Material Handling
Front End Processor M251
Project Template User Guide

10/2014

EIO0000001760.00

www.schneider-electric.com

Schneider Electric

The information provided in this documentation contains general descriptions and/or technical characteristics of the performance of the products contained herein. This documentation is not intended as a substitute for and is not to be used for determining suitability or reliability of these products for specific user applications. It is the duty of any such user or integrator to perform the appropriate and complete risk analysis, evaluation and testing of the products with respect to the relevant specific application or use thereof. Neither Schneider Electric nor any of its affiliates or subsidiaries shall be responsible or liable for misuse of the information contained herein. If you have any suggestions for improvements or amendments or have found errors in this publication, please notify us.

No part of this document may be reproduced in any form or by any means, electronic or mechanical, including photocopying, without express written permission of Schneider Electric.

All pertinent state, regional, and local safety regulations must be observed when installing and using this product. For reasons of safety and to help ensure compliance with documented system data, only the manufacturer should perform repairs to components.

When devices are used for applications with technical safety requirements, the relevant instructions must be followed.

Failure to use Schneider Electric software or approved software with our hardware products may result in injury, harm, or improper operating results.

Failure to observe this information can result in injury or equipment damage.

© 2014 Schneider Electric. All rights reserved.

Table of Contents

Safety Information

Important Information

NOTICE

Read these instructions carefully, and look at the equipment to become familiar with the device before trying to install, operate, 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.

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.

⚠ CAUTION

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 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.

About the Book

At a Glance

Document Scope

This document describes a conveying application based on Modicon M251 Logic Controller. This document is an example of an application used to control a conveyor.

This manual is intended for use by individuals knowledgeable and experienced in material handling technologies.

The following basic knowledge is required:

- basic information on functionality, structure, and configuration of the controllers, drives, and HMI displays
- programming languages: IL, ST, FBD, SFC, LD, CFC

Validity Note

This document has been created with the release of SoMachine V4.1 Material Handling add-on.

Related Documents

Title of Documentation	Reference Number
SoMachine Programming Guide	EIO0000000069
Machine Energy Dashboard Library Guide	EIO0000001163
SoMachine Conveying Application Functions, Conveying Library Guide	EIO0000000201

You can download these technical publications and other technical information from our website at www.schneider-electric.com.

Product Related Information

⚠ WARNING

LOSS OF CONTROL

- The designer of any control scheme must consider the potential failure modes of control paths and, for certain critical control functions, provide a means to achieve a safe state during and after a path failure. Examples of critical control functions are emergency stop and overtravel stop, power outage and restart.
- Separate or redundant control paths must be provided for critical control functions.
- System control paths may include communication links. Consideration must be given to the implications of unanticipated transmission delays or failures of the link.
- Observe all accident prevention regulations and local safety guidelines.[1]
- Each implementation of this equipment must be individually and thoroughly tested for proper operation before being placed 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.

User Comments

We welcome your comments about this document. You can reach us by e-mail at techcomm@schneider-electric.com.

Chapter 1
Front End Processor Application

Overview

The Modicon M251 Logic Controller with Modbus TCP communication capabilities provides connectivity to the SCADA (Supervisory Control And Data Acquisition), MES (Manufacturing Execution Systems) and ERP (Enterprise Resource Planning) systems through Ethernet.

The need to interconnect more and more machines into one management system leads to the structure of intelligent zones. This decentralized approach requires at least one logic controller in each zone.

The Modicon M251 Logic Controller collects the information from all the zones. It acts as the central for all the zones. The advantage of such an architecture is the ability to monitor the complete system without having to look at each of the zone logic controllers and to have one logic controller handling the complete communication upstream to the WMS (Warehouse Management System), other zones or databases.

What Is in This Chapter?

This chapter contains the following topics:

Introduction

This document is intended to provide a quick introduction and programming example to the described application. It is not intended to replace any specific product documentation, nor any of your own design documentation. On the contrary, it offers additional information to the product documentation for installing, configuring and implementing the application.

