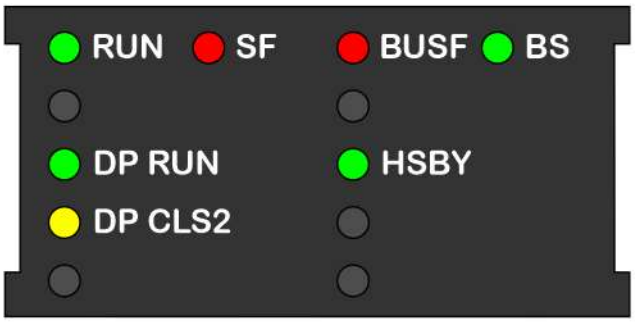


Figure 8.1 - PXM LED Cluster

LED	*Colour*	*Description*
RUN	Green	Module running status
SF	Red	System Fault
BUSF	Red	Profibus DP Fault
BS	Red / Yellow /Green	EtherNet/IP Bus Status
HSBY	Yellow / Green	Hot-standby
DP RUN	Yellow / Green	PROFIBUS DP Running
DP CLS2	Yellow	PROFIBUS DP Class 2 Activity

Table 8.1 – LED Descriptions

Item	RUN	SF	BUSF	BS	Description	Operational State
1	Off	Off	Off	Off	Module is not receiving power.	Off
2	On	On	On	On	Module configuration ignored (DIP Switch 1 On)	Unconfigured
3	Blink	Blink	Blink	Blink	Module is performing its power up self-test.	Initialization
4	Off	On	Off	Off	Module is powered but failed, or Module in Safe-Mode (DIP Switch 2 on) and no IP Address assigned.	Unconfigured
5	Off	On	Off	Blink	Module in Safe-Mode (DIP Switch 2 on) and IP Address assigned.	Unconfigured
6	Off	Blink	Off	On	Module is not configured - Config file missing/invalid.	Unconfigured

7	Off	Blink	Off	Blink	Module configuration uncertain - Local / FDR mismatch.	Unconfigured
8	Off	Blink	Off	Blink	Module configuration unsaved - FDR blank	Unconfigured
9	Off	Blink	Off	Blink	Module configuration transferred from FDR	Unconfigured
10	On	Off	Off	Blink	Module rejects ENIP Connection (incorrect size)	Configured
11	Off	On	Off	On	Module has stopped- fault detected: e.g. MACID duplication	Unconfigured / Configured
12	Blink	Off	Off	Blink	Firmware update.	Firmware Initialization
13	On	Off	-	Off	Started, but no IP address.	Unconfigured / Configured
14	On	Off	-	Blink	Waiting for connection. (IP address but no Forward Open).	Configured
15	On	Off	-	On	Module online (connected)	Connected (Any)
16	Blink	Off	-	On	Module online - At least one device CRC mismatch.	Connected (Any)
17	On	Off	-	Blink	The module's Ethernet connection has timed out.	Configured
18	On	Off	Off	On	PROFIBUS Master is online without error	Connected (Any)
19	On	Off	On	-	PROFIBUS Master is offline in fault.	Connected (Any)
20	On	Off	Blink	-	PROFIBUS Device/s fault	Connected (Any)
21	Blink	Off	Off	Blink	M580 system architecture and PXM config do not match	Connected (Any)

Table 8.2 - Module LED Top Row operation

Note that the dash ("-") indicates any LED state.

Item	DP Run	DP CLS2	HSBY	Description	Operational State
1	Off	Off	-	PROFIBUS Master offline	Unconfigured / Configured/ Connected - OFFLINE
2	On	Off	-	PROFIBUS DPM1 STOP	Connected – STOP
3	On	Off	-	PROFIBUS DPM1 OPERATE	Connected – RUN
4	Blink	Off	-	PROFIBUS DPM1 CLEAR	Connected – CLEAR
5	Not Off	On	-	PROFIBUS Master Class 2 exchange	Connected (Any)
6	-	-	Off	Redundancy Disabled	Unconfigured / Configured/ Connected (Any)
7	-	-	On	Active redundant Master with Standby Ok	Connected (Any)
8	-	-	Blink	Active redundant Master with no Standby	Configured/ Connected (Any)

9	-	-	On	Standby redundant Master	Connected (Any)
10	-	-	Blink	No role redundant Master	Configured/ Connected (Any)

Table 8.3 - Module LED Bottom Row operation

Note that the dash ("-") indicates any LED state.

8.2. MODULE STATUS MONITORING

The PXM provides a range of statistics which can assist with module operation, maintenance, and fault finding. The statistics can be accessed in full by the ProSoft Configurator for Modicon or using the web server in the module.

To view the module's status in the ProSoft Configurator for Modicon environment, the PXM must be online. If the module is not already Online (following a recent configuration download), then right-click on the module and select the *Go Online* option.

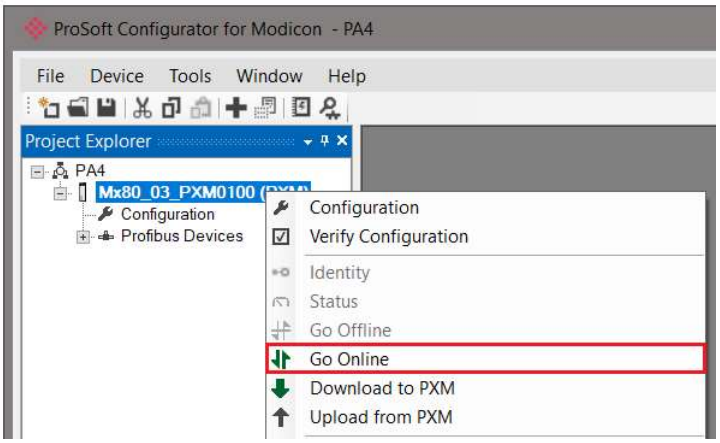


Figure 8.2 - Selecting to Go Online

The Online mode is indicated by the green circle behind the module in the Project Explorer tree.

8.2.1. PXM DP MASTER

The Status monitoring window of the PXM can be opened by either double-clicking on the *Status* item in the Project Explorer tree, or by right-clicking on the module and selecting *Status*.

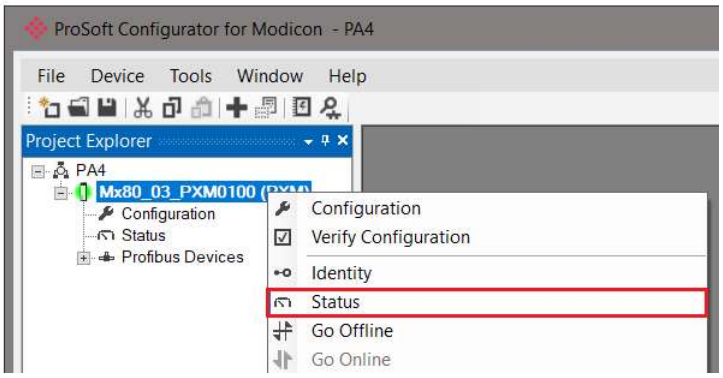


Figure 8.3 - Selecting PXM online Status

The status window contains multiple tabs to display the current status of the module.

8.2.1.1. GENERAL

The General tab displays the following general parameters:

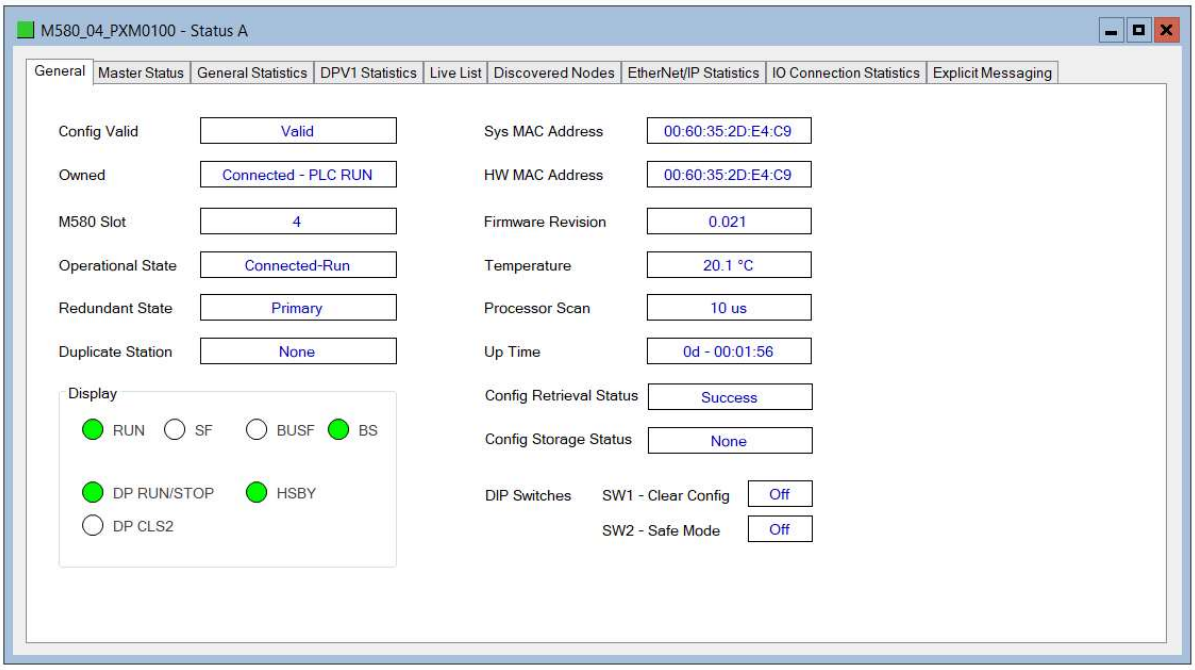


Figure 8.4 – PXM Status monitoring - General

Parameter	Description
Config Valid	Indicates if the downloaded configuration is valid and executing.
Owned	Indicates whether or not the module is currently owned (Class 1) by a M580 PLC and the status of that PLC.

	• Not Connected • Connected – PLC STOP • Connected – PLC RUN In an HSBY system, the Primary PLC controller will establish a connection to both the PXM A and PXM B modules, thus both PXMs will report the status of the Primary PLC.
M580 Slot	This is the physical slot that the PXM is plugged into.
Operational State	This is the operational state of the PXM. The following states can be returned: **Unconfigured** The PXM has not yet been configured using the ProSoft Configurator for Modicon. **Configured** The PXM has been configured by the ProSoft Configurator for Modicon, but is currently not connected to a M580 controller. **Connected-Stop** The PXM has been configured and is connected to a M580 controller and the PROFIBUS operation mode is set to STOP or OFFLINE. **Connected-Run** The PXM has been configured and is connected to a M580 controller and the PROFIBUS operation mode is set to OPERATE or CLEAR. **Fallback** The PXM has been configured and is connected to a M580 controller which is in STOP mode. The PROFIBUS operation mode is set to CLEAR.
Redundant State	This is redundant state of the PXM in HSBY mode. The following states can be returned: **Standalone** The PXM is operating as a Standalone DP Master and is connected to a non-HSBY M580 system. **Primary** The PXM is operating in a HSBY system and is currently the active PXM on the PROFIBUS network. **Standby** The PXM is operating in a HSBY system and is currently the standby PXM on the PROFIBUS network.

Duplicate Station	Indicates whether or not the PXM has detected another PROFIBUS DP station with the same station address as itself and has entered a temporary Back-Off mode. **None** No duplicate detected, normal operation. **Detected – Back-Off** Duplicate PROFIBUS station detected. Module has entered Back-Off mode. In this condition the PXM will not communicate on the PROFIBUS DP network. Although the back-off time is approximately 5 seconds, should the conflicting DP master remain active on the PROFIBUS network, the PXM will continuously re-enter the back-off mode. Because the PXM will never interrogate a slave device with the same station address as itself, this duplicate detection would be triggered only by the addition of another DP Master on the PROFIBUS network. The duplicate detection and subsequent invoking of the Back-off mode would occur if either the additional DP master has the same station address as the PXM, or it is interrogating another slave device with the same station address as the PXM.
Display	The Display frame will replicate what is physically showing on the module LEDs.
Sys MAC Address	Displays the module's unique Ethernet MAC address that has been assigned.
HW MAC Address	Displays the module's unique Ethernet MAC address that is fixed for the specific PCB.
Firmware Revision	The current firmware revision of the PXM.
Temperature	The internal temperature of the module.
Processor Scan	The amount of time (microseconds) taken by the module's processor in the last scan.
Up Time	Indicates the elapsed time since the module was powered-up.
Config Retrieval Status	The status of the last configuration retrieval from the M580 Head module.
Config Storage Status	The status of the last configuration sending to the M580 Head module.
DIP Switch Position	The status of the DIP switches when the module booted. Note that this status will not change if the DIP switches are altered when the module is running.

Table 8.4 - Parameters displayed in the Status Monitoring – General Tab

8.2.1.2. MASTER STATUS

The Master Status tab displays the following general parameters:

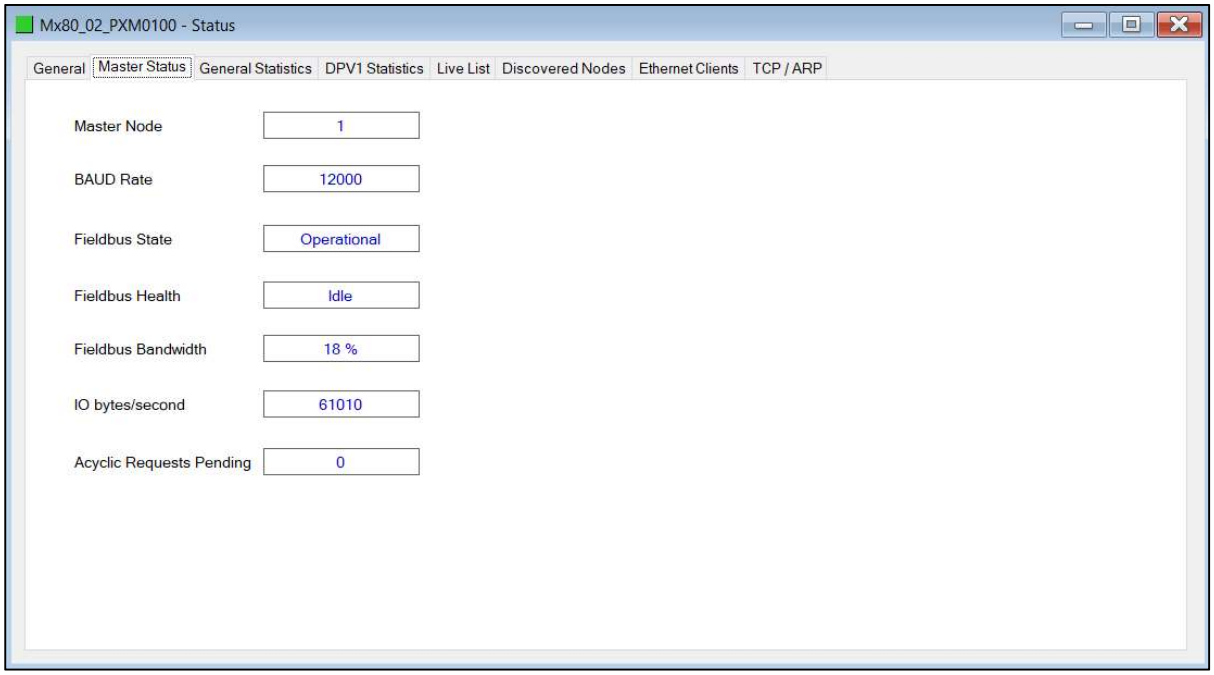


Figure 8.5 – PXM Status monitoring – Master Status

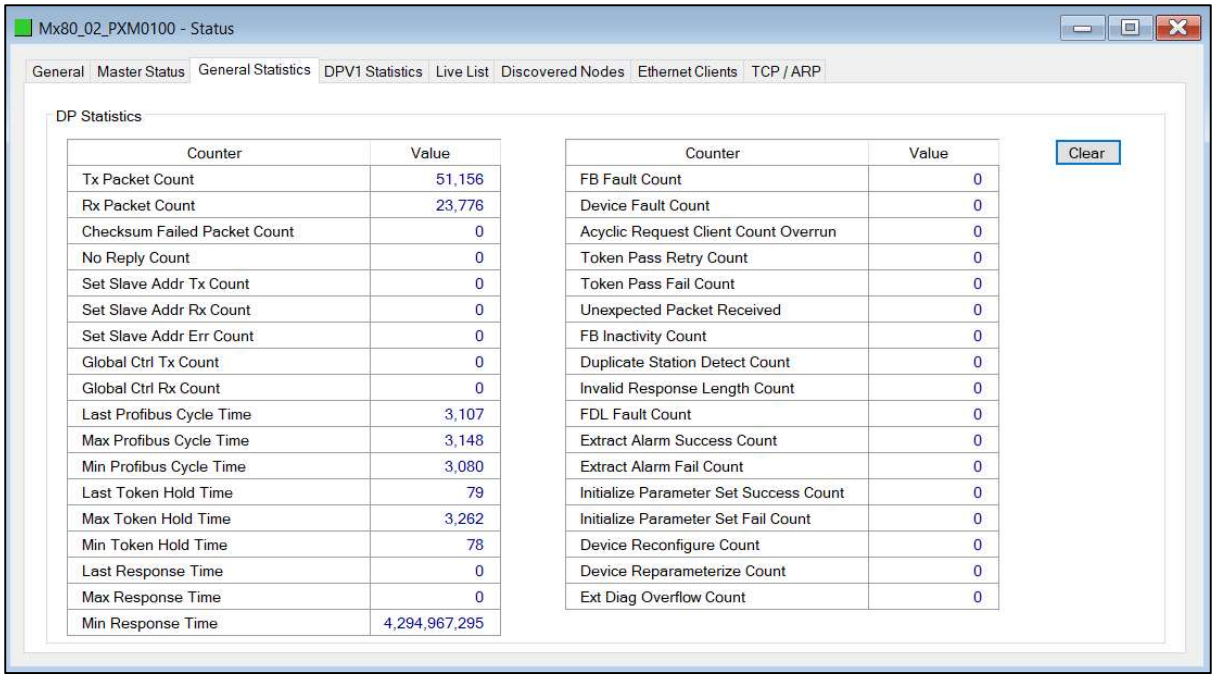
Parameter	Description
Master Node	The PROFIBUS Station address of the local PXM.
BAUD Rate	The BAUD Rate of the PROFIBUS network.
Fieldbus State	The state of the PROFIBUS network: **OFFLINE** The PROFIBUS network is offline and the PXM will not communicate on the network. **STOP** The PROFIBUS network is running and the PXM is communicating on the network, but it will not exchange any process data with any slave device. **OPERATE** The PROFIBUS network is running and the PXM is communicating with all slave devices on the network, and if configured in the PXM, the module will configure and exchange process data with each slave device. **CLEAR**

