# Modicon TM4

## Expansion Modules

## Programming Guide

**Original instructions**

**EIO0000003149.04**
**04/2026**


www.se.com

# Legal Information

The information provided in this document contains general descriptions, technical characteristics and/or recommendations related to products/solutions.

This document is not intended as a substitute for a detailed study or operational and site-specific development or schematic plan. It is not to be used for determining suitability or reliability of the products/solutions for specific user applications. It is the duty of any such user to perform or have any professional expert of its choice (integrator, specifier or the like) perform the appropriate and comprehensive risk analysis, evaluation and testing of the products/solutions with respect to the relevant specific application or use thereof.

The Schneider Electric brand and any trademarks of Schneider Electric SE and its subsidiaries referred to in this document are the property of Schneider Electric SE or its subsidiaries. All other brands may be trademarks of their respective owner.



Schneider Electric does not grant any right or license for commercial use of the document or its content, except for a non-exclusive and personal license to consult it on an "as is" basis.

Schneider Electric reserves the right to make changes or updates with respect to or in the content of this document or the format thereof, at any time without notice.

**To the extent permitted by applicable law, no responsibility or liability is assumed by Schneider Electric and its subsidiaries for any errors or omissions in the informational content of this document, as well as any non-intended use or misuse of the content thereof.**

# Table of Contents

# Safety Information

## Important Information

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 the 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.

# About the Document

## Document Scope

This document describes the configuration of the TM4 expansion modules for EcoStruxure Machine Expert. For further information, refer to the separate documents provided in the online help.

## Validity Note

This document has been updated for the release of EcoStruxure™ Machine Expert V2.6.

## Product Related Information

**⚠WARNING**

**LOSS OF CONTROL**

- Perform a Failure Mode and Effects Analysis (FMEA), or equivalent risk analysis, of your application, and apply preventive and detective controls before implementation.
- Provide a fallback state for undesired control events or sequences.
- Provide separate or redundant control paths wherever required.
- Supply appropriate parameters, particularly for limits.
- Review the implications of transmission delays and take actions to mitigate them.
- Review the implications of communication link interruptions and take actions to mitigate them.
- Provide independent paths for control functions (for example, emergency stop, over-limit conditions, and error conditions) according to your risk assessment, and applicable codes and regulations.
- Apply local accident prevention and safety regulations and guidelines.[1]
- Test each implementation of a system for proper operation before placing it into service.

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

[1] For additional information, refer to NEMA ICS 1.1 (latest edition), *Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control* and to NEMA ICS 7.1 (latest edition), *Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems* or their equivalent governing your particular location.

**⚠WARNING**

**UNINTENDED EQUIPMENT OPERATION**

- Only use software approved by Schneider Electric for use with this equipment.
- Update your application program every time you change the physical hardware configuration.

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

# General Cybersecurity Information

In recent years, the growing number of networked machines and production plants has seen a corresponding increase in the potential for cyber threats, such as unauthorized access, data breaches, and operational disruptions. You must, therefore, consider all possible cybersecurity measures to help protect assets and systems against such threats.

To help keep your Schneider Electric products secure and protected, it is in your best interest to implement the cybersecurity best practices as described in the Cybersecurity Best Practices document.

Schneider Electric provides additional information and assistance:

- Subscribe to the Schneider Electric security newsletter.
- Visit the Cybersecurity Support Portal web page to:
  - Find Security Notifications.
  - Report vulnerabilities and incidents.
- Visit the Schneider Electric Cybersecurity and Data Protection Posture web page to:
  - Access the cybersecurity posture.
  - Learn more about cybersecurity in the cybersecurity academy.
  - Explore the cybersecurity services from Schneider Electric.

# Available Languages of the Document

The document is available in these languages:

- English (EIO0000003149)
- French (EIO0000003150)
- German (EIO0000003151)
- Spanish (EIO0000003152)
- Italian (EIO0000003153)
- Chinese (EIO0000003154)

# Related Documents

| Title of Documentation | Reference Number |
|---|---|
| EcoStruxure Machine Expert - Programming Guide | EIO0000002854 (ENG)<br>EIO0000002855 (FRE)<br>EIO0000002856 (GER)<br>EIO0000002857 (SPA)<br>EIO0000002858 (ITA)<br>EIO0000002859 (CHS) |
| Modicon M241 Logic Controller - Programming Guide | EIO0000003059 (ENG)<br>EIO0000003060 (FRA)<br>EIO0000003061 (GER)<br>EIO0000003062 (SPA)<br>EIO0000003063 (ITA)<br>EIO0000003064 (CHS) |

| Title of Documentation | Reference Number |
|---|---|
| Modicon M251 Logic Controller - Programming Guide | EIO0000003089 (ENG)<br>EIO0000003090 (FRA)<br>EIO0000003091 (GER)<br>EIO0000003092 (SPA)<br>EIO0000003093 (ITA)<br>EIO0000003094 (CHS) |
| TM4 Expansion Modules - Hardware Guide | EIO0000003155 (ENG)<br>EIO0000003156 (FRA)<br>EIO0000003157 (GER)<br>EIO0000003158 (SPA)<br>EIO0000003159 (ITA)<br>EIO0000003160 (CHS) |
| TM4 Expansion Modules - Instruction Sheet | EAV47886 |

To find documents online, visit the Schneider Electric download center (www.se.com/ww/en/download/).

# Information on Non-Inclusive or Insensitive Terminology

As a responsible, inclusive company, Schneider Electric is constantly updating its communications and products that contain non-inclusive or insensitive terminology. However, despite these efforts, our content may still contain terms that are deemed inappropriate by some customers.

# Terminology Derived from Standards

The technical terms, terminology, symbols and the corresponding descriptions in the information contained herein, or that appear in or on the products themselves, are generally derived from the terms or definitions of international standards.

In the area of functional safety systems, drives and general automation, this may include, but is not limited to, terms such as *safety*, *safety function*, *safe state*, *fault*, *fault reset*, *malfunction*, *failure*, *error*, *error message*, *dangerous*, etc.

Among others, these standards include:

| Standard | Description |
|---|---|
| IEC 61131-2:2007 | Programmable controllers, part 2: Equipment requirements and tests. |
| ISO 13849-1:2023 | Safety of machinery: Safety related parts of control systems.<br>General principles for design. |
| EN 61496-1:2020 | Safety of machinery: Electro-sensitive protective equipment.<br>Part 1: General requirements and tests. |
| ISO 12100:2010 | Safety of machinery - General principles for design - Risk assessment and risk reduction |
| EN 60204-1:2006 | Safety of machinery - Electrical equipment of machines - Part 1: General requirements |
| ISO 14119:2013 | Safety of machinery - Interlocking devices associated with guards - Principles for design and selection |
| ISO 13850:2015 | Safety of machinery - Emergency stop - Principles for design |

| Standard | Description |
|---|---|
| IEC 62061:2021 | Safety of machinery - Functional safety of safety-related electrical, electronic, and electronic programmable control systems |
| IEC 61508-1:2010 | Functional safety of electrical/electronic/programmable electronic safety-related systems: General requirements. |
| IEC 61508-2:2010 | Functional safety of electrical/electronic/programmable electronic safety-related systems: Requirements for electrical/electronic/programmable electronic safety-related systems. |
| IEC 61508-3:2010 | Functional safety of electrical/electronic/programmable electronic safety-related systems: Software requirements. |
| IEC 61784-3:2021 | Industrial communication networks - Profiles - Part 3: Functional safety fieldbuses - General rules and profile definitions. |
| 2006/42/EC | Machinery Directive |
| 2014/30/EU | Electromagnetic Compatibility Directive |
| 2014/35/EU | Low Voltage Directive |

In addition, terms used in the present document may tangentially be used as they are derived from other standards such as:

| Standard | Description |
|---|---|
| IEC 60034 series | Rotating electrical machines |
| IEC 61800 series | Adjustable speed electrical power drive systems |
| IEC 61158 series | Digital data communications for measurement and control – Fieldbus for use in industrial control systems |

Finally, the term *zone of operation* may be used in conjunction with the description of specific hazards, and is defined as it is for a *hazard zone* or *danger zone* in the *Machinery Directive* (*2006/42/EC*) and *ISO 12100:2010*.

**NOTE:** The aforementioned standards may or may not apply to the specific products cited in the present documentation. For more information concerning the individual standards applicable to the products described herein, see the characteristics tables for those product references.

# TM4 Description

# TM4 General Description

## Introduction

The range of TM4 expansion modules includes communication modules.

## TM4 Expansion Module Features

The table shows the TM4 expansion module features:

| Module Reference | Type | Terminal Type |
|---|---|---|
| TM4ES4 | Ethernet communication | 4 RJ45 connectors |
| TM4PDPS1 | PROFIBUS DP slave communication | 1 SUB-D 9 pins female connector |

# TM4 Expansion Modules Compatibility

## Introduction

This section describes the compatibility of TM4 expansion modules with controllers.

The TM4 bus supports up to 3 expansion modules. You can mix both Profibus DP (TM4PDPS1) and Ethernet (TM4ES4) expansion modules to the limit of 3 expansions.

## TM4ES4 Ethernet Module Compatibility

The TM4ES4 module has 2 applications:

- **Expansion**: addition of an Ethernet interface to extend the number of Ethernet ports for a controller,

  **NOTE:** If more than 1 TM4ES4 module is installed on the controller, the one closest to the controller is used as **expansion**.
- **Standalone**: Ethernet switch (only getting its power supply from the controller).