The architecture described in this document is not a specific product in the normal commercial sense. It describes an example of how Schneider Electric and third-party components may be integrated to fulfil an industrial application. A detailed functional description or the specification for a specific user application is not part of this document.

Your specific application requirements can be different and will require additional and/or different components, configuration and/or programming logic than that is found in this document. In that case, you will have to adapt the information provided in this document to your particular needs. In all and any cases, pay particular attention in conforming to any safety information, different electrical requirements and normative standards that would apply to your adaptation.

⚠ WARNING

REGULATORY INCOMPATIBILITY

Be sure that all equipment applied and systems designed comply with all applicable local, regional and national regulations and standards.

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

NOTE: There are some major logical and physical components in the application example described herein. They cannot be substituted without completely invalidating the architecture, descriptions, instructions, wiring diagrams and compatibility between the various software and hardware components specified in this document. You must be aware of the consequences of component substitutions or modifications in the architecture described in this document as they can impair the compatibility and interoperability of software and hardware.

A residual risk, as defined by EN/ISO 12100-1, Article 5, will remain if:

- it is necessary to modify the recommended logic and if the added or modified components are not properly integrated in the control circuit.
- you do not follow the required standards applicable to the operation of the machine, or if the adjustments to and the maintenance of the machine are not properly made (it is essential to strictly follow the prescribed machine maintenance schedule).
- the devices connected to any safety outputs do not have mechanically-linked contacts.

⚠ CAUTION

EQUIPMENT INCOMPATIBILITY

Read and thoroughly understand all device and software documentation before attempting any component substitutions or other changes related to the application examples provided in the document.

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

System Requirements

Using the Library

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- Verify the SoMachine libraries contained in your program are the correct version after updating SoMachine software.
- Verify that the library versions updated are consistent with your application specifications.

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

For more detailed information, see Schneider Electric Libraries *(see SoMachine, Functions and Libraries User Guide)*.

For IEC 61131-3 compatibility, the ability to add the EN/ENO input/output automatically to Function Blocks of certain programming languages is available to the programmer. However, for certain applications that require the complex interaction of multiple function blocks, the use of the IEC 61131-3 input to disable a function block in a series of interrelated functions affecting a process may lead to unintended operation of the system as a whole. For the functions contained in the Library that is the topic of the current document, this is especially true.

The EN/ENO inputs and outputs as defined by IEC 61131-3 are maladapted to, and therefore inappropriate for, the targeted application of these functions. Suddenly disabling one function by a falling edge on the EN input would require all outputs of the function block to immediately fall to their default states, and such an unanticipated action would cause in abrupt change to the entire process. The implication is that such an event would have deleterious results that may invoke undesirable consequences. Therefore, the EN/ENO inputs/outputs as defined by IEC 61131-3 are incompatible with the functions contained within this library.

⚠ WARNING

UNINTENDED MACHINE OPERATION

Do not use the EN/ENO functionality defined by IEC 61131-3 to control the behavior of the Application Function blocks.

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

NOTE: Verify that the EN/ENO option is disabled in the compiler options menu of SoMachine.

Conveying System

Overview

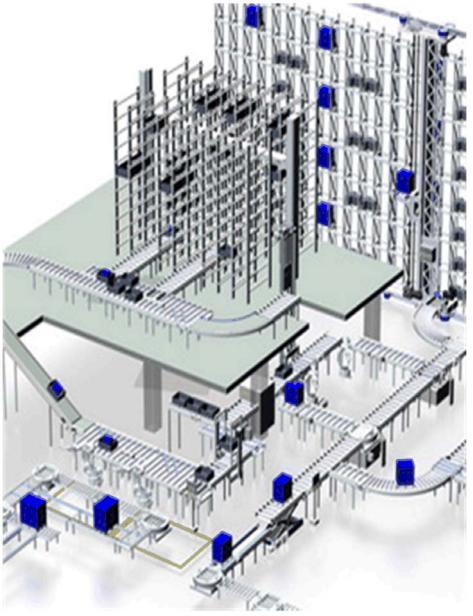
A conveying system is often divided into several zones, each managed by a logic controller.