	The PROFIBUS network is running and the PXM is communicating with all slave devices on the network, and if configured in the PXM, the module will configure and exchange process data with each slave device. **NOTE**: In CLEAR mode the PXM will not send any output data to any slave device.
Fieldbus Health	The health/status of the PROFIBUS network: **Idle** There are not errors on the PROFIBUS network. **Fieldbus Error** There is a PROFIBUS network issues (e.g. cable unplugged, under/over terminated, etc.). **Device Error** At least one slave device has a communication issue (e.g. offline, not exchanging process data, etc.)
Fieldbus Bandwidth	The current load on the PROFIBUS network communication.
IO bytes/second	The number of process variable bytes being exchanged between the PXM and slave devices every second.
Acyclic Requests Pending	The number of acyclic requests (DPV1 Class 1 and Class 2 requests) pending.

Table 8.5 - Parameters displayed in the Status Monitoring – Master Status Tab

8.2.1.3. GENERAL STATISTICS

The General Statistics tab displays the following general parameters:



Counter	Value
Tx Packet Count	51,156
Rx Packet Count	23,776
Checksum Failed Packet Count	0
No Reply Count	0
Set Slave Addr Tx Count	0
Set Slave Addr Rx Count	0
Set Slave Addr Err Count	0
Global Ctrl Tx Count	0
Global Ctrl Rx Count	0
Last Profibus Cycle Time	3,107
Max Profibus Cycle Time	3,148
Min Profibus Cycle Time	3,080
Last Token Hold Time	79
Max Token Hold Time	3,262
Min Token Hold Time	78
Last Response Time	0
Max Response Time	0
Min Response Time	4,294,967,295

Counter	Value
FB Fault Count	0
Device Fault Count	0
Acyclic Request Client Count Overrun	0
Token Pass Retry Count	0
Token Pass Fail Count	0
Unexpected Packet Received	0
FB Inactivity Count	0
Duplicate Station Detect Count	0
Invalid Response Length Count	0
FDL Fault Count	0
Extract Alarm Success Count	0
Extract Alarm Fail Count	0
Initialize Parameter Set Success Count	0
Initialize Parameter Set Fail Count	0
Device Reconfigure Count	0
Device Reparameterize Count	0
Ext Diag Overflow Count	0

Figure 8.6 – PXM Status monitoring – General Statistics

Parameter	Description
Tx Packet Count	The number of PROFIBUS packets transmitted.
Rx Packet Count	The number of PROFIBUS packets Received.
Checksum Failed Packet Count	The number of PROFIBUS packets that had a failed checksum.
No Reply Count	The number of PROFIBUS requests from the PXM where the station did not respond.
Set Slave Addr Tx Count	The number of PROFIBUS Set Slave Address requests sent from the PXM.
Set Slave Addr Rx Count	The number of successful PROFIBUS Set Slave Address responses received from the specific slave device.
Set Slave Addr Err Count	The number of failed PROFIBUS Set Slave Address responses received from the specific slave device.
Global Ctrl Tx Count	The number of PROFIBUS Global Control requests sent from the PXM.
Global Ctrl Rx Count	The number of PROFIBUS Global Control requests received by the PXM.
Last PROFIBUS Cycle Time	The time (in microseconds) the last PROFIBUS Cycle took to complete.
Max PROFIBUS Cycle Time	The maximum time (in microseconds) the PROFIBUS Cycle took to complete.

Min PROFIBUS Cycle Time	The minimum time (in microseconds) the PROFIBUS Cycle took to complete.
Last Token Hold Time	The time (in microseconds) the PXM held the token in the last token rotation.
Max Token Hold Time	The maximum time (in microseconds) the PXM held the token.
Min Token Hold Time	The minimum time (in microseconds) the PXM held the token.
Last Response Time	In a Multi DP Master system, this is the time it took (in microseconds) to respond to the last token passed from another DP Master.
Max Response Time	In a Multi DP Master system, this is the maximum time it took (in microseconds) to respond to a token passed from another DP Master.
Min Response Time	In a Multi DP Master system, this is the minimum time it took (in microseconds) to respond to a token passed from another DP Master.
FB Fault Count	The number of fieldbus faults that have occurred (e.g. devices going offline, corrupted packets, etc.)
Device Fault Count	The number of slave device faults that have occurred (e.g. device stops communicating during data exchange).
Acyclic Request Client Count Overrun	The number of times more than 10 acyclic requests needed to be buffered in which case the PXM will reject the 11^{th} request.
Token Pass Retry Count	In a Multi DP Master system, this is the number of times the token pass from the PXM had to be retransmitted because the receiving DP Master did not respond in time.
Token Pass Fail Count	When the number of consecutive Token Pass Retries reaches the configured token pass retry count after which that DP Master will be assumed as offline.
Unexpected Packet Received	The number of times a response is received from a slave device that was not expected (e.g. incorrect response, response from a different node, etc.).
FB Inactivity Count	The number of times the PXM has determine that there are no other DP Masters on the PROFIBUS network.
Duplicate Station Detect Count	The number of times the PXM has detected that there is another station on the network with the same station address as the local PXM.
Invalid Response Length Count	The number of times a response is received from a slave device where the length is not correct (for example if the slave device is configured to provide 10 bytes of process data and only 5 bytes are returned during data exchange).
FDL Fault Count	The number of Data Link Layer function code faults received.

Extract Alarm Success Count	The number of alarms that have successfully been extracted from slave devices.
Extract Alarm Fail Count	The number of alarms that have **not** successfully been extracted from slave devices.
Initialize Parameter Set Success Count	The number of parameters that have successfully been set after the device has been configured for data exchange.
Initialize Parameter Set Fail Count	The number of parameters that have failed to set after the device has been configured for DPV0 data exchange.
Device Reconfigure Count	The number of times a slave device has been (re)configured for DPV0 data exchange.
Device Reparameterize Count	The number of times a slave device has been (re)parameterized for DPV0 data exchange.
Ext Diag Overflow Count	The number of times a slave device has returned diagnostics data that could not fit into a single PROFIBUS frame.

Table 8.6 - Parameters displayed in the Status Monitoring – General Statistics Tab

8.2.1.4. DPV1 STATISTICS

The DPV1 Statistics tab displays the following general parameters:

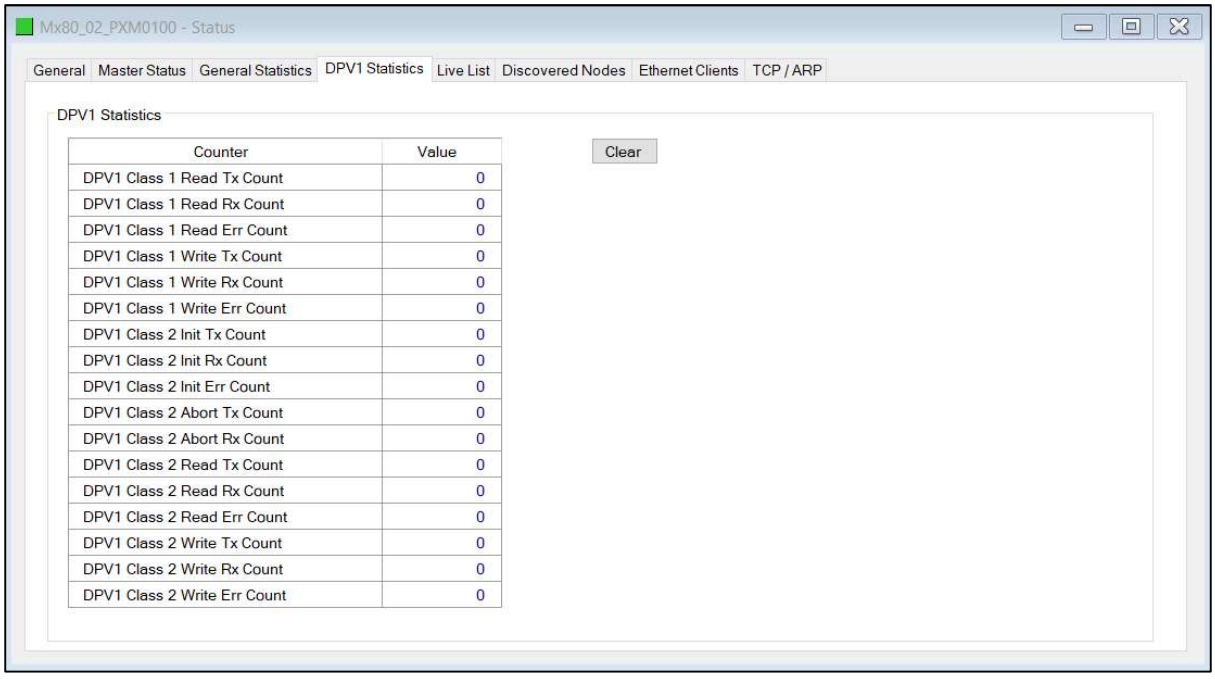


Figure 8.7 – PXM Status monitoring – DPV1 Statistics

Parameter	Description
DPV1 Class 1 Read Tx Count	The number of PROFIBUS DPV1 Class 1 Read requests sent from the PXM.
DPV1 Class 1 Read Rx Count	The number of successful PROFIBUS DPV1 Class 1 Read responses received by the PXM.
DPV1 Class 1 Read Err Count	The number of failed PROFIBUS DPV1 Class 1 Read responses received by the PXM.
DPV1 Class 1 Write Tx Count	The number of PROFIBUS DPV1 Class 1 Write requests sent from the PXM.
DPV1 Class 1 Write Rx Count	The number of successful PROFIBUS DPV1 Class 1 Write responses received by the PXM.
DPV1 Class 1 Write Err Count	The number of failed PROFIBUS DPV1 Class 1 Write responses received by the PXM.
DPV1 Class 2 Init Tx Count	The number of PROFIBUS DPV1 Class 2 Initialize requests sent from the PXM.
DPV1 Class 2 Init Rx Count	The number of successful PROFIBUS DPV1 Class 2 Initialize responses by the PXM.
DPV1 Class 2 Init Err Count	The number of failed PROFIBUS DPV1 Class 2 Initialize responses received by the PXM.
DPV1 Class 2 Abort Tx Count	The number of PROFIBUS DPV1 Class 2 Abort requests sent from the PXM.
DPV1 Class 2 Abort Rx Count	The number of PROFIBUS DPV1 Class 2 Abort messages received by the PXM.
DPV1 Class 2 Read Tx Count	The number of PROFIBUS DPV1 Class 2 Read requests sent from the PXM.
DPV1 Class 2 Read Rx Count	The number of successful PROFIBUS DPV1 Class 2 Read responses received by the PXM
DPV1 Class 2 Read Err Count	The number of failed PROFIBUS DPV1 Class 2 Read responses received by the PXM.
DPV1 Class 2 Write Tx Count	The number of PROFIBUS DPV1 Class 2 Write requests sent from the PXM.
DPV1 Class 2 Write Rx Count	The number of successful PROFIBUS DPV1 Class 2 Write responses received by the PXM.
DPV1 Class 2 Write Err Count	The number of failed PROFIBUS DPV1 Class 2 Write responses received by the PXM.

Table 8.7 - Parameters displayed in the Status Monitoring – DPV1 Statistics Tab

8.2.1.5. LIVE LIST

The Live List tab in the PXM status monitoring provide the user with an overview of all slave devices and DP masters connected to the PROFIBUS network. Each station will be in one of six states that are provided in the Live List page.

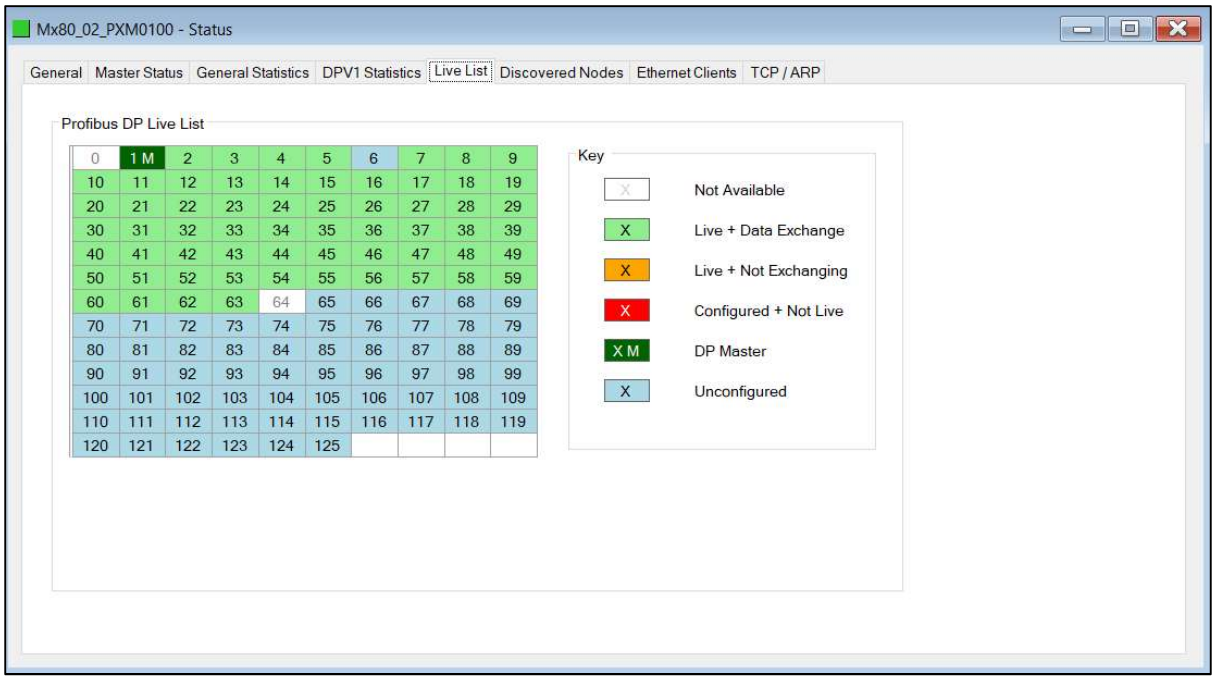


Figure 8.8 – PXM Status monitoring – Live List

8.2.1.6. DISCOVERED NODES

The Discovered Nodes status page provides the user with more detail regarding each station on the PROFIBUS network (when compared to the live list). The user can scan the PROFIBUS network to extract further details from each device. From here the user can add the slave device or change the slave device station address. See the *Device Discovery* section.