The table shows the TM4ES4 Ethernet module compatibility with controllers:

| Controller Reference | Expansion Usage Supported | Standalone Usage Supported | Maximum Number of TM4ES4 Modules |
|---|---|---|---|
| TM241C24R | Yes | Yes | 1 expansion + 2 standalone<br>OR<br>3 standalone |
| TM241CE24R | Yes | Yes | 1 expansion + 2 standalone<br>OR<br>3 standalone |

| Controller Reference | Expansion Usage Supported | Standalone Usage Supported | Maximum Number of TM4ES4 Modules |
|---|---|---|---|
| TM241CEC24R | Yes | Yes | 1 expansion + 2 standalone<br>OR<br>3 standalone |
| TM241C24T | Yes | Yes | 1 expansion + 2 standalone<br>OR<br>3 standalone |
| TM241CE24T | Yes | Yes | 1 expansion + 2 standalone<br>OR<br>3 standalone |
| TM241CEC24T | Yes | Yes | 1 expansion + 2 standalone<br>OR<br>3 standalone |
| TM241C24U | Yes | Yes | 1 expansion + 2 standalone<br>OR<br>3 standalone |
| TM241CE24U | Yes | Yes | 1 expansion + 2 standalone<br>OR<br>3 standalone |
| TM241CEC24U | Yes | Yes | 1 expansion + 2 standalone<br>OR<br>3 standalone |
| TM241C40R | Yes | Yes | 1 expansion + 2 standalone<br>OR<br>3 standalone |
| TM241CE40R | Yes | Yes | 1 expansion + 2 standalone<br>OR<br>3 standalone |
| TM241C40T | Yes | Yes | 1 expansion + 2 standalone<br>OR<br>3 standalone |
| TM241CE40T | Yes | Yes | 1 expansion + 2 standalone<br>OR<br>3 standalone |
| TM241C40U | Yes | Yes | 1 expansion + 2 standalone<br>OR<br>3 standalone |
| TM241CE40U | Yes | Yes | 1 expansion + 2 standalone<br>OR<br>3 standalone |
| TM251MESC | No | Yes | 3 standalone |
| TM251MESE | No | Yes | 3 standalone |
| **NOTE: Standalone** use does not require configuration in EcoStruxure Machine Expert. | | | |

# TM4PDPS1 PROFIBUS DP Expansion Module Compatibility

The TM4PDPS1 module is compatible with M241 and M251 logic controllers.

One TM4PDPS1 module can be added per controller.

# Adding a TM4 Expansion Module

## Adding a TM4 Expansion Module

To add a TM4 expansion module to your controller, select the expansion module in the **Hardware Catalog**, drag it to the **Devices tree**, and drop it on the **COM_Bus** node.

For more information on adding a device to your project, refer to:

• Using the Drag-and-drop Method found in the EcoStruxure Machine Expert Programming Guide

• Using the Contextual Menu or Plus Button found in the EcoStruxure Machine Expert Programming Guide

## Expansion Module Configuration

To configure your TM4 Expansion Module, double-click the expansion module node in the **Devices tree** to display the configuration tabs. The following chapters detail the configuration parameters.

**NOTE:** You do not configure the TM4ES4 when using it as a standalone switch. As such, the TM4ES4 module does not appear in the **Devices tree**.

# Connecting the Controller to a PC

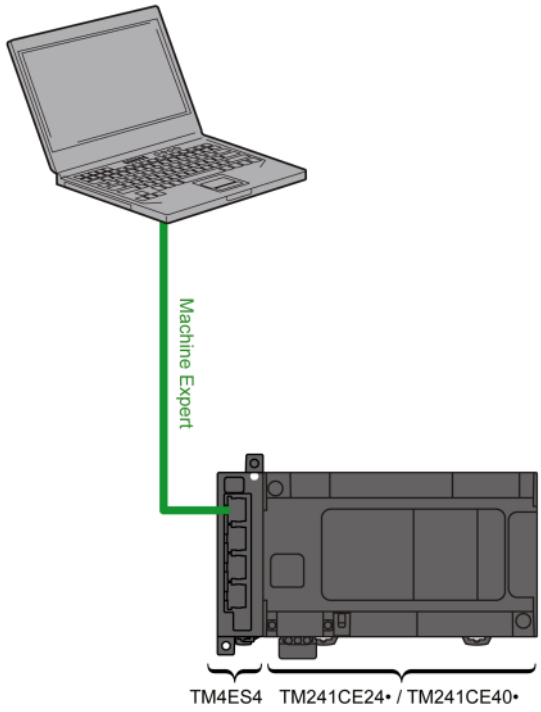
## Overview

To transfer, run, and monitor the applications, connect the controller to a computer that has EcoStruxure Machine Expert installed. Use either a USB cable or an Ethernet connection (for those references that support an Ethernet port).

| ***NOTICE*** |
| --- |
| **INOPERABLE EQUIPMENT**<br>Always connect the communication cable to the PC before connecting it to the controller.<br>**Failure to follow these instructions can result in equipment damage.** |

# Ethernet Port Connection

You can connect the controller to a PC using an Ethernet cable.




To connect the controller to the PC, do the following:

| Step | Action |
|---|---|
| 1 | Connect your Ethernet cable to the PC. |
| 2 | Connect your Ethernet cable to a free Ethernet port on the TM4ES4 expansion module. |

# TM4ES4 Ethernet Module

## Introduction

This chapter describes the configuration of the TM4ES4 Ethernet module when it is used as **Expansion**.

In **Standalone** use, the module does not require configuration in EcoStruxure Machine Expert, and therefore the information in this chapter is not applicable.

Refer to TM4ES4 Ethernet Module Compatibility, page 10 to know the application type according to the controller reference compatibility.

## Ethernet Services

### Introduction

This section describes how to configure the Ethernet services provided by the TM4ES4 expansion module.

### Presentation

#### Ethernet Services

The TM4ES4 expansion module provides an Ethernet interface to extend the number of Ethernet ports for a controller.

The module supports the following controller services:

- Modbus TCP Server/Client, page 19
- Web Server (see Modicon M241 Logic Controller, Programming Guide)
- FTP Server (see Modicon M241 Logic Controller, Programming Guide)
- SNMP (see Modicon M241 Logic Controller, Programming Guide)
- M241 Logic Controller as Target Device on EtherNet/IP, page 20
- M241 Logic Controller as Slave Device on Modbus TCP, page 22
- IEC VAR access, page 15

#### Ethernet Protocol

The Ethernet module supports the following protocols:

- IP (Internet Protocol)
- UDP (User Datagram Protocol)
- TCP (Transmission Control Protocol)
- ARP (Address Resolution Protocol)
- ICMP (Internet Control Messaging Protocol)
- IGMP (Internet Group Management Protocol)

## TCP Server Connections

This table shows the maximum number of TCP server connections for the controller and the TM4ES4 modules:

| Connection Type | Maximum Number of Simultaneous Server Connections |
|---|---|
| Modbus Server | 8 simultaneous TCP server connections maximum for TM4ES4 and controller or for the controller alone. |
| EtherNet/IP Device | 16 |
| FTP Server | 4 |
| Web Server | 10 |

Each server based on TCP manages its own set of connections.

When a client tries to open a Modbus Server connection that exceeds the maximum number of connections, the controller closes the oldest connection. In other cases, the attempt to open a connection is denied.

If all connections are busy (exchange in progress) when a client tries to open a new one, the new connection is denied.

The server connections stay open as long as the controller stays in operational states (*RUN*, *STOP*, *HALT*).

The server connections are closed when leaving or entering operational states (*RUN*, *STOP*, *HALT*), except in case of power outage (because the controller does not have time to close the connections).

For more information about the operational states, refer to the controller state diagram (see Modicon M241 Logic Controller, Programming Guide).

## Available Services

With an Ethernet communication, the **IEC VAR ACCESS** service is supported by the controller. The **IEC VAR ACCESS** service allows an exchange of variables between the controller and an HMI.

In addition, the **NetWork variables** service is supported by the controller. The **NetWork variables** service allows an exchange of data between controllers.

**NOTE:** For more information, refer to the EcoStruxure Machine Expert Programming Guide.

# IP Address Configuration

## Introduction

There are different ways to assign the IP address of the module:

- address assignment by DHCP server
- address assignment by BOOTP server
- fixed IP address
- post configuration file (see Modicon M241 Logic Controller, Programming Guide). If a post configuration file exits, this assignment method has priority over the others.

The IP address can also be changed dynamically through the:

- **Communication Settings** tab (see Modicon M241 Logic Controller, Programming Guide)
- **changeIPAddress** function block (see Modicon M241 Logic Controller, Programming Guide)

**NOTE:** If the attempted addressing method is unsuccessful, the link uses a default IP address, page 17 derived from the MAC address.

Carefully manage the IP addresses because each device on the network requires a unique address. Having multiple devices with the same IP address can cause unintended operation of your network and associated equipment.

**⚠WARNING**

**UNINTENDED EQUIPMENT OPERATION**

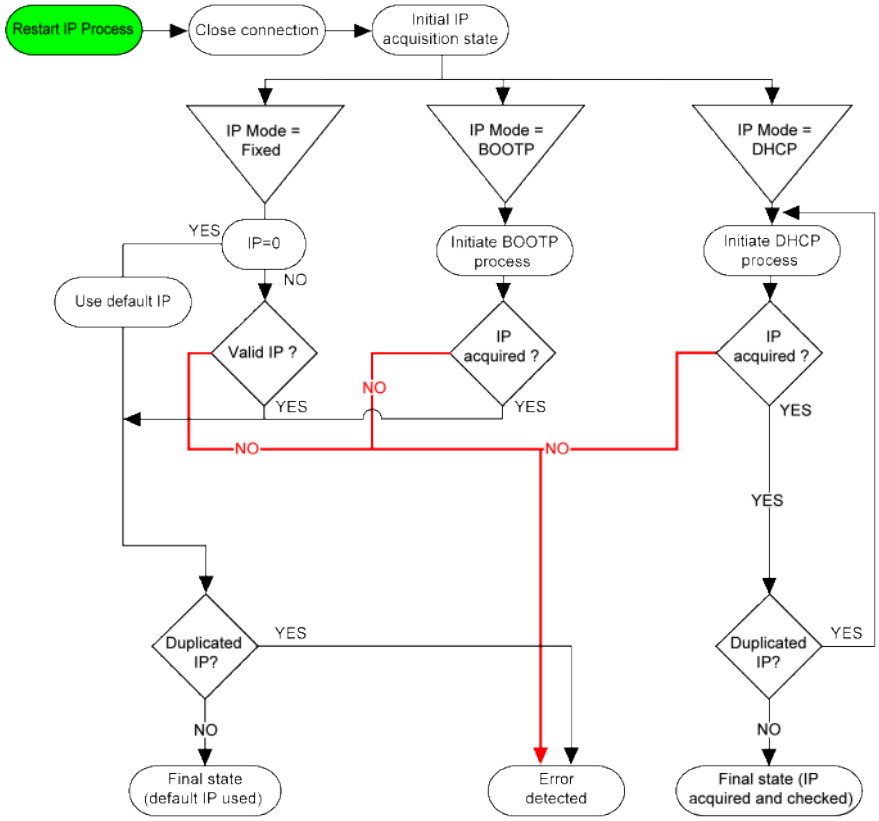
- Verify that there is only one master controller configured on the network or remote link.
- Verify that all devices have unique addresses.
- Obtain your IP address from your system administrator.
- Confirm that the IP address of the device is unique before placing the system into service.
- Do not assign the same IP address to any other equipment on the network.
- Update the IP address after cloning any application that includes Ethernet communications to a unique address.