- zones that include certain application functions and mechanical equipments
- zones that can include an emergency system (safety relays with emergency stop push buttons, trip wire switches)

This example realizes a communication front end of two conveying zones. The front end processor application is designed as the first stage in setting up a conveying management system.

Conveying Example

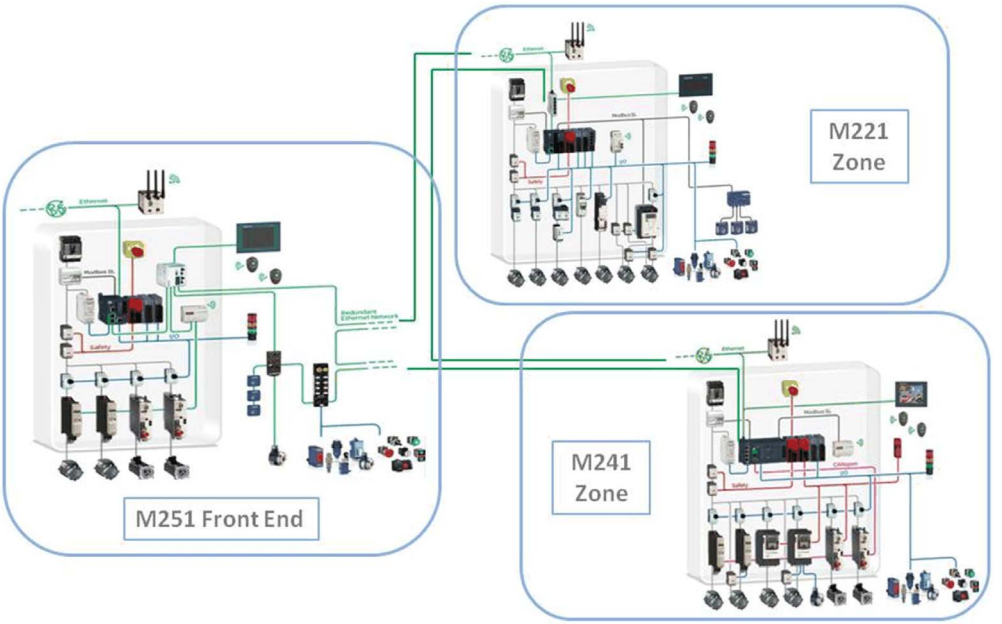
This example realises a communication front end of two different conveying zones:



This front end processor is part of this conveyor application which is controlled by 2 different logic controllers:

- Modicon M241 Logic Controller for zone 1.
- Modicon M221 Logic Controller for the zero pressure accumulation (ZPA) application in zone 2.

The following figure describes the system control architecture:



Application Functions

Overview

The Modicon M251 Logic Controller provides solutions for distributed architectures and modular machines.

You can integrate them in wall-mounted and floor standing control system enclosures.

The embedded Ethernet ports support the File Transfer Protocol (FTP) and a web server. It allows the integration of control system architectures and remote control of machines using applications for smartphones, tablets, and PCs.

This application collects the data from the different zones linked to the conveying system.

In this case, there are two zones: the conveying zone and the zero pressure accumulation (ZPA) zone.

Data is provided to the Magelis HMIGTO4310 linked to the Modicon M251 Logic Controller.

- In monitor mode, it provides the data from both zones.
- In command mode, it is possible to send different commands such as START/STOP or a change of parameter settings such as speed.

NOTE: The Web and FTP servers are powerful tools for reading and writing data, and controlling the state of the logic controller, with full direct (Web Server) or indirect (FTP Server) access to all data in your application. However, if there are security concerns over these functions, you must at a minimum assign a secure password to restrict these services to the Web Server or, in the case of direct access, disable the Web server to prevent unauthorized access to the application.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- Define a secure password for the Web Server, and do not allow unauthorized or otherwise unqualified personnel to use this feature.
- Ensure that there is a local, competent, and qualified observer present when operating on the controller from a remote location.
- You must have a complete understanding of the application and the machine/process it is controlling before attempting to adjust data, stopping an application that is operating, or starting the controller remotely.
- Take the precautions necessary to assure that you are operating on the intended controller by having clear, identifying documentation within the controller application and its remote connection.