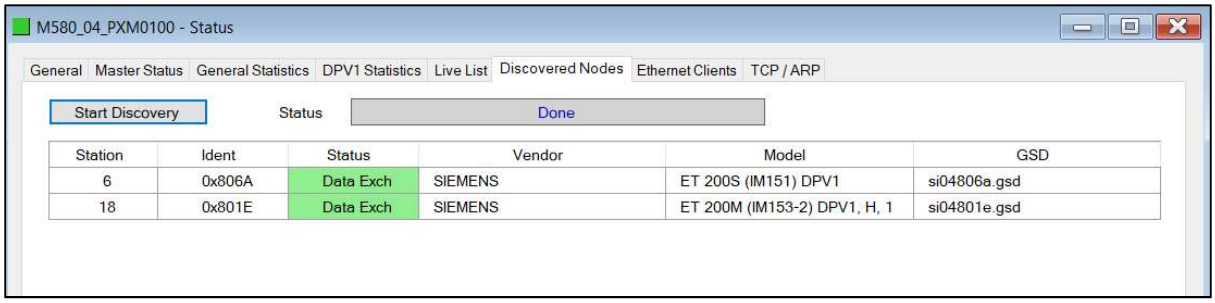


Figure 8.9 – PXM Status monitoring – Discovered Nodes

8.2.1.7. ETHERNET/IP STATISTICS

The EtherNet/IP Statistics tab displays the following parameters:

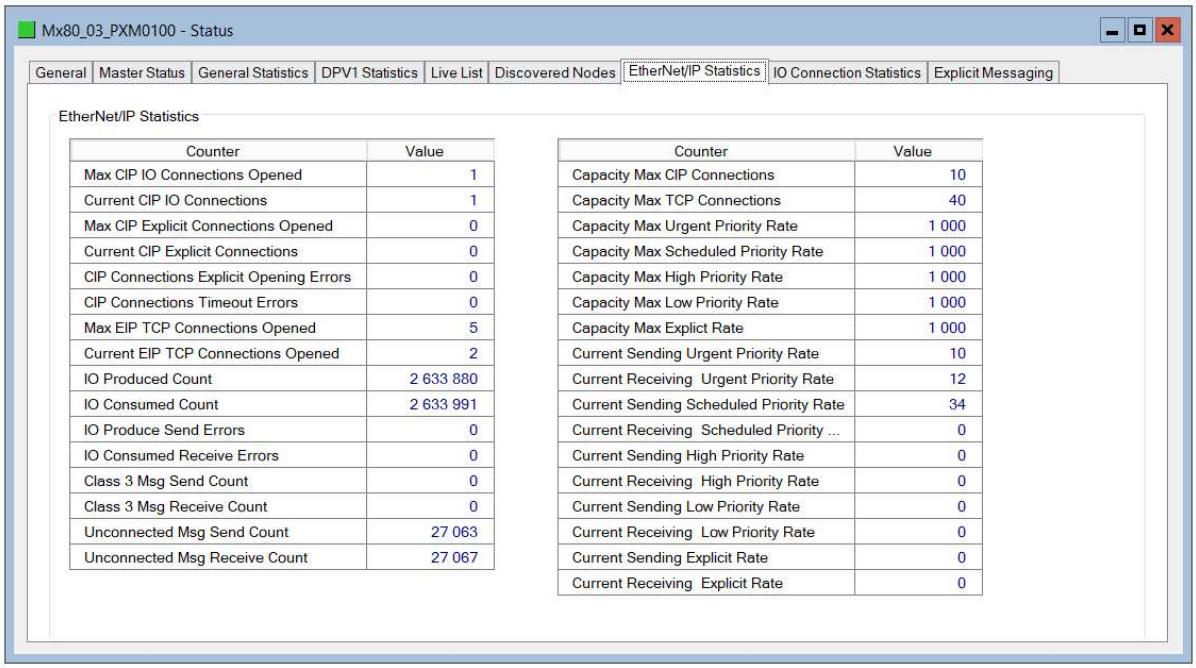


Figure 8.10 – PXM Status monitoring – EtherNet/IP Statistics

Parameter	Description
Max CIP IO Connections Opened	The maximum number of CIP Class 1 connections opened to the module.
Current CIP IO Connections	The number of current CIP IO Class 1 connections opened to the module.
Max CIP Explicit Connections Opened	The maximum number of CIP Class 3 connections opened to the module.
Current CIP Explicit Connections	The number of current CIP IO Class 3 connections opened to the module.
CIP Connections Explicit Opening Errors	The number of Class 3 connections rejected.
CIP Connections Timeout Errors	The number of CIP connections that have timed-out.
Max EIP TCP Connections Opened	The maximum number of TCP connections opened.
Current EIP TCP Connections Opened	The number of current TCP connections open.

IO Produced Count	The number of implicit Produced transactions sent.
IO Consumed Count	The number of implicit Consumed transactions received.
IO Produce Send Errors	The number of implicit Produced errors.
IO Consumed Receive Errors	The number of implicit Consumed errors.
Class 3 Msg Send Count	The number of Class 3 messages sent.
Class 3 Msg Receive Count	The number of Class 3 messages received.
Unconnected Msg Send Count	The number of Unconnected (UCCM) messages sent.
Unconnected Msg Receive Count	The number of Unconnected (UCCM) messages received.
Capacity Max CIP Connections	The maximum number of CIP connections supported by the module.
Capacity Max TCP Connections	The maximum number of TCP connections supported by the module.
Capacity Max Urgent Priority Rate	The maximum number of Urgent Priority messages per second supported by the module.
Capacity Max Scheduled Priority Rate	The maximum number of Scheduled Priority messages per second supported by the module.
Capacity Max High Priority Rate	The maximum number of High Priority messages per second supported by the module.
Capacity Max Low Priority Rate	The maximum number of Low Priority messages per second supported by the module.
Capacity Max Explicit Rate	The maximum number of Explicit messages per second supported by the module.
Current Sending Urgent Priority Rate	The current number of Urgent Priority messages per second sent by the module.
Current Receiving Urgent Priority Rate	The current number of Urgent Priority messages per second received by the module.
Current Sending Scheduled Priority Rate	The current number of Scheduled Priority messages per second sent by the module.
Current Receiving Scheduled Priority Rate	The current number of Scheduled Priority messages per second received by the module.
Current Sending High Priority Rate	The current number of High Priority messages per second sent by the module.
Current Receiving High Priority Rate	The current number of High Priority messages per second received by the module.
Current Sending Low Priority Rate	The current number of Low Priority messages per second sent by the module.

Current Receiving Low Priority Rate	The current number of Low Priority messages per second received by the module.
Current Sending Explicit Rate	The current number of Explicit messages per second sent by the module.
Current Receiving Explicit Rate	The current number of Explicit messages per second received by the module.

Table 8.8 - Parameters displayed in the Status Monitoring – EtherNet/IP Statistics Tab

8.2.1.8. IO CONNECTION STATISTICS

The IO Connection Statistics tab displays the following parameters:

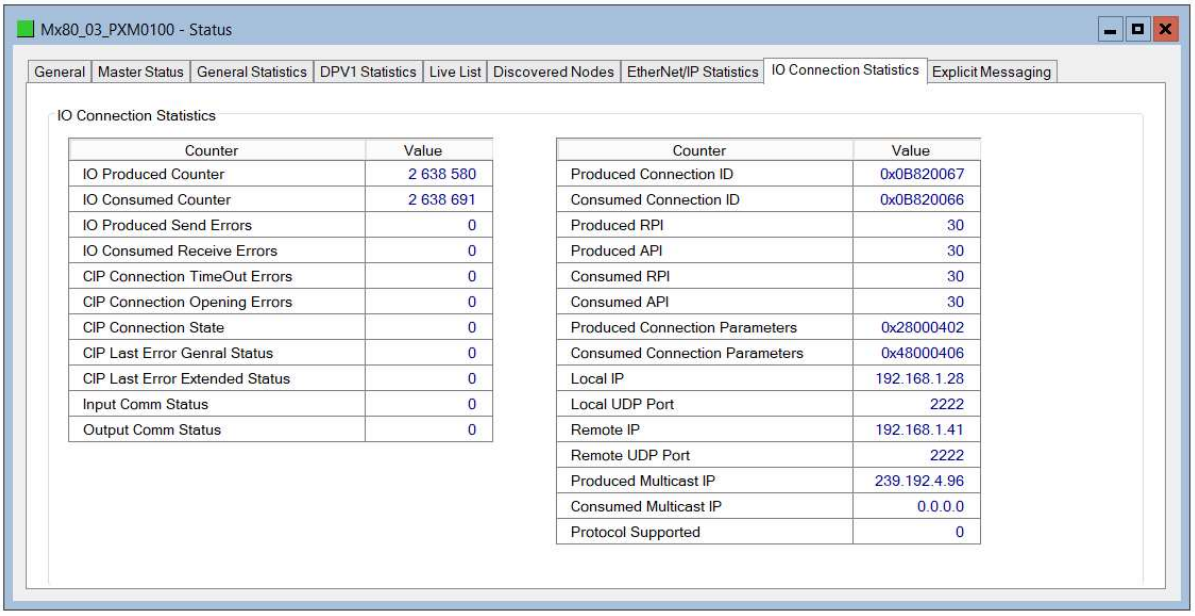


Figure 8.11 – PXM Status monitoring – IO Connection Statistics

Parameter	Description
IO Produced Counter	The number of implicit Produced transactions sent.
IO Consumed Counter	The number of implicit Consumed transactions received.
IO Produced Send Errors	The number of implicit Produced errors.
IO Consumed Receive Errors	The number of implicit Consumed errors.
CIP Connection Time Out Errors	The number of implicit time out errors.
CIP Connection Opening Errors	The number of implicit connections rejected.

CIP Connection State	Current CIP Connection state.
CIP Last Error General Status	General Status of last receive connection failure.
CIP Last Error Extended Status	Extended status of last received connection failure.
Input Comm Status	Input communication status.
Output Comm Status	Output communication status.
Produced Connection ID	Primary Produced Connection ID.
Consumed Connection ID	Primary Consumed Connection ID.
Produced RPI	Primary Produced connection Requested Packet Interval (ms)
Produced API	Primary Produced connection Actual Packet Interval (ms)
Consumed RPI	Primary Consumed connection Requested Packet Interval (ms)
Consumed API	Primary Consumed connection Actual Packet Interval (ms)
Produced Connection Parameters	Produced Connection Parameters
Consumed Connection Parameters	Consumed Connection Parameters
Local IP	Local (module) IP address
Local UDP Port	Local (module) UDP port
Remote IP	Remote (PLC) IP address
Remote UDP Port	Remote (PLC) UDP port
Produced Multicast IP	Produced Multicast IP address
Consumed Multicast IP	Consumed Multicast IP address
Protocol Supported	Additional protocols supported.

Table 8.9 - Parameters displayed in the Status Monitoring – IO Connection Statistics Tab

8.2.1.9. EXPLICIT MESSAGING STATISTICS

The Explicit Messaging tab displays the following parameters:

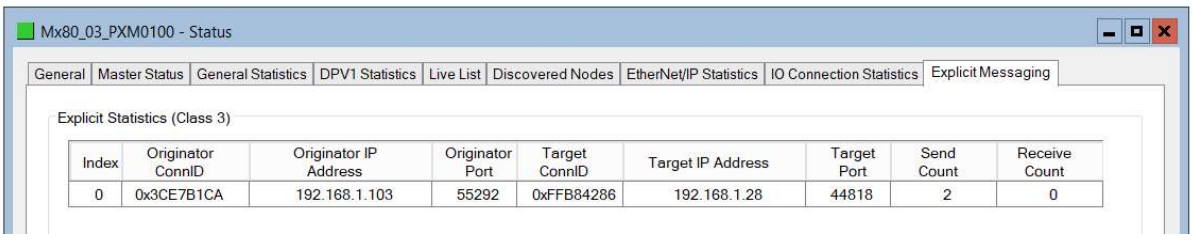


Figure 8.12 – PXM Status monitoring – Explicit Messaging Statistics

Parameter	Description
Index	Explicit Connection index.
Originator Conn ID	Originator CIP Connection Identifier.
Originator IP Address	Originator IP address.
Originator Port	Originator TCP/UDP port.
Target Conn ID	Target CIP Connection Identifier.
Target IP Address	Target IP address.
Target Port	Target TCP/UDP port.
Send Count	Number of explicit messages sent.
Receive Count	Number of explicit messages received.

Table 8.10 - Parameters displayed in the Status Monitoring – Explicit Messaging Tab

8.2.2. DEVICE STATUS

The Status monitoring window of each PROFIBUS slave device connected to the PXM can be opened by right-clicking on the specific slave device in the ProSoft Configurator for Modicon tree and selecting *Status*.

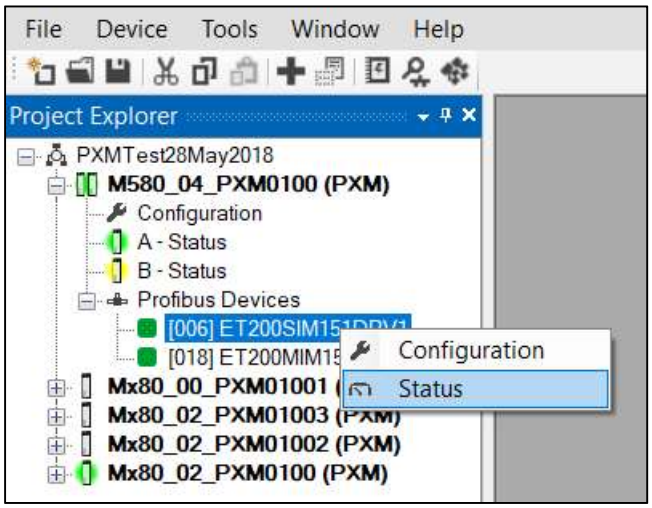


Figure 8.13 - Selecting slave device online Status

The device status window contains multiple tabs to display the current status of the specific slave device.

8.2.2.1. GENERAL

The General tab displays the following general parameters:

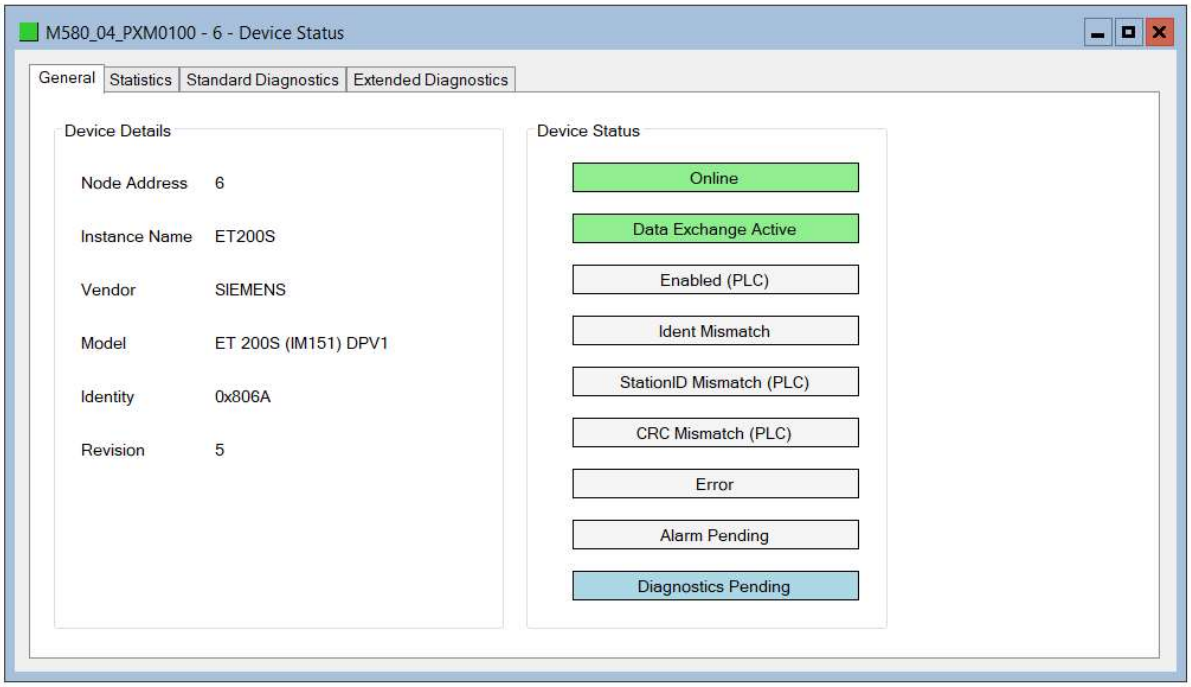


Figure 8.14 – Device Status monitoring - General

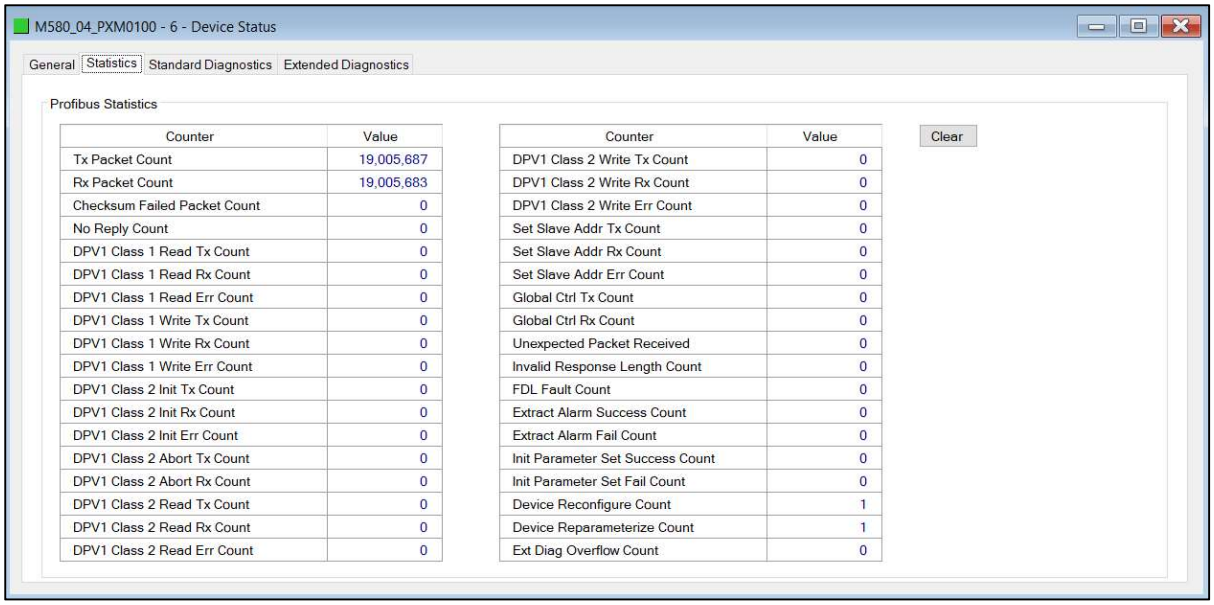
Parameter	Description
Node Address	The selected slave device station address
Instance Name	The configured instance name of the device
Vendor	The device Vendor name.
Model	The device Mode name.
Identity	The device PNO identity.
Revision	The device revision.
Device Status	The current status of the device: **Online** The slave device is online. **Data Exchange Active** The slave device is exchanging DPV0 process data with the PXM. **Disabled (PLC)** The slave device has been disabled from DPV0 data exchange from the M580 controller using the PXM output assembly.