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

**NOTE:** Verify that your system administrator maintains a record of all assigned IP addresses on the network and subnetwork, and inform the system administrator of all configuration changes performed.

## Address Management

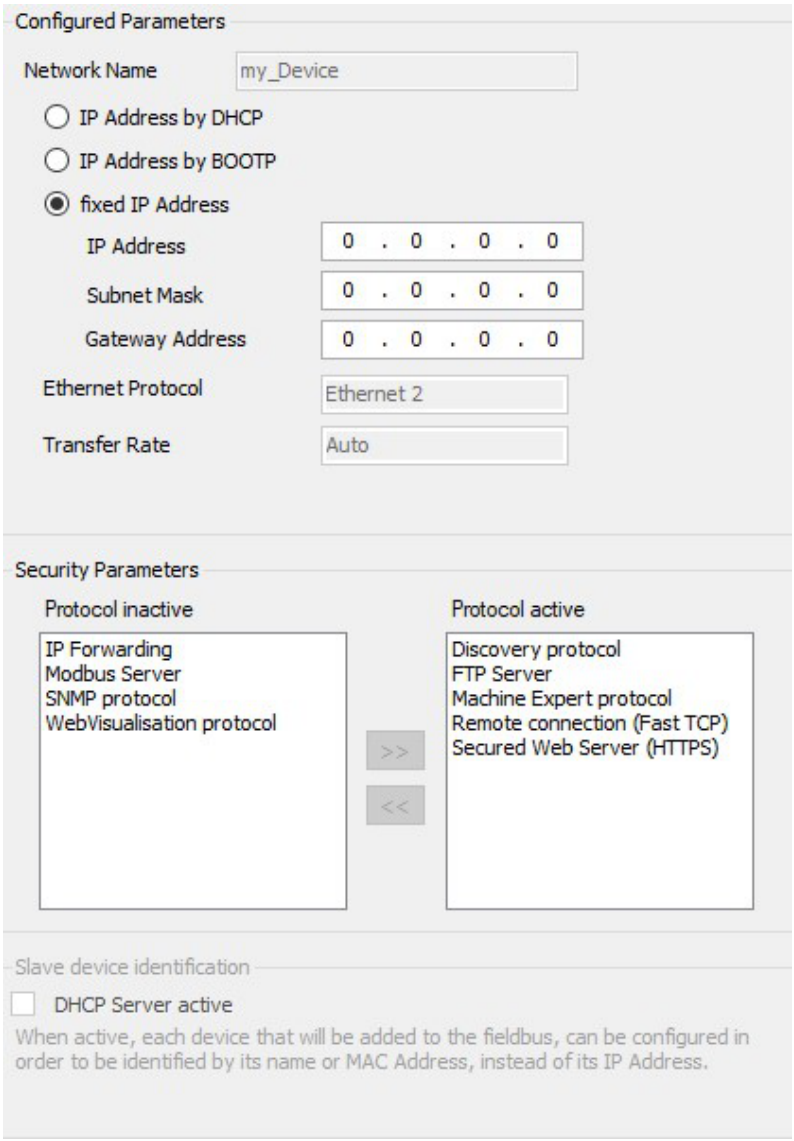
The different types of address systems for the controller are shown in the following diagram:



**NOTE:** If a device programmed to use the DHCP or BOOTP addressing methods is unable to contact its respective server, the module uses the default IP address. However, the process is repeated until the respective server is reached and an IP address is acquired.

## Ethernet Configuration

In the **Devices tree**, double-click **TM4ES4**:




**NOTE:**

- If you are in offline mode, you see the **Configured Parameters** window (displayed above). You can edit the parameters.
- If you are in online mode, you see the **Configured Parameters** and **Current Settings** windows. You cannot edit the parameters.

This table describes the configured parameters:

| Configured Parameters | Description |
|---|---|
| **Network Name** | Used as device name to retrieve IP address through DHCP, maximum 15 characters. |
| **IP Address by DHCP** | IP address is obtained via DHCP. |
| **IP Address by BOOTP** | IP address is obtained via BOOTP. |
| **Fixed IP Address** | IP address, Subnet mask and Gateway Address are defined by the user. |
| **Ethernet Protocol** | Protocol type used: Ethernet 2 |
| **Transfer Rate** | Transfer rate and direction on the bus are automatically configured. |

### Default IP Address

The IP address by default is 11.11.x.x.

The last 2 fields in the default IP address are composed of the decimal equivalent of the last 2 hexadecimal bytes of the MAC address of the module.

The MAC address of the module can be retrieved at the bottom of the front face of the module.

The default subnet mask is 255.0.0.0.

**NOTE:** A MAC address is always written in hexadecimal format, and an IP address in decimal format. You must convert the MAC address to decimal format.

Example: If the MAC address is 00.80.F4.01.**80.F2**, the default IP address is 11.11.**128.242**.

**NOTE:** To take into account the new IP address after the download of a project, reboot the controller by doing a power cycle.

### Subnet Mask

The subnet mask is used to address several physical networks with a single network address. The mask is used to separate the subnetwork and the device address in the host ID.

The subnet address is obtained by retaining the bits of the IP address that correspond to the positions of the mask containing 1, and replacing the others with 0.

Conversely, the subnet address of the host device is obtained by retaining the bits of the IP address that correspond to the positions of the mask containing 0, and replacing the others with 1.

Example of a subnet address:

| | | | | |
|---|---|---|---|---|
| IP address | 192 (11000000) | 1 (00000001) | 17 (00010001) | 11 (00001011) |
| Subnet mask | 255 (11111111) | 255 (11111111) | 240 (11110000) | 0 (00000000) |
| Subnet address | 192 (11000000) | 1 (00000001) | 16 (00010000) | 0 (00000000) |

**NOTE:** The device does not communicate on its subnetwork when there is no gateway.

### Gateway Address

The gateway allows a message to be routed to a device which is not on the current network.

If there is no gateway, the gateway address is 0.0.0.0.

### Security Parameters

This table describes the different security parameters:

| Security Parameters | Description | Default settings |
|---|---|---|
| **Discovery protocol** | This parameter activates/deactivates **Discovery protocol**. When deactivated, Discovery requests are ignored. | Active |
| **FTP Server** | This parameter activates/deactivates the **FTP Server** of the controller. When deactivated, FTP requests are ignored. | Active |
| **IP Forwarding** | This parameter activates/deactivates the **IP Forwarding** service of the controller. When deactivated, devices on the device network are no longer accessible from the control network (Web pages, DTM).<br>**NOTE:** This parameter is only available on the Ethernet_1 network. | Inactive |
| **Machine Expert protocol** | This parameter activates/deactivates the **Machine Expert protocol** on Ethernet interfaces. When deactivated, Machine Expert requests from any device are rejected, including those from the UDP or TCP connection. This means that no connection is possible on Ethernet from a programming PC, from an HMI target that wants to exchange variables with this controller, from an OPC server, or from Controller Assistant. | Active |
| **Modbus Server** | This parameter activates/deactivates the **Modbus Server** of the controller. When deactivated, Modbus requests to the controller are ignored. | Inactive |
| **SNMP protocol** | This parameter activates/deactivates the SNMP server of the controller. When deactivated, SNMP requests are ignored. | Inactive |
| **Remote connection (Fast TCP)** | This parameter activates/deactivates the remote connection. When deactivated, Fast TCP requests are ignored. | Active |
| **Secured Web Server (HTTPS)** | This parameter activates/deactivates the Secured Web server of the controller. When deactivated, HTTPS requests to the controller Secured Web server are ignored. | Active |
| **WebVisualisation protocol** | This parameter activates/deactivates the WebVisualisation pages of the controller. When deactivated, HTTP requests to the controller WebVisualisation protocol are ignored. | Inactive |

### Device Identification

When **DHCP Server active** is selected, devices added to the fieldbus can be configured to be identified by their name or MAC address, instead of their IP address. Refer to DHCP Server (see Modicon M241 Logic Controller, Programming Guide).

# Modbus TCP Server/Client

## Introduction

Unlike Modbus serial link, Modbus TCP/IP is not based on a hierarchical structure, but on a client/server model.

The TM4ES4 module implements both client and server services so that it can initiate communications to other controllers and I/O devices, and to respond to requests from other controllers, SCADA, HMIs and other devices. By default, Modbus Server functionality is not active.

Without any configuration, the TM4ES4 module supports Modbus server.

The Modbus Server/Client is included in the firmware, and does not require any programming action from the user. Due to this feature, it is accessible in RUNNING, STOPPED and EMPTY states.

## Modbus TCP Client

The Modbus TCP client supports the following function blocks from the PLCCommunication library without any configuration:

- ADDM
- READ_VAR
- SEND_RECV_MSG
- SINGLE_WRITE
- WRITE_READ_VAR
- WRITE_VAR

For further information, refer to the Function Block Descriptions (see Modbus and ASCII Read/Write Functions, PLCCommunication Library Guide).