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

NOTE: Access to the Web Server site requires a login on first prompt. For more information on password management, refer to the SoMachine Programming Guide *(see SoMachine, Programming Guide)*.

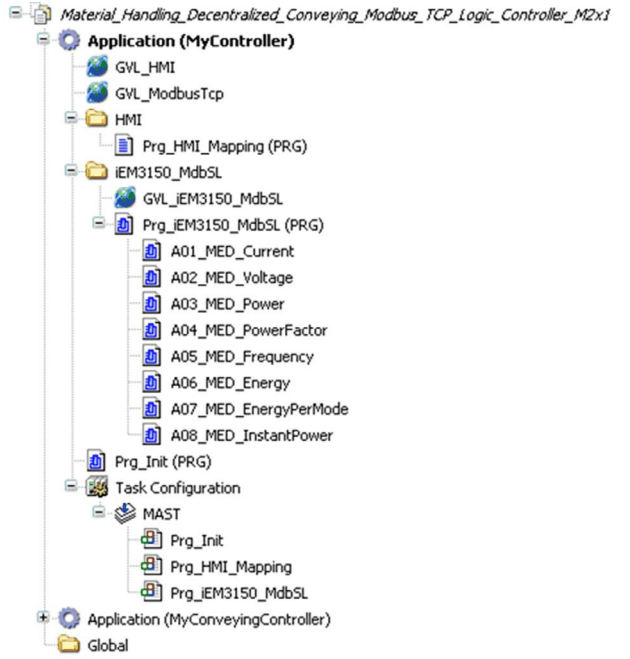
Application Software

Overview

Use this application for the following tasks:

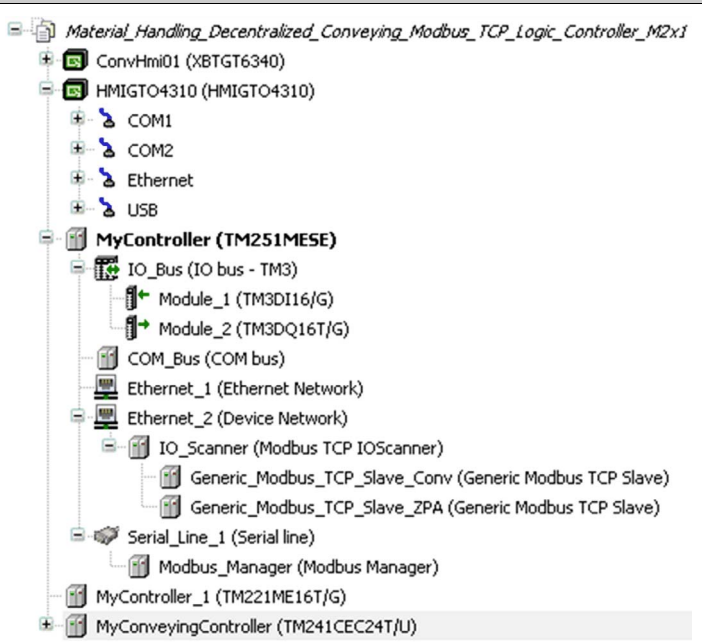
- program the Modicon M251 Logic Controller
- configure the Modbus TCP IOScanner
- program the Magelis HMIGTO4310 display

SoMachine software is used to program the Modicon M251 Logic Controller.