	Identity Mismatch The device configured in the ProSoft Configurator for Modicon and the device online at the specific station address do not match. **StationID Mismatch (PLC)** The station address entered from the M580 controller using the PXM output assembly does not match the station address of the configured slave device. **CRC Mismatch (PLC)** Indicates the mapping from the M580 controller does not match the configured mapping. **Error** Device Error flag. The error flag will be set when one of the following conditions occur: • If there is an ident mismatch during slave parameterization, • When receiving any form of FDL fault (data link layer fault). For example: SAP Not Activated or Resource Not Available. • When the data size of the DPV0 data exchange does not match what has been configured in the PCM. This Error flag is transient and will clear once a valid response is received. **Alarm Pending** An alarm is pending in the specific slave device. **Diagnostics Pending** There is new diagnostics pending in the slave device.

Table 8.11 - Device Status Monitoring – General Tab

8.2.2.2. STATISTICS

The Statistics tab displays the following general parameters:



Counter	Value
Tx Packet Count	19,005,687
Rx Packet Count	19,005,683
Checksum Failed Packet Count	0
No Reply Count	0
DPV1 Class 1 Read Tx Count	0
DPV1 Class 1 Read Rx Count	0
DPV1 Class 1 Read Err Count	0
DPV1 Class 1 Write Tx Count	0
DPV1 Class 1 Write Rx Count	0
DPV1 Class 1 Write Err Count	0
DPV1 Class 2 Init Tx Count	0
DPV1 Class 2 Init Rx Count	0
DPV1 Class 2 Init Err Count	0
DPV1 Class 2 Abort Tx Count	0
DPV1 Class 2 Abort Rx Count	0
DPV1 Class 2 Read Tx Count	0
DPV1 Class 2 Read Rx Count	0
DPV1 Class 2 Read Err Count	0

Counter	Value
DPV1 Class 2 Write Tx Count	0
DPV1 Class 2 Write Rx Count	0
DPV1 Class 2 Write Err Count	0
Set Slave Addr Tx Count	0
Set Slave Addr Rx Count	0
Set Slave Addr Err Count	0
Global Ctrl Tx Count	0
Global Ctrl Rx Count	0
Unexpected Packet Received	0
Invalid Response Length Count	0
FDL Fault Count	0
Extract Alarm Success Count	0
Extract Alarm Fail Count	0
Init Parameter Set Success Count	0
Init Parameter Set Fail Count	0
Device Reconfigure Count	1
Device Reparameterize Count	1
Ext Diag Overflow Count	0

Figure 8.15 – Device Status monitoring - Statistics

Parameter	Description
Tx Packet Count	The number of PROFIBUS packets transmitted.
Rx Packet Count	The number of PROFIBUS packets Received.
Checksum Failed Packet Count	The number of PROFIBUS packets that had a failed checksum.
No Reply Count	The number of PROFIBUS requests from the PXM where the station did not respond.
DPV1 Class 1 Read Tx Count	The number of PROFIBUS DPV1 Class 1 Read requests sent from the PXM to the specific device.
DPV1 Class 1 Read Rx Count	The number of successful PROFIBUS DPV1 Class 1 Read responses received from the specific device.
DPV1 Class 1 Read Err Count	The number of failed PROFIBUS DPV1 Class 1 Read responses received from the specific device.
DPV1 Class 1 Write Tx Count	The number of PROFIBUS DPV1 Class 1 Write requests sent from the PXM to the specific device.
DPV1 Class 1 Write Rx Count	The number of successful PROFIBUS DPV1 Class 1 Write responses received from the specific device.
DPV1 Class 1 Write Err Count	The number of failed PROFIBUS DPV1 Class 1 Write responses received from the specific device.
DPV1 Class 2 Init Tx Count	The number of PROFIBUS DPV1 Class 2 Initialize requests sent from the PXM to the specific device.
DPV1 Class 2 Init Rx Count	The number of successful PROFIBUS DPV1 Class 2 Initialize responses received from the specific device.

DPV1 Class 2 Init Err Count	The number of failed PROFIBUS DPV1 Class 2 Initialize responses received from the specific device.
DPV1 Class 2 Abort Tx Count	The number of PROFIBUS DPV1 Class 2 Abort requests sent from the PXM to the specific device.
DPV1 Class 2 Abort Rx Count	The number of PROFIBUS DPV1 Class 2 Abort messages received from the specific device.
DPV1 Class 2 Read Tx Count	The number of PROFIBUS DPV1 Class 2 Read requests sent from the PXM to the specific device.
DPV1 Class 2 Read Rx Count	The number of successful PROFIBUS DPV1 Class 2 Read responses received from the specific device.
DPV1 Class 2 Read Err Count	The number of failed PROFIBUS DPV1 Class 2 Read responses received from the specific device.
DPV1 Class 2 Write Tx Count	The number of PROFIBUS DPV1 Class 2 Write requests sent from the PXM to the specific device.
DPV1 Class 2 Write Rx Count	The number of successful PROFIBUS DPV1 Class 2 Write responses received from the specific device.
DPV1 Class 2 Write Err Count	The number of failed PROFIBUS DPV1 Class 2 Write responses received from the specific device.
Set Slave Addr Tx Count	The number of PROFIBUS Set Slave Address requests sent from the PXM to the specific device.
Set Slave Addr Rx Count	The number of successful PROFIBUS Set Slave Address responses received from the specific device.
Set Slave Addr Err Count	The number of failed PROFIBUS Set Slave Address responses received from the specific device.
Global Ctrl Tx Count	The number of PROFIBUS Global Control requests sent from the PXM to the specific device.
Global Ctrl Rx Count	The number of PROFIBUS Global Control requests received by the PXM from the specific device.
Unexpected Packet Received	The number of times a response is received from the device that was not expected (e.g. incorrect response, response from a different node, etc.).
Invalid Response Length Count	The number of times a response is received from the device where the length is not correct (for example if the device is configured to provide 10 bytes of process data and only 5 bytes are returned during data exchange).
FDL Fault Count	The number of Data Link Layer function code faults received from the specific device.
Extract Alarm Success Count	The number of alarms that have successfully been extracted from the specific device.

Extract Alarm Fail Count	The number of alarms that have **not** successfully been extracted from the specific device.
Initialize Parameter Set Success Count	The number of parameters that have successfully been set after the device has been configured for data exchange for the specific device.
Initialize Parameter Set Fail Count	The number of parameters that have failed to set after the device has been configured for DPV0 data exchange for the specific device.
Device Reconfigure Count	The number of times the device has been (re)configured for DPV0 data exchange.
Device Reparameterize Count	The number of times the device has been (re)parameterized for DPV0 data exchange.
Ext Diag Overflow Count	The number of times the device has returned diagnostics data that could not fit into a single PROFIBUS frame.

Table 8.12 - device Status Monitoring – Statistics Tab

8.2.2.3. STANDARD DIAGNOSTICS

The Standard Diagnostics tab displays the following general parameters:

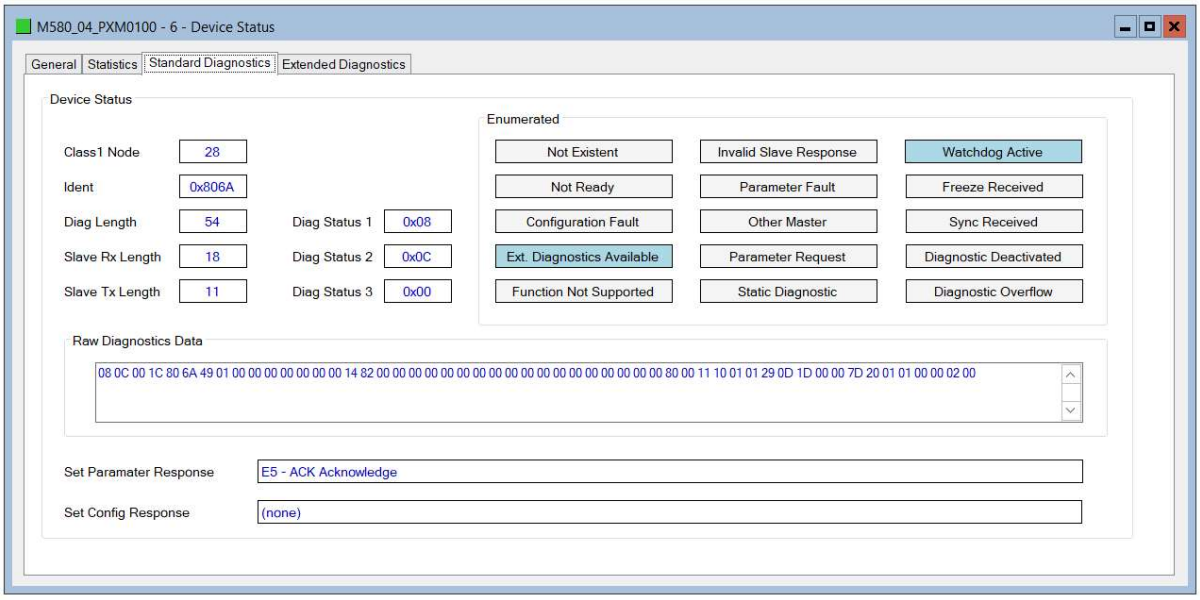


Figure 8.16 – Device Status monitoring – Standard Diagnostics

The diagnostic information displayed represents that which was received from the last Diagnostic read from the device and is therefore not updated continuously. A refresh can be forced by using the Explicit Messaging Utility.

Parameter	Description
Class 1 Node	The station address of the DP Master that configured the specific device for DPV0 communication.
Ident	The PNO Identification number of the device on the PROFIBUS network.
Slave Rx Length	The number of process data (DPV0) bytes expected from the device.
Slave Tx Length	The number of process data (DPV0) bytes that will be sent to the device.
Diag Status 1	The raw diagnostic byte 1.
Diag Status 2	The raw diagnostic byte 2.
Diag Status 3	The raw diagnostic byte 3.
Raw Diagnostics Data	The raw diagnostics in a hexadecimal data string.
Set Parameter Response	The response received after the last Set Parameter command.
Set Config Response	The response received after the last Set Config command.
Enumerated	
Not Existent	Set by the master if slave does not respond.
Not Ready	Slave not ready for data exchange.
Configuration Fault	Slave did not accept last configuration data.
Ext. Diagnostics Available	Slave has extended diagnostic information to report.
Function Not supported	Slave does not support the required function.
Invalid Slave Response	Set by master if slave response is invalid.
Parameter Fault	Slave rejected last parameter frame.
Other Master	Set by master if slave is being parameterized by another master.
Parameter Request	Slave requires parameterization.
Static Diagnostic	Set by slave to cause master to request diagnostic information.
Watchdog Active	Slave watchdog is active.
Freeze Received	Slave has received a Freeze command.
Sync Received	Slaves has received a Sync command.
Diagnostic Deactivated	Set by master if slave has been flagged as inactive.
Diagnostics Overflow	Set if there is more diagnostic information than specified in the last diagnostic retrieval.

Table 8.13 - Device Status Monitoring – Standard Diagnostics Tab

8.2.2.4. EXTENDED DIAGNOSTICS

The Extended Diagnostics are decoded and displayed in a table form. The diagnostics are decoded using the pre-configured GSD file.

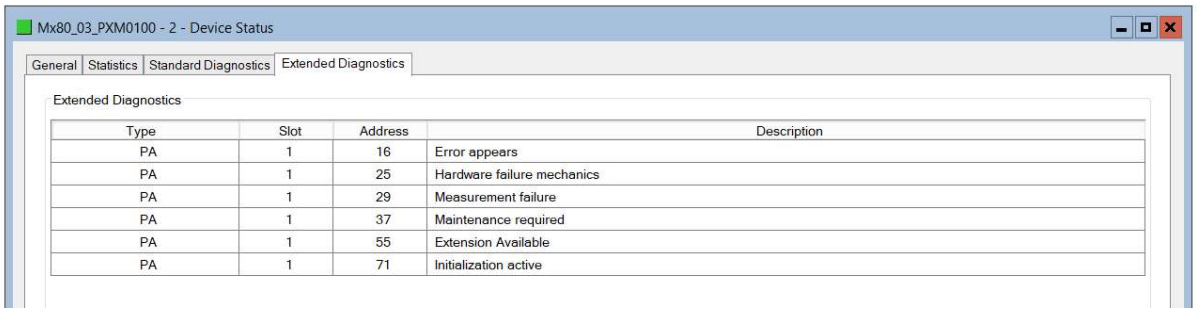


Figure 8.17 – Device Status monitoring – Extended Diagnostics

8.3. PROFIBUS PACKET CAPTURE

The module provides the capability to capture the PROFIBUS traffic for analysis. The will allow the user and the support team to view the packet stream. To invoke the capture of the module, right-click on either the PXM or the ***PROFIBUS Device*** item in the Project Explorer and select the ***DP Packet Capture*** option.

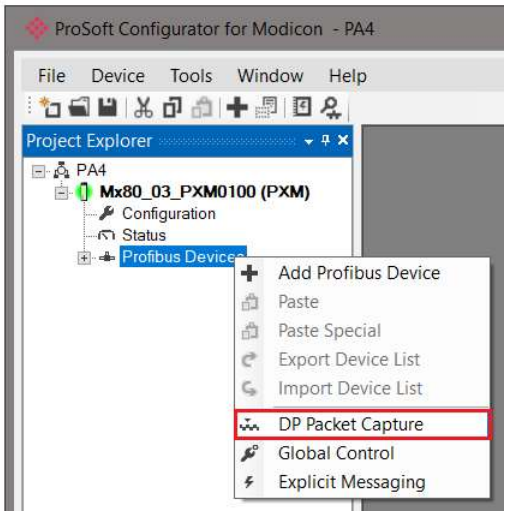


Figure 8.18 - Selecting PROFIBUS Packet Capture

The DP Packet Capture window will open and automatically start capturing all PROFIBUS packets.

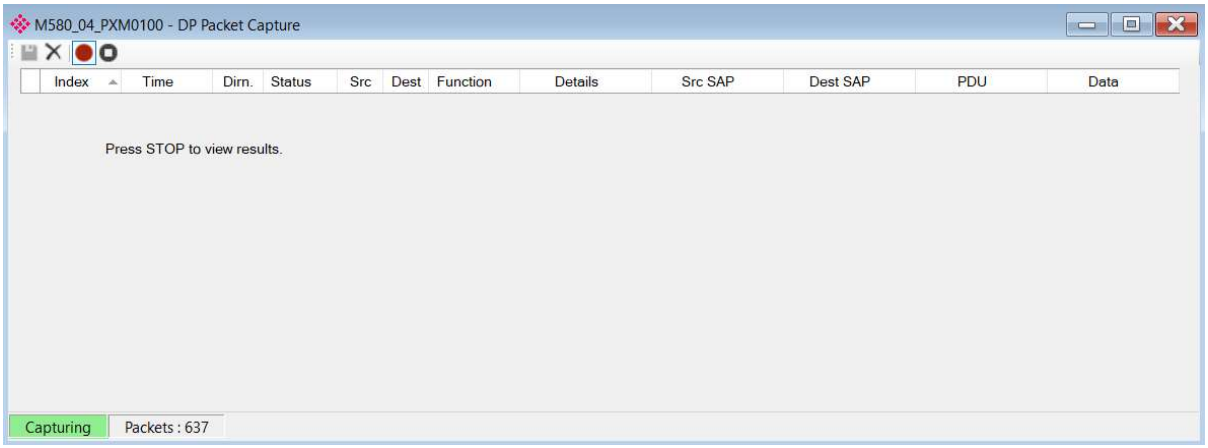


Figure 8.19 - PROFIBUS packet capture

The module will capture packets until the user presses Stop or when 10,000 DP packets have been reached.