## Modbus TCP Server

The Modbus server supports the following Modbus requests:

| Function Code<br>Dec (Hex) | Subfunction<br>Dec (Hex) | Function |
|---|---|---|
| 1 (1h) | – | Read digital outputs (%Q) |
| 2 (2h) | – | Read digital inputs (%I) |
| 3 (3h) | – | Read holding register (%MW) |
| 6 (6h) | – | Write single register (%MW) |
| 8 (8h) | – | Diagnostic |
| 15 (Fh) | – | Write multiple digital outputs (%Q) |
| 16 (10h) | – | Write multiple registers (%MW) |
| 23 (17h) | – | Read/write multiple registers (%MW) |
| 43 (2Bh) | 14 (Eh) | Read device identification |

## Diagnostic Request

The table contains the Data Selection Code list:

| Data Selection Code | Description |
|---|---|
| 0x00 | Reserved |
| 0x01 | Basic Network Diagnostics |
| 0x02 | Ethernet Port Diagnostic |
| 0x03 | Modbus TCP/Port 502 Diagnostics |
| 0x04 | Modbus TCP/Port 502 Connection Table |
| 0x05 - 0x7E | Reserved for other public codes |
| 0x7F | Data Structure Offsets |

# M241 Logic Controller as a Target Device on EtherNet/IP

## Introduction

This section describes the configuration of the M241 Logic Controller as an EtherNet/IP target device.

For further information about EtherNet/IP, refer to the www.odva.org website.

## EtherNet/IP Target Configuration

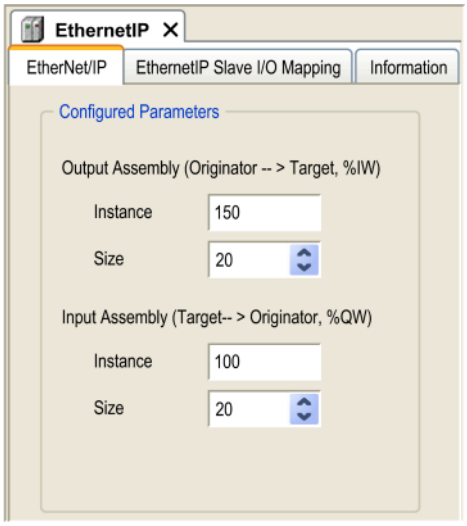
To configure your M241 Logic Controller as a target device on EtherNet/IP, you must:

| Step | Action |
|---|---|
| 1 | Select **EthernetIP** in the **Hardware Catalog**. |
| 2 | Drag and drop it to the **Devices tree** on one of the highlighted nodes.<br>**NOTE:** If the chosen node is **COM_Bus**, a TM4ES4 expansion module is automatically added to your configuration.<br>For more information on adding a device to your project, refer to:<br>• Using the Drag-and-drop Method found in the EcoStruxure Machine Expert Programming Guide<br>• Using the Contextual Menu or Plus Button found in the EcoStruxure Machine Expert Programming Guide |

## EtherNet/IP Parameter Configuration

To configure the EtherNet/IP parameters, double-click **EthernetIP** in the **Devices tree**.

This dialog box is displayed:




The EtherNet/IP configuration parameters are defined as:

- **Instance**:

  Number referencing the input or output Assembly.

- **Size**:

  Number of channels of an input or output Assembly.

  Each channel has a 2-byte memory that stores the value of an *%IWx* or *%QWx* object, where *x* is the channel number.

  For example, if the **Size** of the **Output Assembly** is 20, there are 20 input channels (IW0...IW19) addressing *%IWy...%IW*(y+20-1), where y is the first available channel for the Assembly.

| Element | | Admissible Controller Range | Default Value |
|---|---|---|---|
| **Output Assembly** | **Instance** | 150...189 | 150 |
| | **Size** | 2...250 | 20 |
| **Input Assembly** | **Instance** | 100...149 | 100 |
| | **Size** | 2...250 | 20 |

Refer to the M241 Logic Controller Programming Guide for more information on the following topics:

- Generating an EDS file
- Configuring I/Os
- Objects supported by the controller

# M241 Logic Controller as a Slave Device on Modbus TCP

## Overview

This section describes the configuration of the M241 Logic Controller as a **Modbus TCP Slave Device**.

To configure your M241 Logic Controller as a **Modbus TCP Slave Device**, you must add **Modbus TCP Slave Device** functionality to your controller (see Adding a Modbus TCP Slave Device, page 23).

This functionality creates a specific I/O area in the controller that is accessible with the Modbus TCP protocol. This I/O area is used whenever an external master needs to access the *%IW* and *%QW* objects of the controller. This **Modbus TCP Slave Device** functionality allows you to furnish to this area the controller I/O objects which can then be accessed with a single Modbus read/write registers request.

The **Modbus TCP Slave Device** adds another Modbus server function to the controller. This server is addressed by the Modbus client application by specifying a configured Unit ID (Modbus address) in the range 1...247. The embedded Modbus server of the slave controller needs no configuration, and is addressed by specifying a Unit ID equal to 255. Refer to Modbus TCP Configuration, page 23.

Inputs/outputs are seen from the slave controller: inputs are written by the master, and outputs are read by the master.

The **Modbus TCP Slave Device** can define a privileged Modbus client application, whose connection is not forcefully closed (embedded Modbus connections may be closed when more than 8 connections are needed).

The timeout duration associated to the privileged connection allows you to verify whether the controller is being polled by the privileged master. If no Modbus request is received within the timeout duration, the diagnostic information *i_byMasterIpLost* is set to 1 (TRUE). For more information, refer to the Ethernet Port Read-Only System Variables (see Modicon M241 Logic Controller, System Functions and Variables, PLCSystem Library Guide).

For further information about Modbus TCP, refer to the www.modbus.org website.

## Adding a Modbus TCP Slave Device

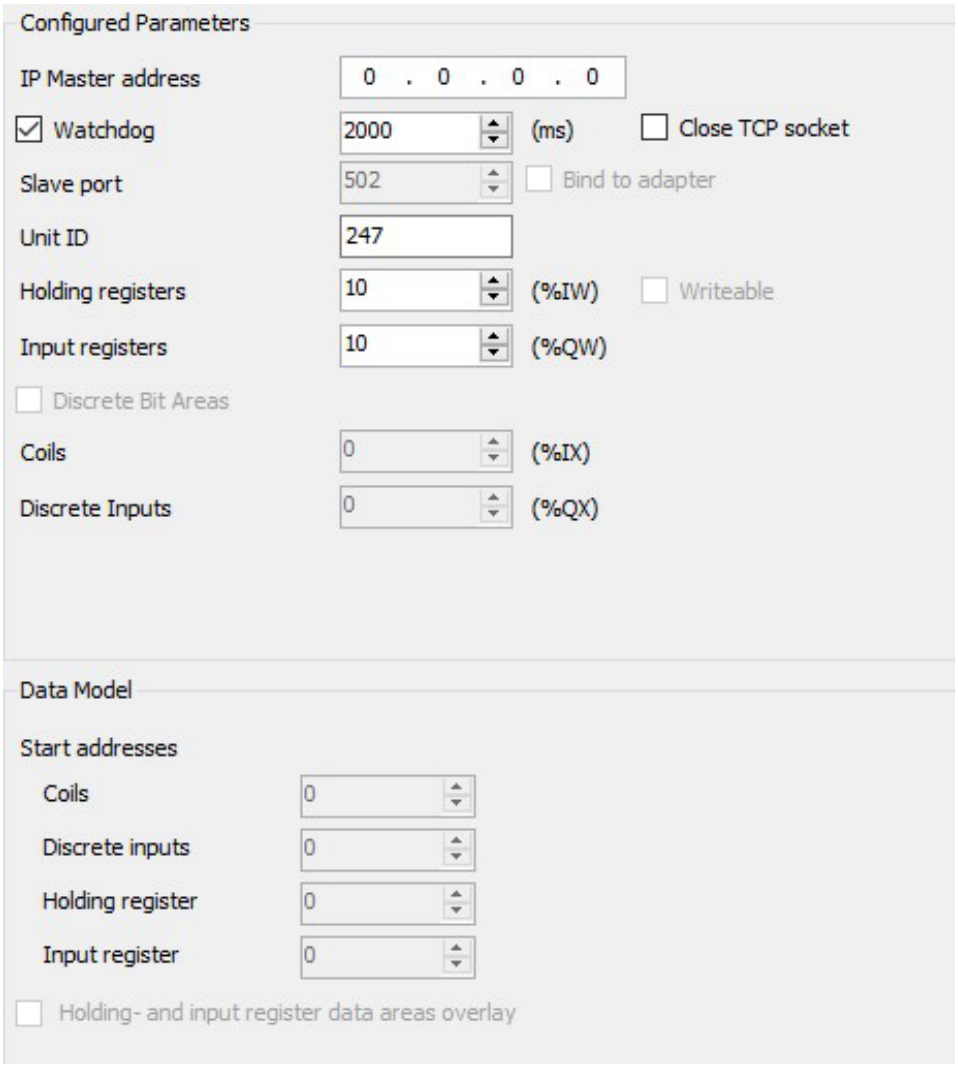
To add Modbus TCP slave device functionality to your M241 Logic Controller:

| Step | Action |
|---|---|
| 1 | Select **Modbus TCP Slave Device** in the **Hardware Catalog**. |
| 2 | Drag and drop it to the **Devices tree** on one of the highlighted nodes.<br>**NOTE:** If the chosen node is **COM_Bus**, a TM4ES4 expansion module is automatically added to your configuration.<br>For more information on adding a device to your project, refer to:<br>• Using the Drag-and-drop Method found in the EcoStruxure Machine Expert Programming Guide<br>• Using the Contextual Menu or Plus Button found in the EcoStruxure Machine Expert Programming Guide |

## Configuring a Modbus TCP Slave Device

To configure the Modbus TCP slave device, double-click **ModbusTCP_Slave_Device** in the **Devices tree**.

This dialog box appears:



| Element | Description |
|---|---|
| **IP Master Address** | IP address of the Modbus master<br>The connections are not closed on this address. |
| **Watchdog** | Watchdog in 500 ms increments<br>**NOTE:** The watchdog applies to the **IP Master Address** unless the address is 0.0.0.0. |
| **Close TCP socket** | When **Close TCP socket** is selected, the TCP socket is closed if the **Watchdog** is enabled and the set time is exceeded. |
| **Slave Port** | Modbus communication port (502) |
| **Unit ID** | Sends the requests to the Modbus TCP slave device (1...247), instead of the embedded Modbus server (255). |
| **Holding Registers (%IW)** | Number of %IW registers to be used in the exchange (2...120) (each register is 2 bytes) |
| **Input Registers (%QW)** | Number of %QW registers to be used in the exchange (2...120) (each register is 2 bytes) |

## Modbus TCP Slave Device I/O Mapping Tab

The I/Os are mapped to Modbus registers from the master perspective in the following way:

- %IWs are mapped from register 0 to n-1 and are R/W (n = Holding register quantity, each %IW register is 2 bytes).
- %QWs are mapped from register n to n+m -1 and are read only (m = Input registers quantity, each %QW register is 2 bytes).