SoMachine	
Application tree	

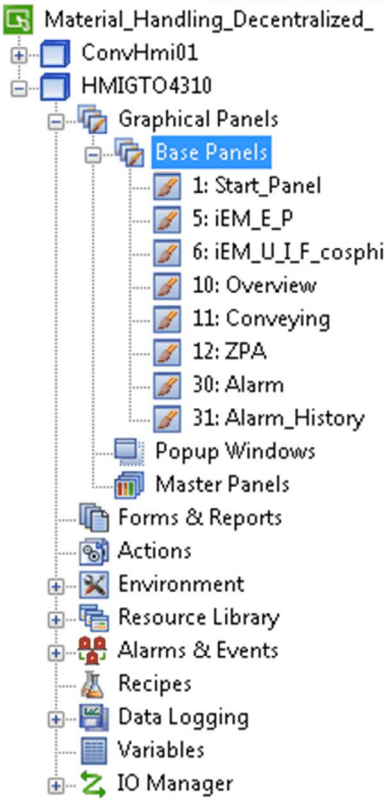
The SoMachine application tree for the Modicon M251 Logic Controller consists of two main applications parts:

- Prg_HMI_Mapping
  This POU executes two functions:
  - Reading data: The POU Prg_HMI_Mapping maps the data read by the Modbus TCP IOScanner from the Modicon M241 Logic Controller and Modicon M221 Logic Controller to the data read by the HMI application.
  - Writing data: The POU Prg_HMI_Mapping maps the data written by the HMI application to the data written by the HMI application to the Modicon M241 Logic Controller and Modicon M221 Logic Controller.
- Prg_iEM3150MdbSL
  This POU reads energy values provided by the power meter iEM3150 and links them to the relevant application function blocks (AFBs) provided with Machine Energy Dashboard library.

SoMachine	
Device Tree	

The embedded IO_Scanner (Modbus TCP IOScanner) is linked to the port Ethernet_2 (Device Network). Therefore, this Ethernet port is used to READ/WRITE data to and from the Modicon M241 Logic Controller and Modicon M221 Logic Controller.

Vijeo-Designer software is used to program the Magelis HMIGTO4310.

Vijeo-Designer	
HMI application	

For more information, refer to Magelis HMIGTO4310 Application *(see page 42)*.

Chapter 2
Application Implementation

Overview

The application can be structured as follows:

- Logic controller application
  - Data exchange between the zone logic controllers
  - Data exchange with the HMI
  - Handling of energy values provided by power meter
- HMI application
  - Monitoring the status of the different zones (read data)
  - Command handling for the different zones (write data)
  - Presenting energy values

What Is in This Chapter?

This chapter contains the following topics:

Data Exchange between the Logic Controllers

Overview

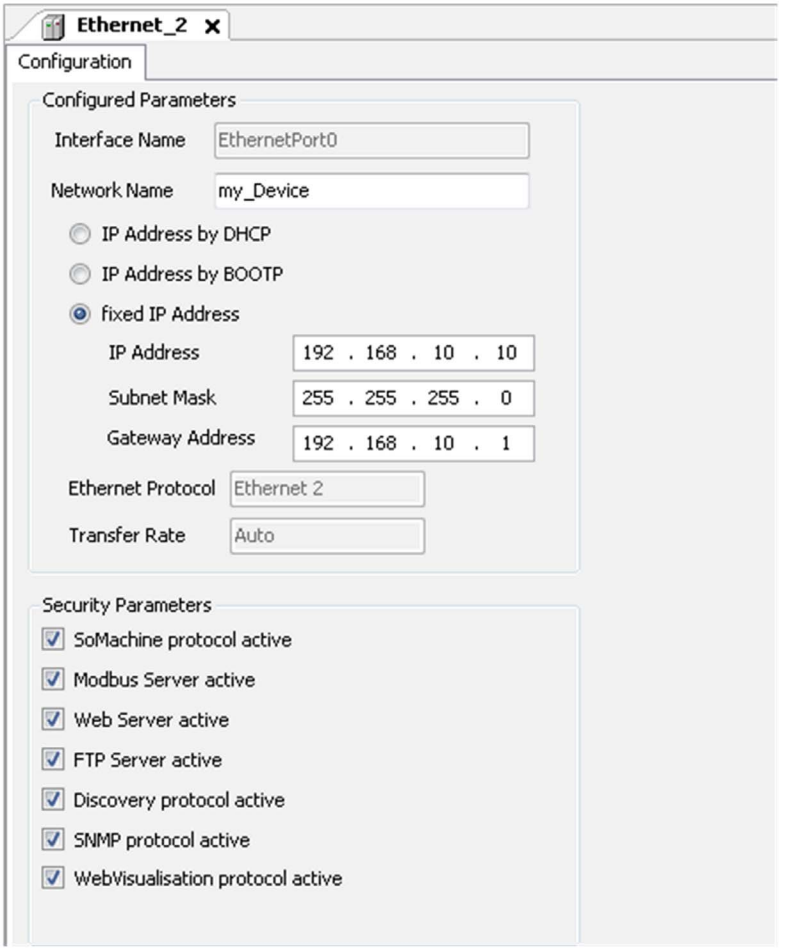
The Modicon M251 Logic Controller application acts as front end processor in this conveying architecture. It collects the relevant data from the different zones and displays them on the HMI used for controlling and monitoring the application.