When the capture process is stopped then the PROFIBUS capture will be presented as shown below.

Figure 8.20 - PROFIBUS Packet Capture complete

The captured PROFIBUS packets are tabulated as follows:

Statistic	Description
Index	The packet index incremented for each packet sent or received.

Time	The time is measured in microseconds (us) and is started at a fraction of a second and continued until the packet capture is done.
Dirn.	The direction of the packet, either transmitted (Tx) or received (Rx).
Status	The status of the packet. Received packets are checked for valid PROFIBUS constructs and valid checksums.
Src	PROFIBUS node address of the message source.
Dest	PROFIBUS node address of the message destination.
Function	The PROFIBUS function (e.g. Token, Request, etc.)
Details	Additional details associated with the PROFIBUS command/function.
Src SAP	The source Service Access Point (SAP) when used.
Dest SAP	The destination Service Access Point (SAP) when used.
PDU	The PROFIBUS packet payload.
Data	The packet's raw data displayed in space delimited hex.

Table 8.14 - PROFIBUS Packet Capture fields

The packet capture can be saved to a file for further analysis, by selecting the Save button on the toolbar. Previously saved PROFIBUS Packet Capture files can be viewed by selecting the *PROFIBUS Packet Capture Viewer* option in the tools menu.

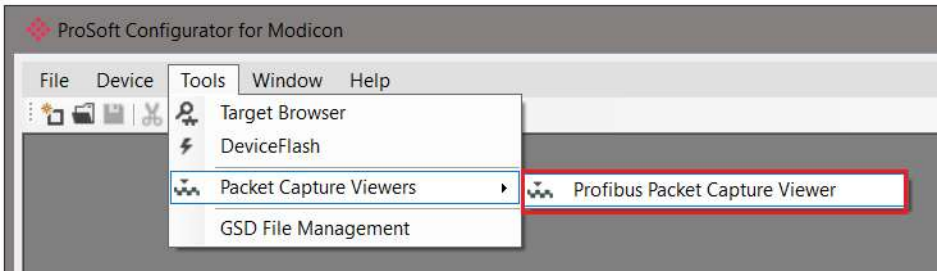


Figure 8.21 - Selecting the PROFIBUS Packet Capture Viewer

8.4. TARGET BROWSER

The ProSoft Configurator for Modicon includes a Target Browser which is able to scan the Ethernet network and display all EtherNet/IP devices with their IP addresses.

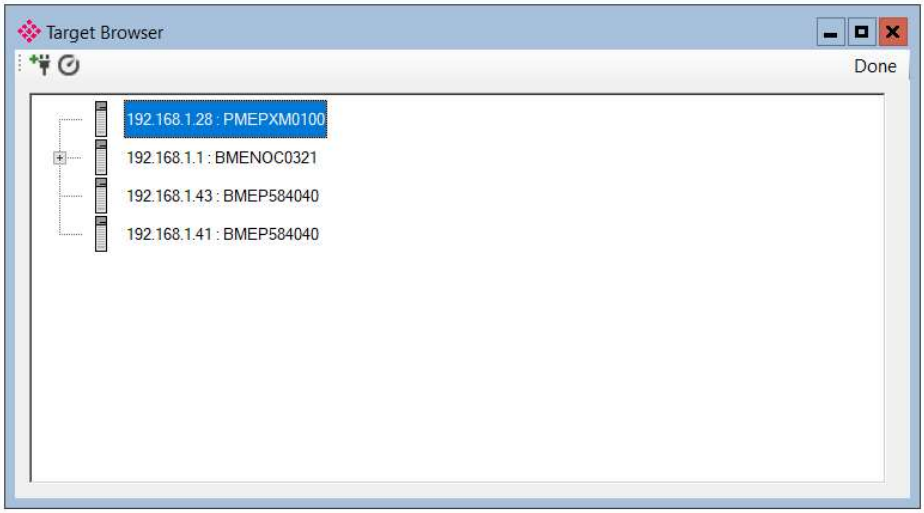


Figure 8.22 – Target Browser

Only devices that respond to broadcast EtherNet/IP commands are displayed. For this reason it is not possible to browse to other modules in a rack when connecting via a NOC module.

To select a device at a specific IP address, select the ***Scan IP Node*** icon on the top left toolbar and enter the required IP address.

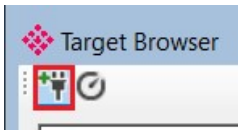


Figure 8.23 – Scan IP Node

Right-clicking on a device in the Target Browser provides a number of context menu items, including Displaying Properties, Port configuration and Reset commands depending on the type of device selected.

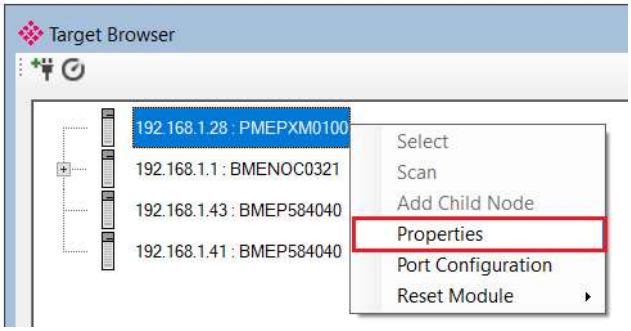


Figure 8.24 – Target Browser – Device context menu

The Device Properties window displays the common attributes of the CIP (ODVA) Identity object.

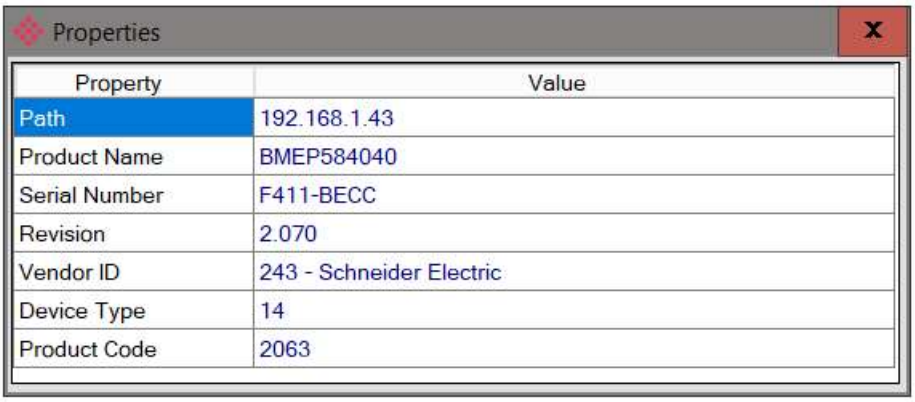


Figure 8.25 – Target Browser – Device Properties

The Port Configuration window can be used to display - and depending on the module type, modify the TCP/IP port settings.

The PXM module's port settings cannot be modified with this utility because the PXM must receive its port settings from the M580 head-module.

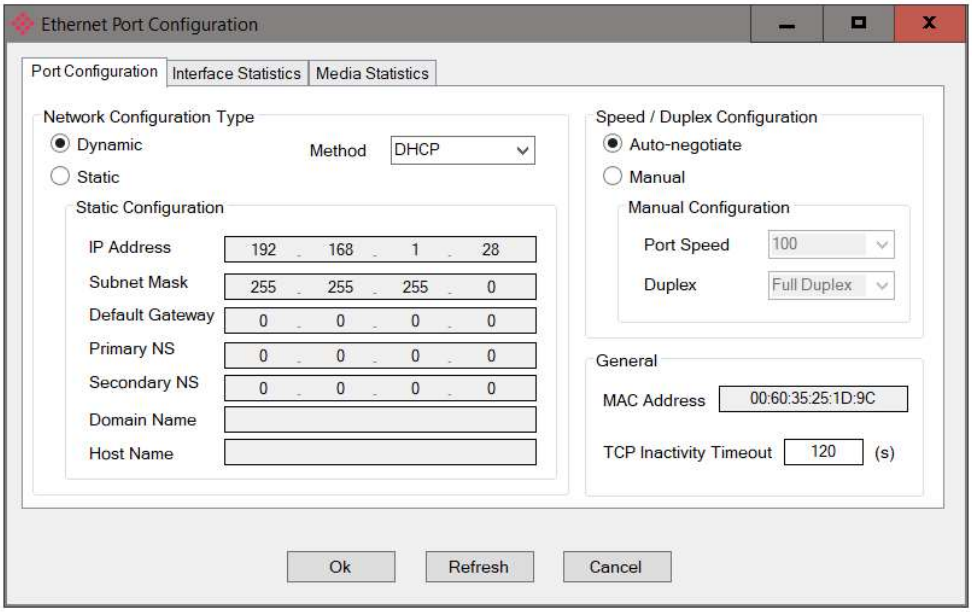


Figure 8.26 – Target Browser – Port Configuration

The Target Browser can also be used to reset an EtherNet/IP module. There are three different reset options:

- Power Cycle – Resets the module by emulating a power cycle
- Factory Defaults – Returns the module to the "out-of-box" configuration
- Factory Defaults Except Communication - Returns the module to the "out-of-box" configuration except for the IP address.

Not all EtherNet/IP modules support the various Reset options. Some options may also be rejected based on the module's current operating state.

⚠ CAUTION
DISRUPTION OF THE CONTROL SYSTEM Before selecting the Reset command on an operating module, ensure that it is safe to do so. **Failure to follow these instructions can result in injury or equipment damage.**

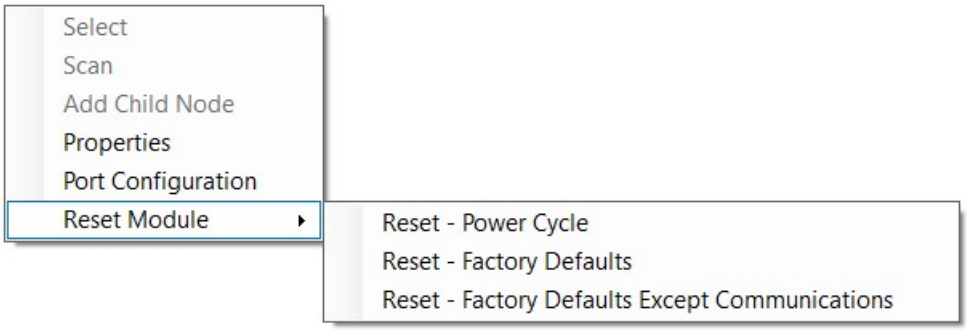


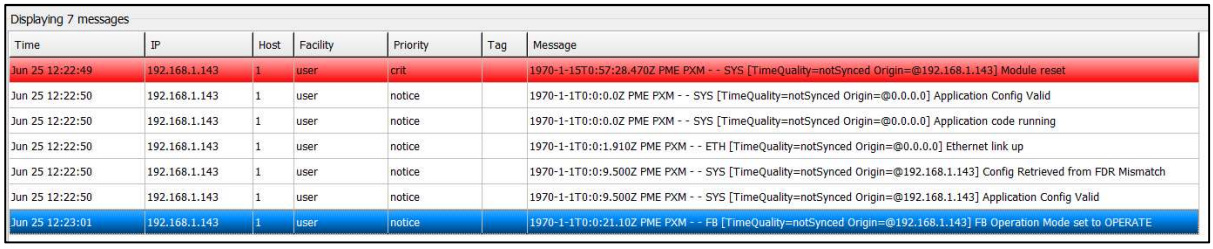
Figure 8.27 – Target Browser – Reset Options

8.5. MODULE EVENT LOG

The PXM can store up to 2048 events in non-volatile (NV) memory. These events can be unloaded to a SysLog Server by configuring the parameters required for the SysLog Server. See the *Security* section in the module configuration for details on how to configure the SysLog Server parameters.

The PXM will connect to the SysLog server using TCP port 601. This may require the SysLog server settings to be adjusted.

Once the PXM has successfully connected to the SysLog Server any buffered events in the module event log will be sent to the SysLog Server and unloaded from the PXM's NV memory.


Displaying 7 messages

Time	IP	Host	Facility	Priority	Tag	Message
Jun 25 12:22:49	192.168.1.143	1	user	crit		1970-1-15T0:57:28.470Z PME PXM - - SYS [TimeQuality=notSynced Origin=@192.168.1.143] Module reset
Jun 25 12:22:50	192.168.1.143	1	user	notice		1970-1-1T0:0:0.0Z PME PXM - - SYS [TimeQuality=notSynced Origin=@0.0.0.0] Application Config Valid
Jun 25 12:22:50	192.168.1.143	1	user	notice		1970-1-1T0:0:0.0Z PME PXM - - SYS [TimeQuality=notSynced Origin=@0.0.0.0] Application code running
Jun 25 12:22:50	192.168.1.143	1	user	notice		1970-1-1T0:0:1.910Z PME PXM - - ETH [TimeQuality=notSynced Origin=@0.0.0.0] Ethernet link up
Jun 25 12:22:50	192.168.1.143	1	user	notice		1970-1-1T0:0:9.500Z PME PXM - - SYS [TimeQuality=notSynced Origin=@192.168.1.143] Config Retrieved from FDR Mismatch
Jun 25 12:22:50	192.168.1.143	1	user	notice		1970-1-1T0:0:9.500Z PME PXM - - SYS [TimeQuality=notSynced Origin=@192.168.1.143] Application Config Valid
Jun 25 12:23:01	192.168.1.143	1	user	notice		1970-1-1T0:0:21.10Z PME PXM - - FB [TimeQuality=notSynced Origin=@192.168.1.143] FB Operation Mode set to OPERATE

Figure 8.28 – PXM Events in a SysLog Server

The possible SysLog events and their descriptions are listed in section 12.3.

8.6. WEB SERVER

The PXM provides a web server allowing a user without the ProSoft Configurator for Modicon or Control Expert to view various diagnostics of the module.

The web server is view-only and thus no parameters or configuration can be altered from the web interface.

The web server pages are not updated automatically. To refresh the page, click on the required menu button again, or select the Refresh button if visible.

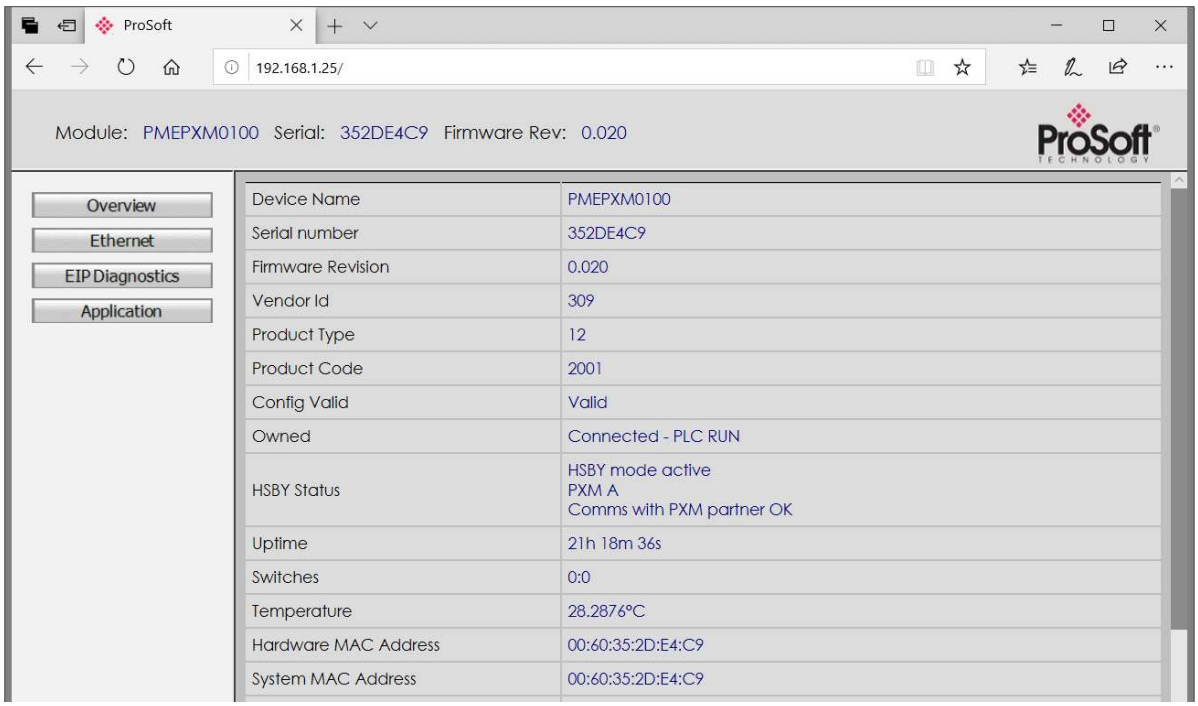


Figure 8.29 - Web interface

The parameters and diagnostics in the webserver will match those in the ProSoft Configurator for Modicon status monitoring of the PXM.