When a **Modbus TCP Slave Device** has been configured, Modbus commands sent to its Unit ID (Modbus address) are handled differently than the same command would be when addressed to any other Modbus device on the network. For example, when the Modbus command 3 (3 hex) is sent to a standard Modbus device, it reads and returns the value of one or more registers. When this same command is sent to the Modbus TCP Slave (see M241 Logic Controller, Programming Guide), it facilitates a read operation by the external I/O scanner.

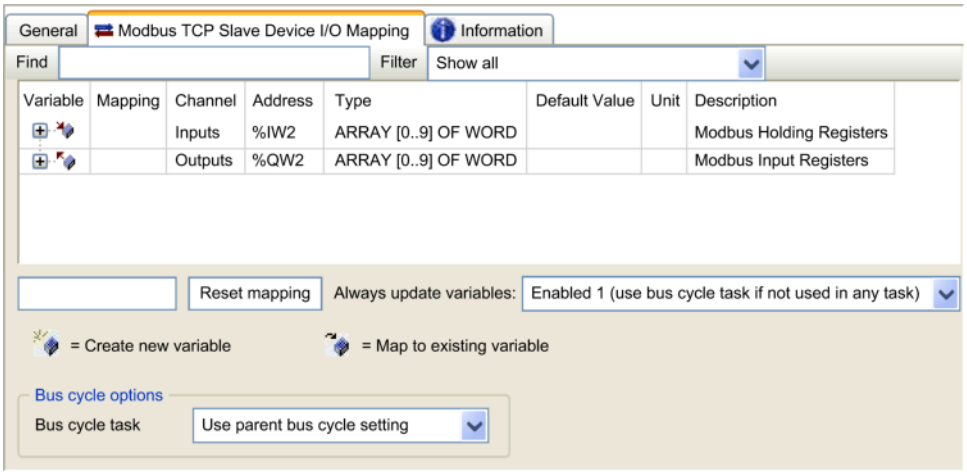
When a **Modbus TCP Slave Device** has been configured, Modbus commands sent to its Unit ID (Modbus address) access the *%IW* and *%QW* objects of the controller instead of the regular Modbus words (accessed when the Unit ID is 255). This facilitates read/write operations by a Modbus TCP IOScanner application.

The **Modbus TCP Slave Device** responds to a subset of the Modbus commands with the purpose of exchanging data with the external I/O scanner. The following Modbus commands are supported by the **Modbus TCP Slave Device**:

| Function Code Dec (Hex) | Function | Comment |
|---|---|---|
| 3 (3) | Read holding register | Allows the master to read *%IW* and *%QW* objects of the device |
| 6 (6) | Write single register | Allows the master to write *%IW* objects of the device |
| 16 (10) | Write multiple registers | Allows the master to write *%IW* objects of the device |
| 23 (17) | Read/write multiple registers | Allows the master to read *%IW* and *%QW* objects of the device and write *%IW* objects of the device |
| Other | Not supported | – |

**NOTE:** Modbus requests that attempt to access registers above n+m-1 are answered by the 02 - ILLEGAL DATA ADDRESS exception code.

To link I/O objects to variables, select the **Modbus TCP Slave Device I/O Mapping** tab:




| Channel | | Type | Description |
|---|---|---|---|
| **Input** | IW0 | WORD | Holding register 0 |
| | ... | ... | ... |
| | IWx | WORD | Holding register x |
| **Output** | QW0 | WORD | Input register 0 |
| | ... | ... | ... |
| | QWy | WORD | Input register y |

The number of words depends on the **Holding Registers (%IW)** and **Input Registers (%QW)** parameters of the **Modbus TCP** tab.

**NOTE:** Output means OUTPUT from Originator controller (%IW for the server/slave controller). Input means INPUT from Originator controller (%QW for the server/slave controller).

**NOTE:** The **Modbus TCP Slave Device** refreshes the *%IW* and *%QW* registers as a single time-consistent unit, synchronized with the IEC tasks (MAST task by default). By contrast, the embedded Modbus TCP server only ensures time-consistency for one word (2 bytes). If your application requires time-consistency for more than one word (2 bytes), use the **Modbus TCP Slave Device**.

For the parameter **Always update variables**, choose one of the following options:

- **Use parent device setting**
- **Enabled 1 (use bus cycle task if not used in any task)** (default setting)
- **Enabled 2 (always in bus cycle task)**

## Bus Cycle Options

In the **Modbus TCP Slave Device I/O Mapping** tab, select the **Bus cycle task** to use:

- **Use parent bus cycle setting** (default setting)
- **MAST**
- **An existing task of the project**: you can select an existing task and associate it to the scanner. For more information about the application tasks, refer to the EcoStruxure Machine Expert Programming Guide.

**NOTE:** There is a corresponding **Bus cycle task** parameter in the I/O mapping editor of the device that contains the **Modbus TCP Slave Device**. This parameter defines the task responsible for refreshing the *%IW* and *%QW* registers.

# Firewall Configuration

## Introduction

This section describes how to configure the firewall of the Modicon M241 Logic Controller.

## Introduction

### Firewall Presentation

In general, firewalls help protect network security zone perimeters by blocking unauthorized access and permitting authorized access. A firewall is a device or set of devices configured to permit, deny, encrypt, decrypt, or proxy traffic between different security zones based upon a set of rules and other criteria.

Process control devices and high-speed manufacturing machines require fast data throughput and often cannot tolerate the latency introduced by an aggressive security strategy inside the control network. Firewalls, therefore, play a significant role in a security strategy by providing levels of protection at the perimeters of the network. Firewalls are an important part of an overall, system level strategy.

**NOTE:** Schneider Electric adheres to industry best practices in the development and implementation of control systems. This includes a "Defense-in-Depth" approach to secure an Industrial Control System. This approach places the controllers behind one or more firewalls to restrict access to authorized personnel and protocols only.

**⚠WARNING**

**UNAUTHENTICATED ACCESS AND SUBSEQUENT UNAUTHORIZED MACHINE OPERATION**

- Evaluate whether your application environments are connected to your critical infrastructure and, if so, take appropriate steps in terms of prevention, based on Defense-in-Depth, before connecting the automation system to any network.
- Limit the number of devices connected to a network to the minimum necessary.
- Isolate your industrial network from other networks inside your company.
- Protect any network against unintended access by using firewalls, VPN, or other, proven security measures, such as an Intrusion Prevention System or Intrusion Detection System.
- Monitor activities within your systems.
- Prevent subject devices from direct access or direct link by unauthorized parties or unauthenticated actions.
- Install certificates that are issued by publicly known Trusted Certificate Authorities.
- Keep your systems up-to-date and rely only on legitimate sources.
- Prepare a recovery plan including backup of your system and process information.

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

For more information on organizational measures and rules covering access to infrastructures, refer to ISO/IEC 27000 series, Common Criteria for Information Technology Security Evaluation, ISO/IEC 15408, IEC 62351, ISA/IEC 62443, NIST Cybersecurity Framework, Information Security Forum - Standard of Good Practice for Information Security and refer to Cybersecurity Guidelines for

EcoStruxure Machine Expert, Modicon and PacDrive Controllers and Associated Equipment.

## Firewall Configuration

There are three ways to manage the controller firewall configuration:

- Static configuration
- Dynamic changes
- Application settings

Script files are used in the static configuration and for dynamic changes.

## Static Configuration

The static configuration is loaded at the controller boot.

The controller firewall can be statically configured by managing a default script file located in the controller. The path to this file is */usr/Cfg/FirewallDefault.cmd*.

**NOTE:** The file name is case sensitive.

## Dynamic Changes

After the controller boot, the controller firewall configuration can be changed by the use of script files.

There are two ways to load these dynamic changes using:

- A physical SD card, page 27.
- A function block, page 28 in the application.

## Application Settings

See Ethernet Configuration (see Modicon M241 Logic Controller, Programming Guide).

# Dynamic Changes Procedure

## Using an SD Card

This table describes the procedure to execute a script file from an SD card:

| Step | Action |
|---|---|
| 1 | Create a valid script file, page 29.<br>For example, name the script file *FirewallMaintenance.cmd*. |
| 2 | Load the script file on the SD card.<br>For example, load the script file in the *usr/Cfg* folder. |
| 3 | In the file *Sys/Cmd/Script.cmd*, add a code line with the command<br>`Firewall_install "/pathname/FileName"`<br>For example, the code line is<br>`Firewall_install "/sd0/usr/Cfg/FirewallMaintenance.cmd"`<br>**NOTE:** The file name is case sensitive. |
| 4 | Insert the SD card on the controller. |

## Using a Function Block in the Application

This table describes the procedure to execute a script file from an application:

| Step | Action |
|---|---|
| 1 | Create a valid script file, page 29.<br>For example, name the script file *FirewallMaintenance.cmd*. |
| 2 | Load the script file in the controller memory.<br>For example, load the script file in the *usr/Syslog* folder with FTP. |
| 3 | Use an ExecuteScript (see Modicon M241 Logic Controller, System Functions and Variables, PLCSystem Library Guide) function block.<br>For example, the **[SCmd]** input is `'Firewall_install "/usr/Syslog/FirewallMaintenance.cmd"'`<br>**NOTE:** The file name is case sensitive. |

# Firewall Behavior

## Introduction

The firewall configuration depends on the action done on the controller and the initial configuration state. There are five possible initial states:

- There is no default script file in the controller.
- A correct script file is present.
- An incorrect script file is present.
- There is no default script file and the application has configured the firewall.
- A dynamic script file configuration has already been executed.