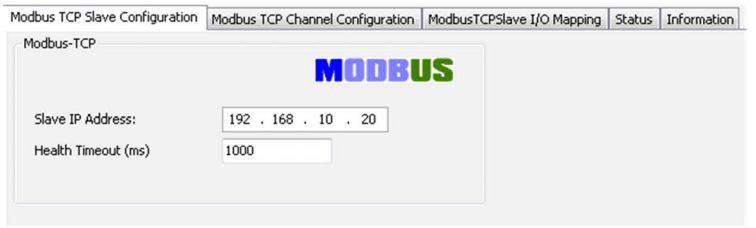
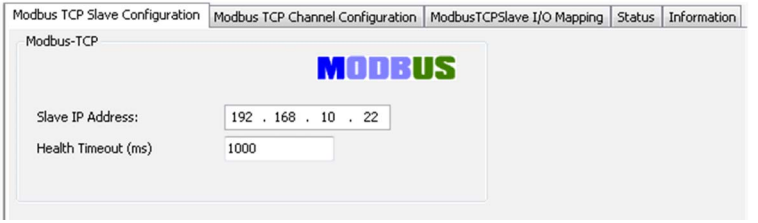
For the communication, the second Ethernet port of the logic controller is used and the Modbus TCP IOScanner is running.

The Modbus TCP IOScanner writes and reads data to and from the Modicon M241 Logic Controller and the Modicon M221 Logic Controller.

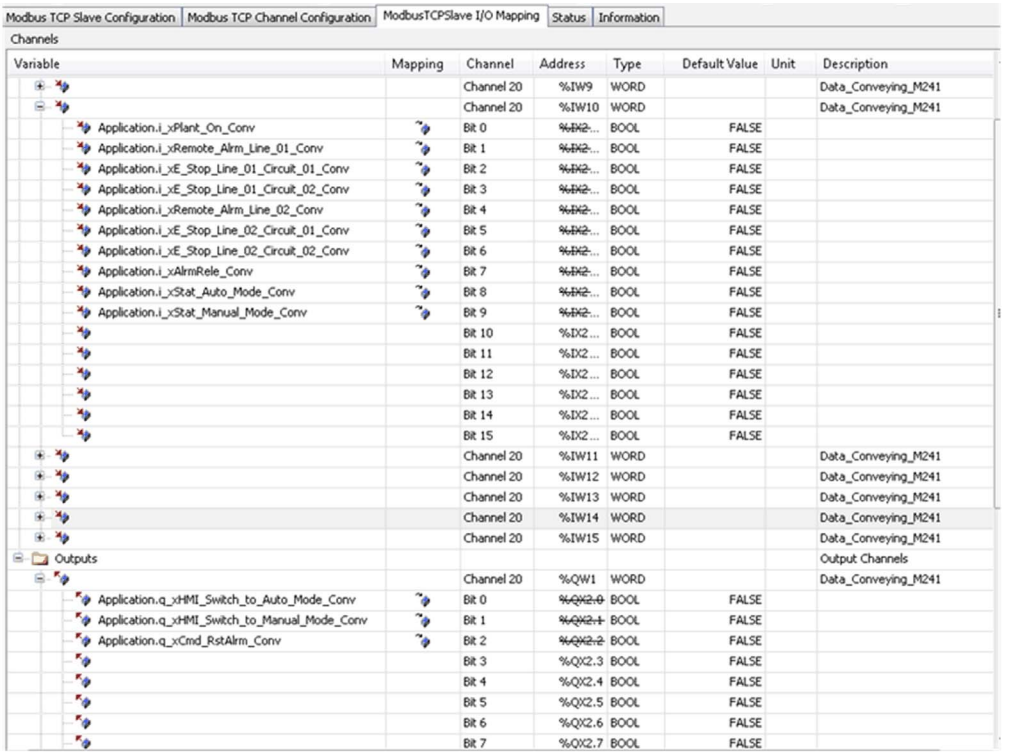
NOTE: The variables are mapped in a POU. For data exchange, only BOOL and WORD data types are used.