9. TROUBLESHOOTING GUIDE

Issue	*Resolution*
PCM - Downloading to PXM	
Unable to download "Failed to Connect"	Check physical Ethernet cabling between PC and M580 controller / NOC.
	Check PXM LED states
	Confirm the correct IP address has been configured in PCM.
	Confirm the PCM HSBY configuration matches that of the system. (Standalone vs HSBY)
	In the DTM Browser in Control Expert, check the correct IP address and identifier has been configured and confirm that DHCP is enabled and set to Identified by Name.
	Confirm that multiple M580 systems are not present on the same network. (Flat architecture is not supported.)
	If connection is using a NOC, confirm that the correct routing / default gateway has been configured.
	Check the PC has not been excluded by the PXM Security settings.
Unable to download "PXM not in correct Mode"	Take PXM out of RUN mode by setting the ***MasterControl.PROFIBUSRun*** bit to 0. OR Place PLC in STOP.
Importing Mapping into Control Expert	
Importing to Control Expert produces one or more **Errors**	Confirm Dynamic Arrays have been enabled in the Control Expert Project Settings.
	Confirm the PXM Instance name matches that configured in PCM.
	Confirm the Generic EDS DTM Connection Count and Connection Size match that configured in PCM.
PROFIBUS Devices not Online	
One or more devices are **Not Online** (Device icon is **Red**)	Confirm PLC in RUN.
	Confirm PXM is connected to PLC. (Check Freshness bit of the connection.)
	In Control Expert, check Master Control bits: ***MasterControl.PROFIBUSRun*** is set to 1, ***MasterControl.PROFIBUSStop*** is set to 0, ***MasterControl.PROFIBUSClear*** is set to 0.
	Check the physical PROFIBUS DP cabling and terminators.
	Confirm the Slave devices have the correct Station Addresses configured.

	Confirm the slave devices are able to operate at the selected BAUD rate.
	Check for duplicate PROFIBUS station addresses.
Devices not Exchanging Data	
One or more devices are Online but **Not Exchanging Data** (Device icon is **Orange**)	In PCM, open the Device Status for the specific the device and monitor the General tab.
	Disabled by PLC: Confirm ***DeviceEnable*** bits are set to 1 for the relevant Station Addresses.
	Require Toggle: Toggle the respective ***DeviceEnable*** bit to 0 and then back to 1. (This may be required after adding a new device online.)
	Ident Mismatch: Confirm the correct GSD file has been selected for that specific station address.
	StationID Mismatch: Ensure that after any PCM configuration change, that the mapping has been exported from PCM and imported into Control Expert.
	CRC Mismatch: Ensure that after any PCM configuration change, that the mapping has been exported from PCM and imported into Control Expert.
	In the specific Device Configuration in PCM, check the Slot configuration. Confirm slot, module type, and data point type and size. Confirm any Module Extended User Parameters have been configured correctly.
Different or incorrect behaviour after power cycle	
PXM does not operate correctly following a power-cycle or re-insertion into rack.	FDR server (in head module) may have a different configuration. Using the PCM: ***Download*** the configuration, and then select the **Upload to FDR** option.

10.TECHNICAL SPECIFICATIONS

10.1. DIMENSIONS

Below are the enclosure dimensions. All dimensions are in millimetres.

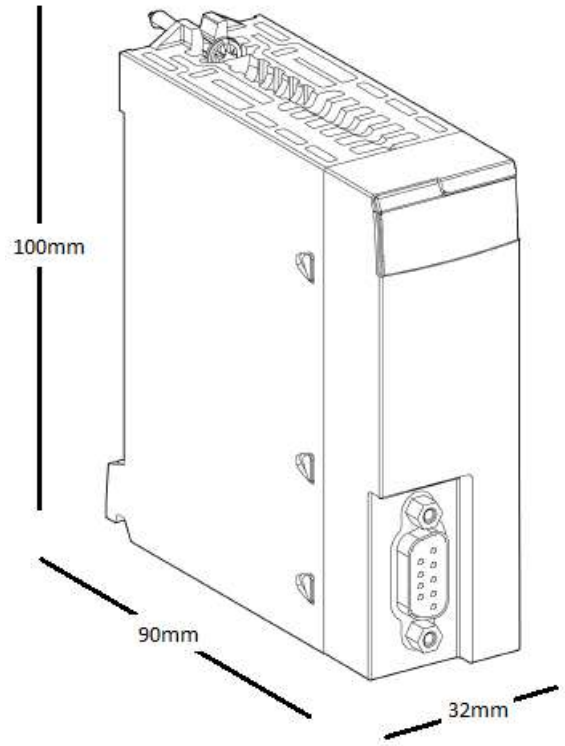


Figure 10.1 - PXM enclosure dimensions

10.2. ELECTRICAL

Specification	Rating
Power requirements	24Vdc derived from backplane
Power consumption	Typical: 43mA @ 24.25V => 1040mW Maximum: 70mA @ 24.25V => 1690mW
Electrical Isolation	1500Vrms for 1 minute (50/60 Hz) - PROFIBUS DP connector to backplane
Connector	M580 Backplane connector
Enclosure rating	IP20, NEMA/UL Open Type
Temperature	Standard Variant: 0 – 60 °C

	Harsh Variant: -25 – 75 °C
Earth connection	Yes, chassis based
Emissions	IEC61000-6-4
ESD Immunity	EN 61000-4-2
Radiated RF Immunity	IEC 61000-4-3
EFT/B Immunity	EFT: IEC 61000-4-4
Surge Immunity	Surge: IEC 61000-4-5
Conducted RF Immunity	IEC 61000-4-6

Table 10.1 - Electrical specification

10.3. PROFIBUS DP

Specification	**Rating**
Connector	Female DB9 connector
Conductor	See *PROFIBUS DP* Section.
Isolated	Yes
BAUD Rate supported	9.6 kbps 19.2 kbps 45.45 kbps 93.75 kbps 187.5 kbps 500 kbps 1.5 Mbps 3 Mbps 6 Mbps 12 Mbps

Table 10.2 – PROFIBUS DP specification

10.4. CERTIFICATIONS

For certification information refer to the product label.

11.PROFIBUS DP

11.1. INTRODUCTION

PROFIBUS is a vendor-independent, open fieldbus standard for a wide range of applications in manufacturing, process and building automation. Vendor independence and openness are guaranteed by the PROFIBUS standard EN 50 170. With PROFIBUS, devices of different manufacturers can communicate without special interface adjustments. PROFIBUS can be used for both high-speed time critical data transmission and extensive complex communication tasks. The PROFIBUS family consists of three compatible versions.

PROFIBUS DP
Optimized for high speed and inexpensive hookup, this PROFIBUS version is designed especially for communication between automation control systems and distributed I/O at the device level. PROFIBUS-DP can be used to replace parallel signal transmission with 24 V or 4-20 mA.

OSI Layer		PROFIBUS		
7	Application	DPV0	DPV1	DPV2
6	Presentation			
5	Session			
4	Transport			
3	Network			
2	Data Link	FDL		
1	Physical	EIA-485	Optical	MBP

Table 9.1 – PROFIBUS Protocol (OSI model)

To utilize these functions, various service levels of the DP protocol were defined:

- DP-V0 provides the basic functionality of DP, including
  - cyclic data exchange,
  - station, module and channel-specific diagnostics

- DP-V1 contains enhancements geared towards process automation, in particular
  - acyclic data communication for parameter assignment
  - alarm handling

- DP-V2 for isochronous mode and data exchange broadcast (slave-to-slave communication)

PROFIBUS PA
PROFIBUS PA is designed especially for process automation. It permits sensors and actuators to be connected on one common bus line through a dedicated DP/PA gateway or link between the PROFIBUS DP and PROFIBUS PA networks, even in intrinsically-safe areas. PROFIBUS PA permits data communication and power over the bus using a 2-wire technology according to the international standard IEC 1158-2.

PROFIBUS FMS
PROFIBUS FMS is the general-purpose solution for communication tasks at the cell level. Powerful FMS services open up a wide range of applications and provide great flexibility. PROFIBUS FMS can also be used for extensive and complex communication tasks. This protocol is the first developed for PROFIBUS, but it is no longer currently used.

PROFIBUS specifies the technical and functional characteristics of a serial fieldbus system with which decentralized digital controllers can be networked together from the field level to the cell level.

11.2. PROFIBUS MASTER AND SLAVE

PROFIBUS distinguishes between master devices and slave devices.

Master devices determine the data communication on the bus. A master can send messages without an external request when it holds the bus access rights (the token). Masters are also called '**active stations**' in the PROFIBUS protocol.

Slave devices are peripheral devices. Typical slave devices include input/output devices, valves, drives and measuring transmitters. They do not have bus access rights and they can only acknowledge received messages or send messages to the master when requested to do so. Slaves are also called '**passive stations**'

11.3. PROFIBUS MASTER CLASS 1 (DPM1) OR CLASS 2 (DPM2)

PROFIBUS DP Master class 1 (DPM1)
A class 1 master handles the normal communication or exchange of data with the slaves assigned to it. This is typically a PLC.

It uses **cyclic communication** to exchange process data with its associated slaves. The class 1 master sets the baud rate and the slave's auto-detect this rate. Each slave device is assigned to one master and only that master may write output data to that slave. Other masters may read information from any slave but can only write output data to their own assigned slaves.

PROFIBUS DP Master class 2 (DPM2)
A class 2 master is a special device primarily used for commissioning slaves and for diagnostic purposes. This is typically a Supervisor. It uses **acyclic communication** over what is known as the **MS2 channel**. A DPM2 does not have to be permanently connected to the bus system.

11.4. CYCLIC COMMUNICATION

The DP master class 1 cyclically exchanges data with all of the slaves assigned to it. This service is configured. During the configuration process, master and slave addresses are assigned, the bus parameters are defined, the types and numbers of modules (in the case of modular slaves) are specified, user-selectable parameter choices are made, etc.

Before data exchange can take place, the master will send parameterization and configuration telegrams to all of its assigned slaves. These parameters and configuration data are checked by the slaves. If both are valid, the master will initiate cyclic I/O data communication with the slave devices.

11.5. ACYCLIC COMMUNICATION

In addition to the cyclic data exchange, the PROFIBUS protocol has the option of acyclic communication. This service is not configured. There are 2 different communication channels possible between the requested master and the slave:

- **MS1 channel** (MS1 connection): can only be established if cyclic data exchange is taking place between that master (DPM1) and the slave
- **MS2 channel** (MS2 connection): is possible with several masters simultaneously, but the connection must be established explicitly by the master.

Acyclic reading and writing of data requires an established MS1 or MS2 connection.
For the MS1 channel, 3 conditions must be satisfied:

- The slave device must support the MS1 channel (key *C1_Read_Write_supp* at 1 in the GSD file)
- The DPV1_enable bit must be set during the parameter assignment
- Data exchange is taking place

For the MS2 channel, the connection must be explicitly initiated by the master. The maximum number of possible MS2 connections to the slave must not be reached. The connection can be closed by either the master or the slave device.

11.6. TOPOLOGY OF PROFIBUS DP

PROFIBUS devices are connected in a bus structure. Up to 32 stations (master or slaves) can be connected in one segment. The bus is terminated by an active bus terminator at the beginning and end of each segment. Both bus terminations must always be powered. When more than 32 stations are used, repeaters (line amplifiers) must be used to connect the individual bus segments.

11.7. PROFIBUS DP CABLE DESCRIPTION

Only one type of cable can be used for PROFIBUS network:

Parameter	Type A
Surge Impedance	135…165Ω (3 to 20 MHz)
Capacity	<30 pF/m
Loop Resistance	<110 Ω/km
Wire gauge	>0.64 mm
Conductor area	>0.34 mm²

Table 9.2 – PROFIBUS DP network cable

The maximum cable length depends on the transmission speed and cable type. The specified cable length can be increased using the repeaters. The use of more than 3 repeaters in series is not recommended.

Baudrate (kbps)	9.6	19.2	93.75	187.5	500	1500	3000-12000
Length A (m)	1200	1200	1200	1000	400	200	100

Table 9.3 – PROFIBUS DP cable length

11.8. PROFIBUS DP CONNECTOR DESCRIPTION

DB9 Pin Description	DB9 Pin#	DB9 Termination with PXM
Chassis ground	1	
Reserved	2	
Data+ / B	3	In case of termination connect this pin to Pin 8 (Data - / A) with a 220 ohm resistor
Tx enable	4	
Isolated ground	5	Connect this pin to Pin 8 (Data - / A) with a 390 ohm resistor
Voltage plus	6	Connect this pin to Pin 3 (Data + / B) with a 390 ohm resistor
Reserved	7	
Data- / A	8	
Reserved	9	

Table 9.4 – PROFIBUS DP connector

12.APPENDIX

12.1. DPV1 RESPONSE STATUS

DP Status	Description
00h	Successful
01h	Unknown command
02h	Invalid command data length
05h	FDL error (see extended error code)
06h	DPV1 Error (see extended error code)
07h	Another command is already in progress for this slave / class 2 connection.
0Ah	Invalid remote address
0Ch	Invalid data length
0Fh	DPV1 is not initialized
11h	Online state expected
13h	Invalid slave response
16h	No space left on command queue
17h	Timeout passed
18h	Previous command in progress

Table 12.1 – DP Status Response codes

12.2. DPV1 EXTENDED STATUS CODES

Extended DP Status	Description
00h	OK
01h	User error, SAP locked
02h	No resource for sending data, tried to send to SAP that was not configured
03h	No service available (SAP does not exist)

04h	Access point blocked
A080h	Application - Read Error
A180h	Application - Write Error
A280h	Application - Module Failure
A880h	Application - Version Conflict
A980h	Application - Feature Not Supported
B080h	Access - Invalid Index
B180h	Access - Write Length Error
B280h	Access - Invalid Slot
B380h	Access - Type Conflict
B480h	Access - Invalid Area
B580h	Access - State Conflict
B680h	Access - Access Denied
B780h	Access - Invalid Range
B880h	Access - Invalid Parameter
B980h	Access - Invalid Type
BA80h	Access - User Specific Error - 10
BB80h	Access - User Specific Error - 11
BC80h	Access - User Specific Error - 12
BD80h	Access - User Specific Error - 13
BE80h	Access - User Specific Error - 14
BF80h	Access - User Specific Error - 15
C080h	Resource - Read Constrain Conflict
C180h	Resource - Write Constrain Conflict
C280h	Resource - Resource Busy
C380h	Resource - Resource Unavailable
C880h	Resource - User Specific Error - 8
C980h	Resource - User Specific Error - 9
CA80h	Resource - User Specific Error - 10
CB80h	Resource - User Specific Error - 11
CC80h	Resource - User Specific Error - 12
CD80h	Resource - User Specific Error - 13

CE80h	Resource - User Specific Error - 14
CF80h	Resource - User Specific Error - 15

Table 12.2 – DP Extended Status Response codes

12.3. SYSLOG EVENTS

Event	Description
Class 1 Fwd Open Request - Assembly Group x	Received a forward open from the M580 controller on connection group x.
Class 1 Fwd Open Large Request - Assembly Group x	Received a large forward open from the M580 controller on connection group x.
Class 1 Fwd Close Request - Assembly Group x	Received a forward close from the M580 controller on connection group x.
Class 1 Connection Timeout - Assembly Group x	Connection group x has timed out its Cyclic Class 1 connection.
Snap: Name Mismatch: “x” vs “y”	When the SNAP message was received it was determined that the instance name in the configuration and the name generated from the SNAP message do not match (where x is the name generated in the SNAP while y is the instance name in the configuration).
Snap: Name match: “x”	When the SNAP message was received it was determined that the instance name in the configuration and the name generated from the SNAP message match.
First CIP packet for Conn x received	The first packet for the specific class 1 connection group x has been received.
Firmware update started	New firmware is being loaded on to the module.
Flash update CRC mismatch	The checksum for the firmware sent to the module has failed.
Signature failed	The signature for the firmware sent to the module has failed.
Update Hash Failed	The hash verification for the firmware sent to the module has failed.
HSBY Master Comms Error	There is a CIP error when receiving the inter-PXM message from the partner PXM.
Lost Comms to NTP Server at IP: x.x.x.x	The module has lost communication with the NTP server at IP address x.x.x.x
Reclaim Active Master A/B	Once a M580 swap has occurred the local PXM detects that the M580 controller has swapped and clears the latches for a PXM swap.
Lost Comms to PXM at IP: x.x.x.x	The module has lost communication with the partner PXM at IP address x.x.x.x in HSBY operation.
Comms established to PXM at IP: x.x.x.x	The module has established communication with the partner PXM at IP address x.x.x.x in HSBY operation.
HSBY Holdover Time Expired	The HSBY Swap transition time has expired.