**NOTE:** To determine whether the firewall is configured and enabled, consult the message logger.

## No Default Script File

| If... | Then ... |
|---|---|
| Boot of the controller | Firewall is not configured. No protection is activated. |
| Execute dynamic script file | Firewall is configured according to the dynamic script file. |
| Execute dynamic incorrect script file | Firewall is not configured. No protection is activated. |
| Download application | Firewall is configured according to the application settings. |

## Default Script File Present

| If... | Then ... |
|---|---|
| Boot of the controller | Firewall is configured according to the default script file. |
| Execute dynamic script file | The whole configuration of the default script file is deleted.<br>Firewall is configured according to the dynamic script file. |
| Execute dynamic incorrect script file | Firewall is configured according to the default script file.<br>The dynamic script file is not taken into account. |
| Download application | The whole configuration of the application is ignored.<br>Firewall is configured according to the default script file. |

## Incorrect Default Script File Present

| If... | Then ... |
|---|---|
| Boot of the controller | Firewall is not configured. No protection is activated |
| Execute dynamic script file | Firewall is configured according to the dynamic script file. |
| Execute dynamic incorrect script file | Firewall is not configured. No protection is activated. |
| Download application | Firewall is configured according to the application settings. |

## Application Settings with No Default Script File

| If... | Then ... |
|---|---|
| Boot of the controller | Firewall is configured according to the application settings. |
| Execute dynamic script file | The whole configuration of the application settings is deleted.<br>Firewall is configured according to the dynamic script file. |
| Execute dynamic incorrect script file | Firewall is configured according to the application settings. The dynamic script file is not taken into account. |
| Download application | The whole configuration of the previous application is deleted.<br>Firewall is configured according to the new application settings. |

## Execute Dynamic Script File Already Executed

| If... | Then ... |
|---|---|
| Boot of the controller | Firewall is configured according to the dynamic script file configuration (see note). |
| Execute dynamic script file | The whole configuration of the previous dynamic script file is deleted.<br>Firewall is configured according to the new dynamic script file. |
| Execute dynamic incorrect script file | Firewall is configured according to the previous dynamic script file configuration. The dynamic incorrect script file is not taken into account. |
| Download application | The whole configuration of the application is ignored<br>Firewall is configured according to the dynamic script file. |
| **NOTE:** If an SD card containing a cybersecurity script is plugged into the controller, booting is blocked. First remove the SD card to correctly boot the controller. | |

# Firewall Script Commands

## Overview

This section describes how script files (default script files or dynamic script files) are written so that they can be executed during the booting of the controller or during a specific command triggered.

**NOTE:** The MAC layer rules are managed separately and have more priority over other packet filter rules.

## Script File Syntax

The syntax of script files is described in Script Syntax Guidelines.

# General Firewall Commands

The following commands are available to manage the Ethernet firewall of the M241 Logic Controller:

| Command | Description |
|---|---|
| `Firewall Enable` | Blocks the frames from the Ethernet interfaces. If no specific IP address is authorized, it is not possible to communicate on the Ethernet interfaces.<br>**NOTE:** By default, when the firewall is enabled, the frames are rejected. |
| `Firewall Disable` | Firewall rules are not applied. Frames are not blocked. |
| `Firewall Ethx Default Allow` (1) | Frames are accepted by the controller. |
| `Firewall Ethx Default Reject`(1) | Frames are rejected by the controller.<br>**NOTE:** By default, if this line is not present, it corresponds to the command `Firewall Eth1 Default Reject`. |
| **(1)** Where Ethx =<br>• Eth1: Ethernet_1<br>• Eth2: TM4ES4 | |

## Specific Firewall Commands

The following commands are available to configure firewall rules for specific ports and addresses:

| Command | Range | Description |
|---|---|---|
| `Firewall Eth1 Allow IP •.•.•.•` | • = 0...255 | Frames from the specified IP address are allowed on all port numbers and port types. |
| `Firewall Eth1 Reject IP •.•.•.•` | • = 0...255 | Frames from the specified IP address are rejected on all port numbers and port types. |
| `Firewall Eth1 Allow IPs •.•.•.• to •.•.•.•` | • = 0...255 | Frames from the IP addresses in the specified range are allowed for all port numbers and port types. |
| `Firewall Eth1 Reject IPs •.•.•.• to •.•.•.•` | • = 0...255 | Frames from the IP addresses in the specified range are rejected for all port numbers and port types. |
| `Firewall Eth1 Allow port_type port Y` | Y = (destination port numbers) | Frames with the specified destination port number are allowed. |
| `Firewall Eth1 Reject port_type port Y` | Y = (destination port numbers) | Frames with the specified destination port number are rejected.<br>**NOTE:** When IP forwarding is activated, rules with reject port only filter frames with current controller as destination. They are not applied for the frames routed by the current controller. |
| `Firewall Eth1 Allow port_type ports Y1 to Y2` | Y = (destination port numbers) | Frames with a destination port number in the specified range are allowed. |
| `Firewall Eth1 Reject port_type ports Y1 to Y2` | Y = (destination port numbers) | Frames with a destination port number in the specified range are rejected. |
| `Firewall Eth1 Allow IP •.•.•.• on port_type port Y` | • = 0...255<br>Y = (destination port numbers) | Frames from the specified IP address and with the specified destination port number are allowed. |
| `Firewall Eth1 Reject IP •.•.•.• on port_type port Y` | • = 0...255<br>Y = (destination port numbers) | Frames from the specified IP address and with the specified destination port number are rejected. |
| `Firewall Eth1 Allow IP •.•.•.• on port_type ports Y1 to Y2` | • = 0...255<br>Y = (destination port numbers) | Frames from the specified IP address and with a destination port number in the specified range are allowed. |
| `Firewall Eth1 Reject IP •.•.•.• on port_type ports Y1 to Y2` | • = 0...255<br>Y = (destination port numbers) | Frames from the specified IP address and with a destination port number in the specified range are rejected. |
| `Firewall Eth1 Allow IPs •1.•1.•1.•1 to •2.•2.•2.•2 on port_type port Y` | • = 0...255<br>Y = (destination port numbers) | Frames from an IP address in the specified range and with the specified destination port number are allowed. |
| `Firewall Eth1 Reject IPs •1.•1.•1.•1 to •2.•2.•2.•2 on port_type port Y` | • = 0...255<br>Y = (destination port numbers) | Frames from an IP address in the specified range and with the specified destination port number are rejected. |
| `Firewall Eth1 Allow IPs •1.•1.•1.•1 to •2.•2.•2.•2 on port_type ports Y1 to Y2` | • = 0...255<br>Y = (destination port numbers) | Frames from an IP address in the specified range and with a destination port number in the specified range are allowed. |
| `Firewall Eth1 Reject IPs •1.•1.•1.•1 to •2.•2.•2.•2 on port_type ports Y1 to Y2` | • = 0...255<br>Y = (destination port numbers) | Frames from an IP address in the specified range and with a destination port number in the specified range are rejected. |
| `Firewall Eth1 Allow MAC ••:••:••:••:••:••` | • = 0...F | Frames from the specified MAC address ••:••:••:••:•• are allowed.<br>**NOTE:** When the rules to allow the MAC address are applied, only the listed MAC addresses can communicate with the controller, even if other rules are allowed. |
| `Firewall Eth1 Reject MAC ••:••:••:••:••:••` | • = 0...F | Frames with the specified MAC address ••:••:••:••:•• are rejected. |

**NOTE:** The `port_type` can be TCP or UDP.

## Script Example

```
; Enable FireWall. All frames are rejected;

FireWall Enable;

; Allow frames on Eth1

FireWall Eth1 Default Allow;

; Block all Modbus Requests on all IP address

Firewall Eth1 Reject tcp port 502;

; Reject frames on Eth2

FireWall Eth2 Default Reject;

; Allow Fast TCP on interface ETH1. This allow to connect to the
controller using TCP

Firewall Eth1 Allow TCP port 11740;

; Allow FTP active connection for IP address 85.16.0.17

FireWall Eth2 Allow IP 85.16.0.17 on tcp ports 20 to 21;
```

**NOTE:** IP addresses are converted to CIDR format.

For example:

"`FireWall Eth2 Allow IPs 192.168.100.66 to 192.168.100.99 on tcp port 44818;`", is separated into 7:

- `192.168.100.66/31`
- `192.168.100.68/30`
- `192.168.100.72/29`
- `192.168.100.80/28`
- `192.168.100.96/27`
- `192.168.100.128/26`
- `192.168.100.192/29`

To prevent a firewall error, use the entire subnet configuration.

**NOTE:** Characters are limited to 200 per line, including comments.

# Used Ports

This table lists the port numbers used by the M241 Logic Controller services:

| Service | Port numbers | Default configuration (1) |
|---|---|---|
| Programming software | TCP 1105, 11740 (Fast TCP)<br>UDP 1740 | Yes |
| FTP over TLS | TCP 21 | Yes |
| HTTP (2) | TCP 80 | Yes |
| HTTPS | TCP 443 | Yes |
| Modbus TCP | TCP 502 | No |
| OPC UA | TCP 4840 | No |
| Discovery | UDP 27126, 27127 | Yes |
| SNMP | UDP 161 | No |
| NVL | UDP 1202 | No |
| EtherNet/IP | TCP 44818<br>UDP 2222 | No |
| WebVisualisation (2) | TCP 8080 (HTTP), 8089 (HTTPS) | No |
| DHCP Client | UDP 67 | No |
| DHCP Server | UDP 68 | Yes |
| TFTP | UDP 69 | No |

**(1)** The default configuration is used when a default project is loaded into the controller. The port status is the same as the factory settings.

**(2)** HTTP is automatically redirected to HTTPS.

# TM4PDPS1 PROFIBUS DP Slave Module

## Introduction

This chapter describes the configuration of the TM4PDPS1 PROFIBUS DP slave module.

## PROFIBUS DP Slave Module Configuration

### Add a PROFIBUS DP Slave Module

#### Overview

With the PROFIBUS protocol the data is exchanged according to the master-slave principle. Only the master can initialize communication. The slaves respond to requests from masters. Several masters can coexist on the same bus. In this case, the slave I/O can be read by all the masters. However, a single master has write access to the outputs. The number of data items exchanged is defined during the configuration.

For the PROFIBUS master, the GSD file of the TM4PDPS1 module is located on *Drive:\Program Files\Schneider Electric\EcoStruxure Machine Expert Software \V2.3\LogicBuilder\GSD\SE100E83.GSD*.

The GSD file is also available on www.se.com/ww/en/download/.