IOScanner Configuration

SoMachine	
Ethernet_2 port configuration	

SoMachine	
Modbus TCP slave configuration for the Modicon M241 Logic Controller.	
Modbus TCP slave configuration for the Modicon M221 Logic Controller.	

Modbus TCP Slave I/O Mapping for the M241 Logic Controller



Modbus TCP Slave Configuration | Modbus TCP Channel Configuration | ModbusTCPSlave I/O Mapping | Status | Information

Channels

Variable	Mapping	Channel	Address	Type	Default Value	Unit	Description
		Channel 20	%IW9	WORD			Data_Conveying_M241
		Channel 20	%IW10	WORD			Data_Conveying_M241
Application.i_xPlant_On_Conv		Bit 0	%IX2...	BOOL	FALSE		
Application.i_xRemote_Alrm_Line_01_Conv		Bit 1	%IX2...	BOOL	FALSE		
Application.i_xE_Stop_Line_01_Circuit_01_Conv		Bit 2	%IX2...	BOOL	FALSE		
Application.i_xE_Stop_Line_01_Circuit_02_Conv		Bit 3	%IX2...	BOOL	FALSE		
Application.i_xRemote_Alrm_Line_02_Conv		Bit 4	%IX2...	BOOL	FALSE		
Application.i_xE_Stop_Line_02_Circuit_01_Conv		Bit 5	%IX2...	BOOL	FALSE		
Application.i_xE_Stop_Line_02_Circuit_02_Conv		Bit 6	%IX2...	BOOL	FALSE		
Application.i_xAlrmRele_Conv		Bit 7	%IX2...	BOOL	FALSE		
Application.i_xStat_Auto_Mode_Conv		Bit 8	%IX2...	BOOL	FALSE		
Application.i_xStat_Manual_Mode_Conv		Bit 9	%IX2...	BOOL	FALSE		
		Bit 10	%IX2...	BOOL	FALSE		
		Bit 11	%IX2...	BOOL	FALSE		
		Bit 12	%IX2...	BOOL	FALSE		
		Bit 13	%IX2...	BOOL	FALSE		
		Bit 14	%IX2...	BOOL	FALSE		
		Bit 15	%IX2...	BOOL	FALSE		
		Channel 20	%IW11	WORD			Data_Conveying_M241
		Channel 20	%IW12	WORD			Data_Conveying_M241
		Channel 20	%IW13	WORD			Data_Conveying_M241
		Channel 20	%IW14	WORD			Data_Conveying_M241
		Channel 20	%IW15	WORD			Data_Conveying_M241
Outputs							Output Channels
		Channel 20	%QW1	WORD			Data_Conveying_M241
Application.q_xHMI_Switch_to_Auto_Mode_Conv		Bit 0	%QX2.0	BOOL	FALSE		
Application.q_xHMI_Switch_to_Manual_Mode_Conv		Bit 1	%QX2.1	BOOL	FALSE		
Application.q_xCmd_RstAlrm_Conv		Bit 2	%QX2.2	BOOL	FALSE		
		Bit 3	%QX2.3	BOOL	FALSE		
		Bit 4	%QX2.4	BOOL	FALSE		
		Bit 5	%QX2.5	BOOL	FALSE		
		Bit 6	%QX2.6	BOOL	FALSE		
		Bit 7	%QX2.7	BOOL	FALSE		