HSBY Bus Inactivity Time Expired	The standby PXM in HSBY detected inactivity exceeding the configured period on the DP Bus.
HSBY Switch Go	The local PXM (active or standby) performs the actual swap in DP Masters.
HSBY Send Switch Arm Retry Limit	The standby PXM arm command sends has reached the retry limit without any feedback from the active PXM, in which case it will forcefully take over the DP Bus.
HSBY Send Switch Arm	The standby PXM sent a Switch Arm command to the active PXM to finish the last Profibus cycle and inform the standby to take over
HSBY Send Switch Go	Since receiving the arm command from the standby , the active PXM has finished it last Profibus Cycle and has sent the Go command to the standby PXM to take over the DP Bus.
Config (with signature x) sent to FDR - s	The configuration has been sent to the FDR (where x is the configuration signature and s is the filename).
Config Retrieved (with signature x) and Updated from FDR	The configuration has been retrieved from the FDR (where x is the configuration signature)
Config Retrieved (with signature x) from FDR Mismatch	The configuration has been retrieved from the FDR does not match the configuration on the PXM non-volatile (where x is the configuration signature)
Config Retrieved (with signature x) from FDR Match	The configuration has been retrieved from the FDR matches the configuration on the PXM non-volatile (where x is the configuration signature)
Config Retrieved from FDR is Blank	The configuration retrieved from the FDR is blank.
Config Retrieved from FDR Failed x	Could not retrieve the configuration from the FDR (where x is the status code: 1 – Success, 2 – Timeout, 3 – Incorrect Opcode, 4 – Packet Missing)
Config sent to FDR Failed %d	Could not send the configuration to the FDR (where x is the status code: 1 – Success, 2 – Timeout, 3 – Incorrect Opcode, 4 – Packet Missing)
HSBY Recv Switch Arm	The active PXM has received the switch arm command from the standby PXM.
HSBY Recv Switch Go	The standby PXM has received the Go command from the active PXM to take over the DP Bus.
Security List Fail for SNMP at Port x	Access to the SNMP port has failed, because SNMP has not been enabled in the security settings.
Security List Fail for HTTP at Port x	Access to the HTTP port has failed, because HTTP has not been enabled in the security settings.
ACL Subnet Fail for x.x.x.x	ACL has block IP address x.x.x.x because it did not meet the ACL subnet criteria.
ACL Security Fail for x.x.x.x at Port y	ACL has block IP address x.x.x.x on port y because it did not meet the ACL subnet criteria.
Application Config Too Big	The downloaded configuration is too big for the allowed application size.
Application Config (with signature x) Valid	The PXM has successfully loaded configuration from the NV memory (with signature x).
Application Config Failed - x vs y	The PXM has failed to load configuration from the NV memory (with signature x vs y).

Cfg: Name Mismatch: x vs y	Configuration instance name does not match the name in the SNAP message (where x is the name generated in the SNAP while y is the instance name in the configuration).
Cfg: Name match: x	When the configuration was loaded it was determined that the instance name in the configuration and the name generated from the SNAP message match.
NAND Config Read Error	Failed to read the NAND memory.
No active Class 2 Clients	When trying to send the DPV1 Class 2 message, it was determined that there is no connection open for this specific node at the specific SAP.
Fallback to Master Not Ready To Enter Ring	In a multi DP Master system, the local PXM has not received a Token from any DP Master on the bus for the fallback amount of time in which case it resets the parameters to rejoin the token ring.
FB Alarm Ack Retry Limit Reached	The retry limit has been reached by the PXM trying to acknowledge the alarm in the DP device.
FB Token Not Expected	A DP token was received while waiting for the response from a field device.
Alarm Negative Ack Received	While trying to acknowledge the alarm in the DP device, a negative response was received.
Alarm Extract No Resource	While trying to acknowledge the alarm in the DP device the field device replied with no resources available.
Urgent Diagnostics Client Overflow	The amount of DP devices listed for reading urgent diagnostic messages has overflowed.
Class 2 Client overflow	A new DPV1 Class 2 request was received, but the limit on the amount of open DPV1 Class 2 connections has been reached.
Unknown Relay Response	An unknown DPV1 Class 1/2 response was received.
Failed to set init parameter index x for node y	The PXM has failed to set the parameter at index x for node y during device startup (using the init parameters configured).
HSBY set DP to Active	The local PXM has been set to the active DP Master in a HSBY system.
HSBY set DP to Standby	The local PXM has been set to the standby DP Master in a HSBY system.
FB Alarm Ack Timeout Retry Limit Reached	The retry limit has been reached by the PXM trying to acknowledge the alarm in the DP device due to the device not responding.
FB Operation Mode set to OFFLINE	The local PXM Fieldbus mode has been set to OFFLINE.
FB Operation Mode set to STOP	The local PXM Fieldbus mode has been set to STOP.
FB Operation Mode set to OPERATE	The local PXM Fieldbus mode has been set to OPERATE.
FB Operation Mode set to CLEAR	The local PXM Fieldbus mode has been set to CLEAR.
Application code running	The PXM has successfully booted with the application firmware.
Boot error	The PXM has not successfully booted with the application firmware.

Table 12.3 – SysLog Event Descriptions

12.4. VERIFICATION NOTIFICATIONS

Domain	Description	Severity
General	Invalid Device (A) IP address 'x'	Error
General	Invalid Device B IP address 'x'	Error
General	Invalid Instance Name 'x'	Error
General	Invalid NTP Update Interval. (1-32000)s	Error
Master	Invalid DP Node Address. (0-125)	Error
Master	Invalid Gap Update Factor. (1-100)	Error
Master	Invalid Token Retry Limit. (1-5)	Error
HSBY	Invalid HSBY Holdover Time: 'x'. (>=290)	Error
HSBY	Invalid HSBY DP Deadtime: 'x'. (>=10)	Error
HSBY	Invalid HSBY Switch Over Cmd Rate: 'x'. (2-20)	Error
Slave	Verifying [InstanceName] ([VendorName] [ModelName])	Info
Slave	Invalid Station Address. (0-125)	Error
Slave	Device Station Address clashes with Master Station Address.	Error
Slave	Illegal Device Name: 'x'	Error
Slave	Device WatchDog value must be greater than Profibus Cycle.	Error
Slave	Device does not support current BAUD rate 'x'	Warning
Slave	Duplicate Station Address: 'x'	Error
Slave	Duplicate Instance Name: 'x'	Error
Slave		Error
Slave	Illegal Slot Data Point Description: 'x'	Error
Slave	Duplicate Slot Data Point Description 'x'	Error
Slave	Byte Length not a multiple of Data Type: [Description] - Length: 'y' + '' DataType: 'x'	Error
Slave	Invalid Mapping - Connection Size too small.	Error

Table 12.4 – Verification Notifications

12.5. ADDITIONAL CIP OBJECTS

12.5.1. PXM GENERAL STATUS

REQUEST:

Parameter	Description
Service	Get Attribute Single
Class	0x301 (Hex)
Instance	1
Attribute	1

Table 12.5 – PXM General Status Request

REPLY:

Parameter	*Data Type*	*Description*
Status	DINT	Bit0 – Config Valid Bit1 - Owned Both Connections Ok Bit2 – HSBY Mode Configured Bit3 – HSBY Masters Config Mismatch Bit4 – HSBY Mode Active 0 = Standby, 1 = Active Bit5 - PXMAB 0 = A, 1 = B Bit6 – Comms With PXM Partner OK Bit7 – Profibus Running Bit8 - Reserved Bit9 – Partner Owned Bit10 – Partner Profibus Running Bit11 – Multiple ENIP Connection Bit12 - Duplicate DP Station Bit13 – PLC In Stop
Master Config CRC	INT	Downloaded configuration CRC
Master Node Address	BYTE	Node address of the DP Master (TS)
Master BAUD Rate	BYTE	0 - 9.6 1 - 19.2 2 - 31.25 3 - 45.45 4 - 93.75 5 - 187.5 6 - 500 7 - 1500 8 - 3000 9 - 6000 10 - 12000
Up Time	DINT	Seconds since start

Process Cycles	DINT	Number of CPU cycles in last second.
Device Temperature	REAL	Internal Temperature of CPU
Hardware MAC Address	BYTE[6]	Hardware MAC Address
System MAC Address	BYTE[6]	System MAC Address
DIP Switch Status	BYTE	DIP Switch Status
M580 Slot Number	BYTE	M580 Slot position
Firmware Major Rev	BYTE	Firmware Major Revision
Firmware Minor Rev	BYTE	Firmware Minor Revision
Operation State	BYTE	FXM operating mode state 0 = Initialization 1 =Unconfigured 2 = Configured 3 = Connected-Stop 4 = Connected-Run 5 = Connected-Clear
Redundant State	BYTE	FXM operating mode regarding the Redundancy 1 = Standalone , 2 = Primary Alone, 3 = Primary Assisted, 4 = Standby Alone, 5 = Standby Assisting
Display	BYTE[8]	1 byte per led : 0 = Off, 1=Green, 2=Red, 3=Yellow (Add 4 if flashing.) 5=Flash-Green, 6=Flash-Red, 7=Flash-Yellow Byte0 = Module Run, Byte1 = SF Byte2 = BUSF Byte3 = BS Byte4 = Profibus Run Byte5 = CLS2 Byte6 = HSBY Byte7 = (Reserved)
Ethernet Status	BYTE	Main Ethernet Status Bit 0 : Link up/down for Ethernet port 1 (0: link is down, 1: link is up), Bit 1 : reserved for Ethernet port 2, Bit 2 : reserved for Ethernet port 3, Bit 3 : Reserved, Bit 4 : EtherNet/IP RPI CCOTF in Progress (0: not in progress, 1: in progress), Bit 5 : Redundancy status / backup path available (0: not available, 1: available), Bit 6 : Redundant owner is present (0: not available, 1: available), Bit 7 : Global service status (0: one or more services not operating normally, 1: all operational)

Ethernet Services	BYTE	Detailed Ethernet Services Status. One bit for each user-observable feature (0: NOK, 1: OK/NA) Bit 0 : reserved RSTP Service 0, service is not operating normally 1, service is operating normally or disabled or not implemented Bit 1 : SNTP Service 0, service is not operating normally 1, service is operating normally or Disabled Bit 2 : reserved Port502 Service Bit 3 : SNMP 0, service is not operating normally 1, service is operating normally or Disabled Bit 4 : FDR regular client 0, unable to download PRM file (during CCOTF or HSBY switchover) 1, service is operating normally Bit 5 Firmware Upgrade 0, Firmware Upgrade unauthorized 1, service is operating normally Bit 6 : Web 0 : no web page available 1 : service is operating normally or Disabled Bit 7 :Syslog Server 0 : Event logging disrupted 1 : service is operating normally or Disabled
Syslog Status	BYTE	Bit 0 , set to 1 syslog client does not receive acknowledge from the syslog server
Syslog Buffer Free	INT	Free place in % of the events buffer
Syslog Lost Events	INT	Number of Events lost since the last restart
Fieldbus State	BYTE	PXM operating mode regarding the Fieldbus (0=Idle) ; 1= No-Conf ; 2= PROFIBUS Offline 3= PROFIBUS STOP 4 = NA 5 = PROFIBUS OPERATE 6 = PROFIBUS CLEAR
Fieldbus Health	BYTE	PXM diagnostic regarding the Field BUS (0= Idle) 1 = Field Bus Error, 2 = Device(s) Fault , 3 = Device(s) Error , 4 = Device(s) Fault & Error
Fieldbus Bandwidth	INT	Current Field Bus load in %
Slaves Prog List	Bool[128]	1 bit per slave device : 1 = Programmed (in config), 0 = Not expected (Not conf or disabled)

Slaves Live List	Bool[128]	1 bit per slave device : 1 = responding ; 0= not responding or disabled
Slaves Diag List	Bool[128]	1 bit per slave device : 1 = Error or Fault on expected slave 0 = no error reported
Slaves Data Exchange Active	Bool[128]	1 bit per slave device : 1 = Data Exchange State is active 0 = Data Exchange State is **not** active
IO Bytes Per Second	DINT	Count of Cyclic Data exchange per second
Acyclic Request Client Count	BYTE	Count of Acyclic requests pending
Duplicate Station Present	BYTE	Bit 0 - If set then there is another device on the network with the same node address
Config Retrieval Status	BYTE	0 - Not Config Receive Attempt 1 - Successful Config Retrieval from head 2 - Failed Config Retrieval: Timeout 3 - Failed Config Retrieval: Incorrect Op-Code 4 - Failed Config Retrieval: Missing Block
Config Sending Status	BYTE	0 - Not Config Sending Attempt 1 - Successful Config Sent to head 2 - Failed Config Sending: Timeout 3 - Failed Config Sending: Incorrect Op-Code 4 - Failed Config Sending: Missing Block
PXM Role Name	BYTE[16]	Name build for backplane

Table 12.6 – PXM General Status Reply

12.5.2. PXM MASTER STATISTICS

REQUEST:

Parameter	Description
Service	Get Attribute Single Get Attribute All, Reset
Class	0x305 (Hex)
Instance	1
Attribute	(As below)

Table 12.7 – PXM Master Statistics Request

REPLY:

Attribute	Parameter	Data Type	Description
1	TxPacketCount	DINT	Tx Packet Count
2	RxPacketCount	DINT	Rx Packet Count
3	ChecksumFailedPacketCount	DINT	Checksum Failed Packet Count
4	NoReplyCount	DINT	No Reply Count
5	DPV1Class1ReadTxCount	DINT	DPV1 Class 1 Read Tx Count
6	DPV1Class1ReadRxCount	DINT	DPV1 Class 1 Read Rx Count
7	DPV1Class1ReadErrCount	DINT	DPV1 Class 1 Read Err Count
8	DPV1Class1WriteTxCount	DINT	DPV1 Class 1 Write Tx Count
9	DPV1Class1WriteRxCount	DINT	DPV1 Class 1 Write Rx Count
10	DPV1Class1WriteErrCount	DINT	DPV1 Class 1 Write Err Count
11	DPV1Class2InitTxCount	DINT	DPV1 Class 2 Init Tx Count
12	DPV1Class2InitRxCount	DINT	DPV1 Class 2 Init Rx Count
13	DPV1Class2InitErrCount	DINT	DPV1 Class 2 Init Err Count
14	DPV1Class2AbortTxCount	DINT	DPV1 Class 2 Abort Tx Count
15	DPV1Class2AbortRxCount	DINT	DPV1 Class 2 Abort Rx Count
16	DPV1Class2ReadTxCount	DINT	DPV1 Class 2 Read Tx Count
17	DPV1Class2ReadRxCount	DINT	DPV1 Class 2 Read Rx Count
18	DPV1Class2ReadErrCount	DINT	DPV1 Class 2 Read Err Count
19	DPV1Class2WriteTxCount	DINT	DPV1 Class 2 Write Tx Count
20	DPV1Class2WriteRxCount	DINT	DPV1 Class 2 Write Rx Count
21	DPV1Class2WriteErrCount	DINT	DPV1 Class 2 Write Err Count
22	SetSlaveAddrTxCount	DINT	Set Slave Address Tx Count
23	SetSlaveAddrRxCount	DINT	Set Slave Address Rx Count
24	SetSlaveAddrErrCount	DINT	Set Slave Address Err Count
25	GlobalCtrlTxCount	DINT	Global Ctrl Tx Count
26	GlobalCtrlRxCount	DINT	Global Ctrl Rx Count
27	LastProfibuscycletime	DINT	Last Profibus cycle time
28	MaxProfibuscycletime	DINT	Max Profibus cycle time
29	MinProfibuscycletime	DINT	Min Profibus cycle time
30	LastTokenHoldtime	DINT	Last Token Hold time
31	MaxTokenHoldtime	DINT	Max Token Hold time
32	MinTokenHoldtime	DINT	Min Token Hold time
33	LastResponseTime	DINT	Last Response Time
34	MaxResponseTime	DINT	Max Response Time
35	MinResponseTime	DINT	Min Response Time
36	FBFaultCount	DINT	FB Fault Count
37	DeviceFaultCount	DINT	Device Fault Count
38	AcyclicRequestClientCountOverrun	DINT	Acyclic Request Client count overrun
39	TokenPassRetryCount	DINT	Token Pass Retry Count
40	TokenPassFailCount	DINT	Token Pass Fail Count
41	UnexpectedPacketReceived	DINT	Unexpected Packet Received

42	FBInactivityCount	DINT	FB Inactivity Count
43	DuplicateStationDetectCount	DINT	Duplicate Station Detect Count
44	InvalidResponseLengthCount	DINT	Invalid Response Length Count
45	FDLFaultCount	DINT	FDL Fault Count
46	ExtractAlarmSuccessCount	DINT	Extract Alarm Success Count
47	ExtractAlarmFailCount	DINT	Extract Alarm Fail Count
48	InitParameterSetSuccessCount	DINT	Initialize parameter set succeeded
49	InitParameterSetFailCount	DINT	Initialize parameter set failed
50	DeviceReconfigureCount	DINT	Number of times devices have been reconfigured
51	DeviceReparameterizeCount	DINT	Number of times devices have been reparameterize
52	ExtDiagOverflowCount	DINT	Number of times the extended diagnostics have overflowed