There are 2 types of exchange services supported by this module:

- I/O cyclic frames exchanges, page 36
- Acyclic data exchanges with Profibus DPV1 function, page 38

#### Add a PROFIBUS DP Slave Module

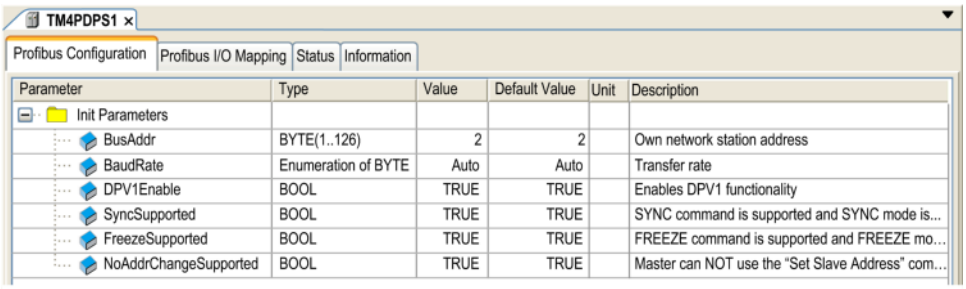
Refer to Adding a TM4 Expansion Module, page 12.

**NOTE:** Adding PROFIBUS increases the associated task cycle time by several milliseconds and the starting time by several seconds.

### Configure the PROFIBUS DP Slave Module

#### PROFIBUS DP Slave Module Configuration

In the **Devices tree**, double-click **My Controller > COM_Bus > TM4PDPS1**:



The following parameters are provided in the **Profibus Configuration** tab:

| Parameter | Value | Default Value | Description |
|---|---|---|---|
| **BusAddr** | **1...126** | **2** | PROFIBUS DP slave address<br>The address 126 is reserved. |
| **BaudRate** (kBaud) | **9.6**<br>**19.2**<br>**45.45**<br>**93.75**<br>**187.5**<br>**500**<br>**1500**<br>**3000**<br>**6000**<br>**12000**<br>**Auto** | **Auto** | PROFIBUS transmission rate |
| **DPV1Enable** | **TRUE**<br>**FALSE** | **TRUE** | **TRUE** indicates that the Profibus DPV1 functions for acyclic communication, page 38 is enabled. |
| **SyncSupported** | **TRUE**<br>**FALSE** | **TRUE** | **TRUE** indicates that the Synchronization mode is enabled. |
| **FreezeSupported** | **TRUE**<br>**FALSE** | **TRUE** | **TRUE** indicates that the Freeze mode is enabled. |
| **NoAddrChangeSupported** | **TRUE**<br>**FALSE** | **TRUE** | **TRUE** indicates that the PROFIBUS master cannot change the address. |

# Input / Output Devices Objects

## Introduction

To exchange data between the controller and a PROFIBUS master, it is important to understand the role of the TM4PDPS1 module.

The TM4PDPS1 module is an intermediate between the PROFIBUS master and the controller, and data is exchanged by using virtual I/O devices that you define when configuring the TM4PDPS1 module. The virtual devices are not physical I/O modules, but are logical input and output objects within the TM4PDPS1 module that you can then map to memory within the controller. These input and output objects are read from and written to by the PROFIBUS master. In turn, the module reads and writes this data to I/O memory locations in the controller so that you can use the data within your application program.

## Virtual I/O Devices

The virtual I/O devices you define within the TM4PDPS1 module can be either input or output, and can vary in size as defined by the table:

| Name | Number of I/O | Format |
|---|---|---|
| 12 word input (5B hex) | 12 | WORD |
| 12 word output (6B hex) | 12 | WORD |
| 16 byte input (1F hex) | 16 | BYTE |
| 16 byte output (2F hex) | 16 | BYTE |
| 2 byte input (11 hex) | 2 | BYTE |
| 2 byte output (21 hex) | 2 | BYTE |
| 2 word input (51 hex) | 2 | WORD |
| 2 word output (61 hex) | 2 | WORD |
| 20 word input (40 hex, 53 hex) | 20 | WORD |
| 20 word output (80 hex, 53 hex) | 20 | WORD |
| 32 word input (40 hex, 5F hex) | 32 | WORD |
| 32 word output (80 hex, 5F hex) | 32 | WORD |
| 4 word input (53 hex) | 4 | WORD |
| 4 word output (63 hex) | 4 | WORD |
| 8 byte input (17 hex) | 8 | BYTE |
| 8 byte output (27 hex) | 8 | BYTE |
| 8 word input (57 hex) | 8 | WORD |
| 8 word output (67 hex) | 8 | WORD |

Once you have defined these virtual input and/or output devices within the TM4PDPS1 expansion module, you can then map these devices to memory locations within the controller. The type of memory objects you map these virtual I/O devices to depends on the type of exchange you define between the master and the slave.

# Data Exchange

## Introduction

This section provides further information on the exchange of data between the TM4PDPS1 module and the PROFIBUS master.

## I/O Cyclic Exchange

### Introduction

In order to exchange input / output data between the PROFIBUS DP slave module and the PROFIBUS master in a cyclic way, define the variables in the **Profibus-Modules I/O Mapping** tab.

The %IW addresses of the controller are the output values supplied by the PROFIBUS DP master.

The %QW addresses of the controller are applied to the input of the PROFIBUS DP master.

**NOTE:** When you use the PROFIBUS module TM4PDPS1, it is mandatory to:

- configure a dedicated PROFIBUS task without watchdog (do not use the MAST task)
- assign the dedicated PROFIBUS task a lower priority than the MAST task (for example, if the MAST task has a priority value 1, the TaskProfibus must have a priority value 10)
- not set the PROFIBUS task cycle time faster than 10 ms. The typical cycle time of the bus cycle task is 10 ms.

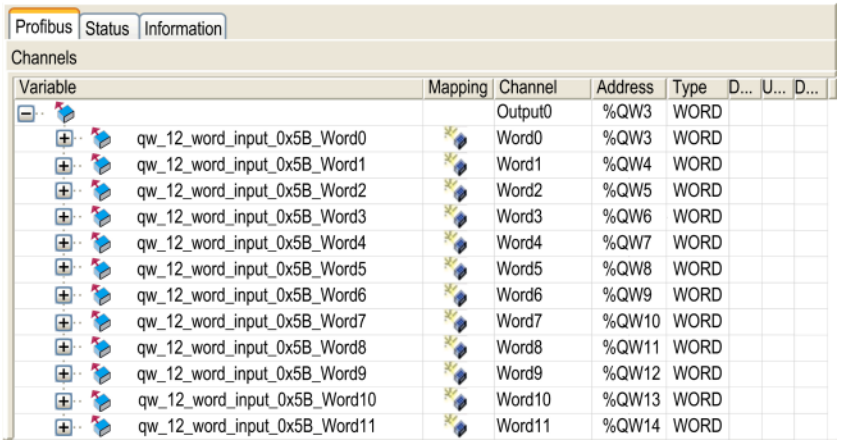
For more information about PROFIBUS task configuration, refer to the chapter PROFIBUS – Bus Cycle Task (see CODESYS PROFIBUS Online Help).

## I/O Mapping Table Creation

To create your I/O mapping table for the TM4PDPS1, proceed as follows:

| Step | Action |
|---|---|
| 1 | Select the **Devices & Modules** tab in the **Hardware Catalog** and click **Communication**. |
| 2 | Expand the **Profibus** node, choose the I/O device to add and drag-and-drop it onto TM4PDPS1.<br><br>**Result:** The module is added to **My Controller > COM_Bus > TM4PDPS1** area of the **Devices tree**. |

The variables for the exchange are automatically created in the %IWx and %QWx of the **Profibus I/O Mapping** tab. Double-click the I/O device you added to access this screen:




The tabs of the configuration window are described in the table below:

| Tab Name | Description |
|---|---|
| **Profibus I/O Mapping** | This tab contains the variables for data exchange. |
| **Status** | This tab provides diagnostic information, page 39. |
| **Information** | This tab provides further information on the selected input or output module. |

## PROFIBUS Virtual I/O Behavior

The table describes the status of the PROFIBUS I/O depending on:

- the controller status
- the PROFIBUS communication state (value of **PROFIBUS_R.i_CommState** of **PLCSystem** library)

| Controller State | Controller PROFIBUS I/O State |
|---|---|
| STOPPED | The %QW addresses are managed as it is configured in the **PLC Settings** tab of the controller configuration screen.<br>The %IW addresses are managed as it is configured in the **PLC Settings** tab of the controller configuration screen. |
| RUNNING | The %IW addresses are updated by the master.<br>The %QW addresses are sent to the master. |
| HALT | The %QW addresses are managed as it is configured in the **PLC Settings** tab of the controller configuration screen.<br>The %IW addresses keep the last correct value sent by the master. |

| Communication Status | Value of PROFIBUS_R.i_ CommState | Controller PROFIBUS I/O State |
|---|---|---|
| PROFIBUS Master is stopped | 4 (Operate mode) | The %IW addresses are set to 0 by the master.<br>The %QW addresses are sent to the master. |
| Watchdog is detected | 2 (Stop) | The %QW addresses are not sent to the master.<br>The %IW addresses keep the last correct value sent by the master. |

# Acyclic Exchange with PROFIBUS DPV1 Functions

## Introduction

The PROFIBUS DPV1 enhancement additionally supports acyclic data exchange between a PROFIBUS DPV1 master and DPV1 slaves. It allows access to %MW variables.

To use these functions between a PROFIBUS DPV1 master and the TM4PDPS1 module, the parameter **DPV1Enable** must be set to TRUE (default value), page 34.

## Data Addressing

Data addressing in the logic controller is %MW.

The **Profibus status** of the controller must be in **Operate** state; therefore it can be updated even if the logic controller is not running.

The %MW variables are automatically updated by the I/O driver whenever a DPV1 message is received.

It is based on PROFIBUS DPV1 read and write functions.

The logic address is the number of the %MW addressed.

## Direct / Indirect Addressing

Two different types of addressing are available for acyclic exchange:

| Addressing Type | Number of Requests for Read/Write %MW Variables | Description |
|---|---|---|
| Direct addressing | 1 | The address of the %MW variable is coded directly by **Slot** and **Index** fields.<br>The following restrictions apply to direct addressing:<br>• **Slot** field (**DU1**): FF hex value is not allowed.<br>• **Index** field (**DU2**): FF hex, E9 hex, and EA hex values are not allowed. |
| Indirect addressing | 2 | • The first request sends the address of the first %MW that the master will read or write.<br>• The second request reads or writes one or several values of the %MW variable. |

The table shows how to create requests for accessing the %MW from the PROFIBUS DPV1 master:

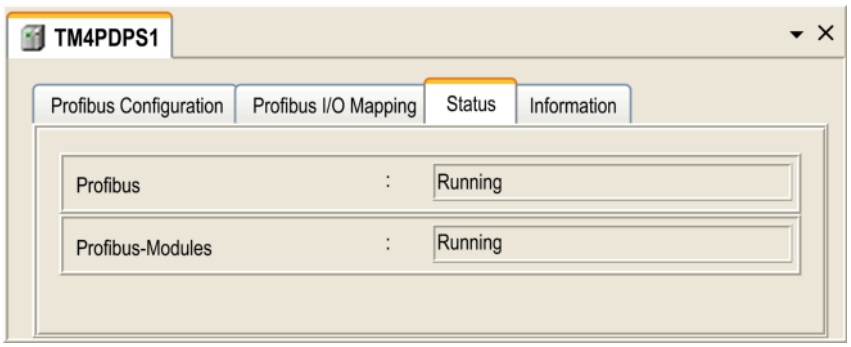
| Addressing | | DU0: DPV1 Function Number | DU1: Slot | DU2: Index | DU3: Length(1) (in Bytes) | DPV1 Data Frame |
|---|---|---|---|---|---|---|
| | | 1 Byte | 1 Byte | 1 Byte | 1 Byte | N Byte |
| Direct addressing | Write | 5F hex (write) | MSB of the %MW address | LSB of the %MW address | Length to read | Values to write |
| | Read | 5E hex (read) | MSB of the %MW address | LSB of the %MW address | Length to write | – |
| Indirect addressing | Send address (Step 1) | 5F hex (write) | 1 | E9 hex | 2 | %MW address |
| | Read (Step 2) | 5E hex (read) | 1 | EA hex | Length to read | – |
| | Write (Step 2) | 5F hex (write) | 1 | EA hex | Length to write | Values to write |
| **(1)** An even value must be entered for the **Length** field (the length in byte of one %MW is 2). | | | | | | |

# Diagnostic

## Diagnostic Information

### Displaying General Diagnostics Data

To display general diagnostic data, open the **Status** tab of the TM4PDPS1 configuration window:



## Monitoring the Status of the TM4PDPS1 Module

You can monitor the status of the TM4PDPS1 module with the `PROFIBUS_R` system data type described in the M241 Controller PLCSystem Library Guide or M251 Controller PLCSystem Library Guide depending on your controller.

## Fallback Management

When there is a PROFIBUS communication interruption (`i_CommState`=0), the outputs of the TM4PDPS1 are maintained to the last state transmitted by the PROFIBUS master.

The Fail Safe Mode as defined by the PROFIBUS DP standard is not supported by the TM4PDPS1 module.

## Messages on Detected Errors

Use `i_CommError` of the `PROFIBUS_R` system data type to visualize the detected error displayed.

The following message is displayed when no error is detected:

| Name | Value | Meaning |
|---|---|---|
| SUCCESS | 0 hex | No error detected. |

The following message is displayed when Runtime errors are detected:

| Name | Value | Meaning |
|---|---|---|
| WATCHDOG_ TIMEOUT | C000000C hex | The watchdog time has been exceeded. |

The following messages are displayed when Initialization errors are detected:

| Name | Value | Meaning |
|---|---|---|
| INIT_FAULT | C0000100 hex | The initialization was not successful. |
| DATABASE_ ACCESS_FAILED | C0000101 hex | Access to data memory was not successful. |

The following messages are displayed when Configuration errors are detected:

| Name | Value | Meaning |
|---|---|---|
| NOT_CONFIGURED | C0000119 hex | The TM4PDPS1 PCI module is not configured. |
| CONFIGURATION_ FAULT | C0000120 hex | A configuration error has been detected. |
| INCONSISTENT_ DATA_SET | C0000121 hex | Inconsistent set data have been detected. |
| DATA_SET_ MISMATCH | C0000122 hex | A mismatch of set data has been detected. |
| INSUFFICIENT_ LICENSE | C0000123 hex | A license error has been detected. |
| PARAMETER_ ERROR | C0000124 hex | A parameter error has been detected. |
| INVALID_ NETWORK_ ADDRESS | C0000125 hex | The network address is not correct. |
| SECURITY_ MEMORY | C0000126 hex | The security memory is not available. |

The following messages are displayed when Network errors are detected:

| Name | Value | Meaning |
|---|---|---|
| COMM_NETWORK_FAULT | C0000140 hex | A network communication error has been detected. |
| COMM_CONNECTION_CLOSED | C0000141 hex | The communication connection has been closed. |
| COMM_CONNECTION_TIMEOUT | C0000142 hex | A communication connection timeout has been detected. |
| COMM_DUPLICATE_NODE | C0000144 hex | A duplicate node has been detected. |
| COMM_CABLE_DISCONNECT | C0000145 hex | A disconnected cable has been detected. |
| PROFIBUS_CONNECTION_TIMEOUT | C009002E hex | A PROFIBUS connection timeout has been detected. |

# Glossary

## A

**ARP:**

(*address resolution protocol*) An IP network layer protocol for Ethernet that maps an IP address to a MAC (hardware) address.

## B

**BOOTP:**

(*bootstrap protocol*) A UDP network protocol that can be used by a network client to automatically obtain an IP address (and possibly other data) from a server. The client identifies itself to the server using the client MAC address. The server, which maintains a pre-configured table of client device MAC addresses and associated IP addresses, sends the client its pre-configured IP address. BOOTP was originally used as a method that enabled diskless hosts to be remotely booted over a network. The BOOTP process assigns an infinite lease of an IP address. The BOOTP service utilizes UDP ports 67 and 68.

## C

**configuration:**

The arrangement and interconnection of hardware components within a system and the hardware and software parameters that determine the operating characteristics of the system.

**control network:**

A network containing logic controllers, SCADA systems, PCs, HMI, switches, ...

Two kinds of topologies are supported:

- flat: all modules and devices in this network belong to same subnet.
- 2 levels: the network is split into an operation network and an inter-controller network.

These two networks can be physically independent, but are generally linked by a routing device.

## D

**device network:**

A network that contains devices connected to a specific communication port of a logic controller. This controller is seen as a master from the devices point of view.

**DHCP:**

(*dynamic host configuration protocol*) An advanced extension of BOOTP. DHCP is more advanced, but both DHCP and BOOTP are common. (DHCP can handle BOOTP client requests.)

**DNS:**

(*domain name system*) The naming system for computers and devices connected to a LAN or the Internet.

## E

**EDS:**

(*electronic data sheet*) A file for fieldbus device description that contains, for example, the properties of a device such as parameters and settings.

**EtherNet/IP:**

(*Ethernet industrial protocol*) An open communications protocol for manufacturing automation solutions in industrial systems. EtherNet/IP is in a family of networks that implement the common industrial protocol at its upper layers. The supporting organization (ODVA) specifies EtherNet/IP to accomplish global adaptability and media independence.

## F

**FTP:**

(*file transfer protocol*) A standard network protocol built on a client-server architecture to exchange and manipulate files over TCP/IP based networks regardless of their size.

## I

**ICMP:**

(*Internet control message protocol*) Reports errors detected and provides information related to datagram processing.

**IP:**

(*Internet protocol* Part of the TCP/IP protocol family that tracks the Internet addresses of devices, routes outgoing messages, and recognizes incoming messages.

## L

**LSB:**

(*least significant bit/byte*) The part of a number, address, or field that is written as the right-most single value in conventional hexadecimal or binary notation.

## M

**MAC address:**

(*media access control address*) A unique 48-bit number associated with a specific piece of hardware. The MAC address is programmed into each network card or device when it is manufactured.

**MIB:**

(*management information base*) An object database that is monitored by a network management system like SNMP. SNMP monitors devices are defined by their MIBs. Schneider Electric has obtained a private MIB, groupeschneider (3833).

**MSB:**

(*most significant bit/byte* The part of a number, address, or field that is written as the left-most single value in conventional hexadecimal or binary notation.

## N

**node:**

An addressable device on a communication network.

## P

**PCI:**

(*peripheral component interconnect*) An industry-standard bus for attaching peripherals.

**Profibus DP:**

(*Profibus decentralized peripheral*) An open bus system uses an electrical network based on a shielded 2-wire line or an optical network based on a fiber-optic cable. DP transmission allows for high-speed, cyclic exchange of data between the controller CPU and the distributed I/O devices.

**protocol:**

A convention or standard definition that controls or enables the connection, communication, and data transfer between 2 computing system and devices.

## R

**RPI:**

(*requested packet interval)* The time period between cyclic data exchanges requested by the scanner. EtherNet/IP devices publish data at the rate specified by the RPI assigned to them by the scanner, and they receive message requests from the scanner with a period equal to RPI.

## S

**SNMP:**

(*simple network management protocol*) A protocol that can control a network remotely by polling the devices for their status and viewing information related to data transmission. You can also use it to manage software and databases remotely. The protocol also permits active management tasks, such as modifying and applying a new configuration.

## T

**TCP:**

(*transmission control protocol*) A connection-based transport layer protocol that provides a simultaneous bi-directional transmission of data. TCP is part of the TCP/IP protocol suite.

## U

**UDP:**

(*user datagram protocol*) A connectionless mode protocol (defined by IETF RFC 768) in which messages are delivered in a datagram (data telegram) to a destination computer on an IP network. The UDP protocol is typically bundled with the Internet protocol. UDP/IP messages do not expect a response, and are therefore ideal for applications in which dropped packets do not require retransmission (such as streaming video and networks that demand real-time performance).

# Index

## A



## C



## D



## E



## F



## M



## P



## S

Schneider Electric
35 rue Joseph Monier
92500 Rueil Malmaison
France

www.se.com



EIO0000003149.04