Table 12.8 – PXM Master Statistics Reply

12.5.3. PROFIBUS END-POINT DIAGNOSTIC ADMIN

REQUEST:

Parameter	Description
Service	Get Attribute Single Get Attribute All
Class	0x306 (Hex)
Instance	0
Attribute	(As below)

Table 12.9 – Profibus End-Point Diagnostic Admin Request

REPLY:

Attribute	Parameter	Data Type	Description
1	VersionHigh	DINT	Version High
2	VersionLow	DINT	Version Low
3	DeviceCount	DINT	Device Count
4	AddressDevice1	BYTE	Address Device 1
…	…	…	…
n	AddressDevicen	BYTE	Address Device n

Table 12.10 – Profibus End-Point Diagnostic Admin Reply

12.5.4. PROFIBUS END-POINT DIAGNOSTIC

REQUEST:

Parameter	Description
Service	Get Attribute Single Get Attribute All
Class	0x306 (Hex)
Instance	Device Index (From Admin)
Attribute	(As below)

Table 12.11 – Profibus End-Point Diagnostic Request

REPLY:

Attribute	Parameter	Data Type	Description
1	Station	BYTE	station address of slave
2	Status	INT	Bit 0 - Online Bit 1 - Data Exchange Active Bit 2 - Ident Mismatch Bit 3 – Disabled By Output Assembly Bit 4 - Error Bit 5 - Alarm Pending Bit 6 - Diagnostics Pending Bit 7 - Output Assembly Node Address Mismatch Bit 8 - Mapping CRC Mismatch
3	RxDataLen	BYTE	Data received from slave length in bytes
4	TxDataLen	BYTE	Data to be sent to slave length in bytes
5	DiagMaxLen	BYTE	Maximum length of diag status response
6	DiagLen	BYTE	Diagnostic from slave length in bytes
7	DiagSts1	BYTE	Status byte 1 from slave
8	DiagSts2	BYTE	Status byte 2 from slave
9	DiagSts3	BYTE	Status byte 3 from slave
10	DiagStn	BYTE	Station that configured Slave
11	DiagID_hi	BYTE	ID hi byte sent back from slave
12	DiagID_lo	BYTE	ID lo byte sent back from slave
13	DiagData	ANY_ARRAY_BYTE	Vendor defined diagnostic info from slave

Table 12.12 – Profibus End-Point Diagnostic Reply

12.5.5. PROFIBUS END-POINT STATISTICS ADMIN

REQUEST:

Parameter	Description
Service	Get Attribute Single Get Attribute All
Class	0x307 (Hex)
Instance	0
Attribute	(As below)

Table 12.13 – Profibus End-Point Statistics Admin Request

REPLY:

Attribute	Parameter	Data Type	Description
1	VersionHigh	DINT	Version High
2	VersionLow	DINT	Version Low
3	DeviceCount	DINT	Device Count
4	AddressDevice1	BYTE	Address Device 1
…	…	…	…
n	AddressDevicen	BYTE	Address Device n

Table 12.14 – Profibus End-Point Statistics Admin Reply

12.5.6. PROFIBUS END-POINT STATISTICS

REQUEST:

Parameter	Description
Service	Get Attribute Single Get Attribute All, Reset
Class	0x307 (Hex)
Instance	Device Index (From Admin)
Attribute	(As below)

Table 12.15 – Profibus End-Point Statistics Request

REPLY:

Attribute	Parameter	Data Type	Description

1	TxPacketCount	DINT	Tx Packet Count
2	RxPacketCount	DINT	Rx Packet Count
3	ChecksumfailedpacketCount	DINT	Checksum failed packet Count
4	NoReplyCount	DINT	No Reply Count
5	DPV1Class1ReadTxCount	DINT	DPV1 Class 1 Read Tx Count
6	DPV1Class1ReadRxCount	DINT	DPV1 Class 1 Read Rx Count
7	DPV1Class1ReadErrCount	DINT	DPV1 Class 1 Read Err Count
8	DPV1Class1WriteTxCount	DINT	DPV1 Class 1 Write Tx Count
9	DPV1Class1WriteRxCount	DINT	DPV1 Class 1 Write Rx Count
10	DPV1Class1WriteErrCount	DINT	DPV1 Class 1 Write Err Count
11	DPV1Class2InitTxCount	DINT	DPV1 Class 2 Init Tx Count
12	DPV1Class2InitRxCount	DINT	DPV1 Class 2 Init Rx Count
13	DPV1Class2InitErrCount	DINT	DPV1 Class 2 Init Err Count
14	DPV1Class2AbortTxCount	DINT	DPV1 Class 2 Abort Tx Count
15	DPV1Class2AbortRxCount	DINT	DPV1 Class 2 Abort Rx Count
16	DPV1Class2ReadTxCount	DINT	DPV1 Class 2 Read Tx Count
17	DPV1Class2ReadRxCount	DINT	DPV1 Class 2 Read Rx Count
18	DPV1Class2ReadErrCount	DINT	DPV1 Class 2 Read Err Count
19	DPV1Class2WriteTxCount	DINT	DPV1 Class 2 Write Tx Count
20	DPV1Class2WriteRxCount	DINT	DPV1 Class 2 Write Rx Count
21	DPV1Class2WriteErrCount	DINT	DPV1 Class 2 Write Err Count
22	SetSlaveAddrTxCount	DINT	Set Slave Address Tx Count
23	SetSlaveAddrRxCount	DINT	Set Slave Address Rx Count
24	SetSlaveAddrErrCount	DINT	Set Slave Address Err Count
25	GlobalCtrlTxCount	DINT	Global Ctrl Tx Count
26	GlobalCtrlRxCount	DINT	Global Ctrl Rx Count
27	UnexpectedPacketReceived	DINT	Unexpected Packet Received
28	InvalidResponseLengthCount	DINT	Invalid Response Length Count
29	FDLFaultCount	DINT	FDL Fault Count
30	ExtractAlarmSuccessCount	DINT	Extract Alarm Success Count
31	ExtractAlarmFailCount	DINT	Extract Alarm Fail Count
32	InitParameterSetSuccessCount	DINT	Initialize parameter set succeeded
33	InitParameterSetFailCount	DINT	Initialize parameter set failed
34	DeviceReconfigureCount	DINT	Number of times devices have been reconfigured
35	DeviceReparameterizeCount	DINT	Number of times devices have been reparameterize
36	ExtDiagOverflowCount	DINT	Number of times the extended diagnostics have overflowed

Table 12.16 – Profibus End-Point Statistics Reply

12.5.7. PROFIBUS DATA EXCHANGE ADMIN

REQUEST:

Parameter	Description
Service	Get Attribute Single Get Attribute All
Class	0x308 (Hex)
Instance	0
Attribute	(As below)

Table 12.17 – Profibus Data Exchange Admin Request

REPLY:

Attribute	Parameter	Data Type	Description
1	VersionHigh	DINT	Version High
2	VersionLow	DINT	Version Low
3	DeviceCount	DINT	Device Count
4	AddressDevice1	BYTE	Address Device 1
…	…	…	…
n	AddressDevicen	BYTE	Address Device n

Table 12.18 – Profibus Data Exchange Admin Reply

12.5.8. PROFIBUS DATA EXCHANGE

REQUEST:

Parameter	Description
Service	Get Attribute Single Get Attribute All
Class	0x308 (Hex)
Instance	Device Index (From Admin)
Attribute	(As below)

Table 12.19 – Profibus Data Exchange Request

REPLY:

Attribute	Parameter	Data Type	Description
1	ProfibusStation	BYTE	Profibus Station Address
2	SizeIn	BYTE	Size Input

3	SizeOut	BYTE	Size Output
4	DataIn	BYTE[244]	Data Input
5	DataOut	BYTE[244]	Data Output

Table 12.20 – Profibus Data Exchange Reply

12.5.9. EtherNet/IP Connection Diagnostics

REQUEST:

Parameter	Description
Services	Get Attribute Single, Get Attribute All, Reset, Get and Clear
Class	0x350 (Hex)
Instance	1
Attribute	2

Table 12.21 – EtherNet/IP Connection Diagnostics Request

REPLY:

Parameter	Data Type	Description
MaxCIPIOConnectionsOpened	UINT	Maximum number of CIP IO Connections opened
CurrentCIPIOConnections	UINT	Number of CIP IO Connections currently opened
MaxCIPExplicitConnectionsOpened	UINT	Maximum number of CIP Explicit Connections opened
CurrentCIPExplicitConnections	UINT	Number of CIP Explicit Connections currently opened
CIPConnectionsExplicitOpeningErrors	UINT	Incremented at each attempt to open a CIP connection that fails
CIPConnectionsTimoutErrors	UINT	Incremented when a CIP connection is timed out
MaxEIPTCPConnectionsOpened	UINT	Maximum Number of TCP connection opened and used for EIP communication
CurrentEIPTCPConnectionsOpened	UINT	Number of TCP connections currently opened and used for EIP communication

Table 12.22 EtherNet/IP Connection Diagnostics Reply

12.5.10. ETHERNET/IP IO MESSAGE DIAGNOSTICS

REQUEST:

Parameter	Description
Service	Get Attribute Single, Get Attribute All, Reset, Get and Clear
Class	0x350 (Hex)
Instance	1
Attribute	3

Table 12.23 – EtherNet/IP IO Message Diagnostics Request

REPLY:

Parameter	Data Type	Description
IOProdCount	UINT	Incremented each time a Class 0/1 CIP Message is sent
IOconsumptionCount	UINT	Incremented each time a Class 0/1 CIP Message is received
IOprodSendErrorsCount	UINT	Incremented each Time a Class 0/1 Message is not sent
IOconsumptionReceiveErrorsCount	UINT	Incremented each time a consumption is received with an error

Table 12.24 – EtherNet/IP IO Message Diagnostics Reply

12.5.11. ETHERNET/IP EXPLICIT MESSAGE DIAGNOSTICS

REQUEST:

Parameter	Description
Service	Get Attribute Single, Get Attribute All, Reset, Get and Clear
Class	0x350 (Hex)
Instance	1
Attribute	4

Table 12.25 – EtherNet/IP Explicit Message Diagnostics Request

REPLY:

Parameter	Data Type	Description
Class3MsgSendCount	UINT	Incremented each time a Class 3 CIP Message is sent
Class3MsgRecCount	UINT	Incremented each time a Class 3 CIP Message is received
UCMMMsgSendCount	UINT	Incremented each time an UCMM Message is sent
UCMMMsgReceiveCount	UINT	Incremented each time an UCMM Message is received.

Table 12.26 – EtherNet/IP Explicit Message Diagnostics Reply

12.5.12. ETHERNET/IP COMMUNICATION CAPACITY

REQUEST:

Parameter	Description
Service	Get Attribute Single, Get Attribute All, Reset, Get and Clear
Class	0x350 (Hex)
Instance	1
Attribute	5

Table 12.27 – EtherNet/IP Communication Capacity Request

REPLY:

Parameter	Data Type	Description
CapacityMaxCIPConnections	UINT	Max supported CIP Connections
CapacityMaxTCPConnections	UINT	Max supported TCP connections
CapacityMaxUrgentPriorityRate	UINT	Max CIP transport class 0/1 Urgent priority messages Packets/s
CapacityMaxScheduledPriorityRate	UINT	Max CIP transport class 0/1 Scheduled priority messages Packets/s
CapacityMaxHighPriorityRate	UINT	Max CIP transport class 0/1 High priority messages Packets/s
CapacityMaxLowPriorityRate	UINT	Max CIP transport class 0/1 Low priority messages Packets/s
CapacityMaxExplictRate	UINT	Max CIP transport class 2/3 or other EIP messages Packets/s

Table 12.28 – EtherNet/IP Communication Capacity Reply

12.5.13. EtherNet/IP Bandwidth Diagnostics

REQUEST:

Parameter	Description
Service	Get Attribute Single, Get Attribute All, Reset, Get and Clear
Class	0x350 (Hex)
Instance	1
Attribute	6

Table 12.29 – EtherNet/IP Bandwidth Diagnostics Request

REPLY:

Parameter	Data Type	Description
CurrentSendingUrgentPriorityRate	UINT	CIP transport class 0/1 Urgent priority messages Packets/s sent
CurrentReceiveUrgentPriorityRate	UINT	CIP transport class 0/1 Urgent priority messages Packets/s received
CurrentSendingScheduledPriorityRate	UINT	CIP transport class 0/1 Scheduled priority messages Packets/s sent
CurrentReceiveScheduledPriorityRate	UINT	CIP transport class 0/1 Scheduled priority messages Packets/s received
CurrentSendingHighPriorityRate	UINT	CIP transport class 0/1 High priority messages Packets/s sent
CurrentReceiveHighPriorityRate	UINT	CIP transport class 0/1 High priority messages Packets/s received
CurrentSendingLowPriorityRate	UINT	CIP transport class 0/1 Low priority messages Packets/s sent
CurrentReceiveLowPriorityRate	UINT	CIP transport class 0/1 Low priority messages Packets/s received
CurrentSendingExplicitRate	UINT	CIP transport class 2/3 or other EIP messages packets sent
CurrentReceiveionExplicitRate	UINT	CIP transport class 2/3 or other EIP messages packets received

Table 12.30 – EtherNet/IP Bandwidth Diagnostics Reply

12.5.14. IO Connection Diagnostic

REQUEST:

Parameter	Description
Service	Get Attribute Single
Class	0x352 (Hex)
Instance	0
Attribute	(As below)

Table 12.31 – IO Connection Diagnostic Request

REPLY:

Attribute	Parameter	Data Type	Description
1	Revision	UINT	Revision
2	MaxInstances	UINT	Number of connections

Table 12.32 – IO Connection Diagnostic Reply

12.5.15. IO CONNECTION INFORMATION

REQUEST:

Parameter	Description
Service	Get Attribute Single, Get Attribute All, Reset
Class	0x352 (Hex)
Instance	(Connection)
Attribute	1

Table 12.33 – IO Connection Information Request

REPLY:

Parameter	Data Type	Description
IO Product Counter	UINT	Incremented at each production
IO Consumption Counter	UINT	Incremented at each consumption
IO Product Send Error	UINT	Incremented each time a production is not sent
IO Consumption Receive Error	UINT	Incremented each time a consumption is received with an error
CIP Connection TimeOut errors	UINT	Incremented when a connection is timed out
CIP Connection Opening errors	UINT	Incremented at each attempt to open a connection that fails

CIP Connection State	UINT	State of the CIP IO connection
CIP Last Error General Status	UINT	“General Status” of the last error detected on the connection
CIP Last Error Extended Status	UINT	“Extended Status” of the last error detected on the connection
Input Com Status	UINT	Communication Status of the Inputs
Output Comm status	UINT	Communication Status of the Outputs

Table 12.34 – IO Connection Information Reply

12.5.16. IO CONNECTION DIAGNOSTICS

REQUEST:

Parameter	Description
Service	Get Attribute Single, Get Attribute All, Reset
Class	0x352 (Hex)
Instance	(Connection)
Attribute	2

Table 12.35 – IO Connection Diagnostics Request

REPLY:

Parameter	Data Type	Description
Production Connection ID	UINT	Connection ID for Production
Consumption Connection ID	UINT	Connection ID for Consumption
Production RPI	UINT	RPI for production
Production API	UINT	API for production
Consumption RPI	UINT	RPI for consumption
Production API	UINT	API for consumption
Production Connection parameters	UINT	Connection parameters for production
Consumption Connection parameters	UINT	Connection parameters for consumption
Local IP	UINT	Local IP Address
Local UDP port	UINT	Local UDP port
Remote IP	UINT	Remote IP
Remote UDP port	UINT	Remote UDP port
Production Multicast IP	UINT	Multicast IP used for production
Consumption Multicast IP	UINT	Multicast IP used for consumption
Protocol supported	UINT	Protocol(s) supported on the connection

Table 12.36 – IO Connection Diagnostics Reply

12.5.17. EXPLICIT CONNECTION DIAGNOSTIC

REQUEST:

Parameter	Description
Service	Get Attribute Single, Get Attribute All, Reset
Class	0x353 (Hex)
Instance	1
Attribute	(As below)

Table 12.37 – Explicit Connection Diagnostic Request

REPLY:

Attribute	Parameter	Data Type	Description
1	OriginatorConnectionID	UINT	O to T Connection ID
2	OriginatorIP	UINT	Originator IP Address
3	OriginatorTCPport	UINT	Originator TCP Port
4	TargetConnectionID	UINT	T to O Connection ID
5	TargetIP	UINT	Target IP Address
6	TargetTCPport	UINT	Target TCP Port
7	MsgSendCounter	UINT	Incremented each time a Class 3 CIP Message is sent on the connection
8	MsgReceiveCounter	UINT	Incremented each time a Class 3 CIP Message is received on the connection

Table 12.38 – Explicit Connection Diagnostic Reply

13.INDEX

R

S

T

U

W