NOTE: All data exchanged with the Modicon M241 Logic Controller are mapped to the %IWxx and %QWxx register listed in ModbusTCPSlave I/O Mapping.

READ data from M241 Logic Controller: Conveyor Application

Variable Name	Data Type	Description
`i_xE_Stop_Line_01_Circuit_01_Conv`	BOOL	Emergency stop state line 1 circuit 1 TRUE: no emergency stop FALSE: emergency stop pending
`i_xE_Stop_Line_01_Circuit_02_Conv`	BOOL	Emergency stop state line 1 circuit 2 TRUE: no emergency stop FALSE: emergency stop pending
`i_xE_Stop_Line_02_Circuit_01_Conv`	BOOL	Emergency stop state line 2 circuit 1 TRUE: no emergency stop FALSE: emergency stop pending

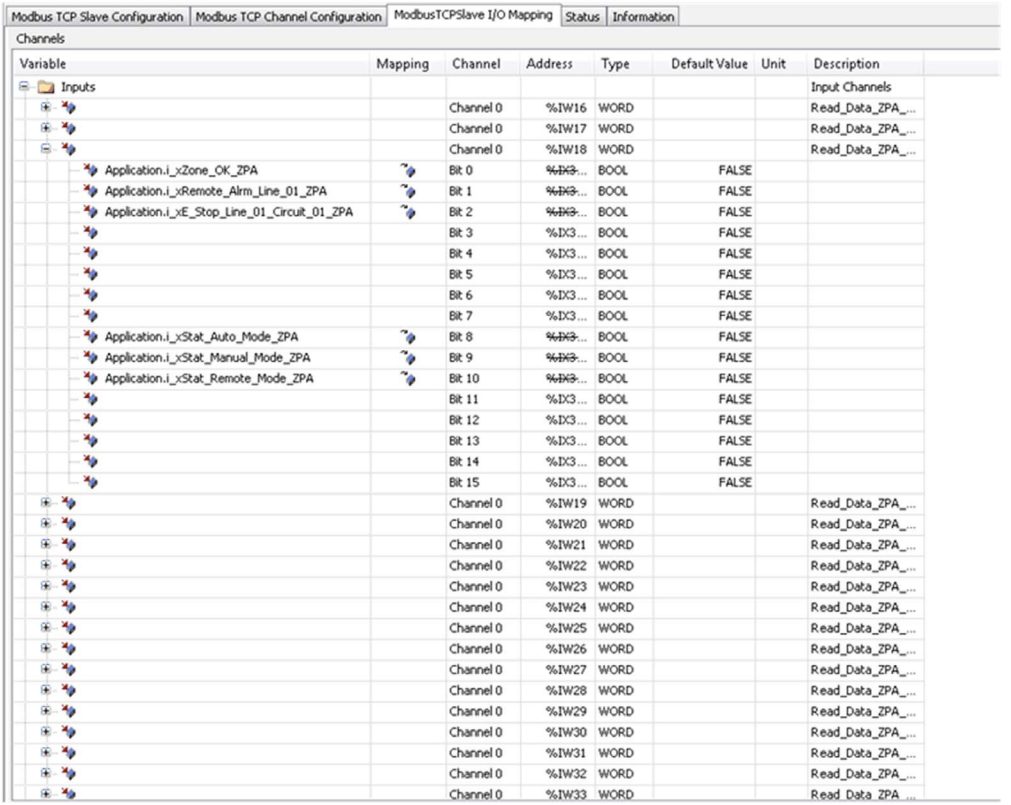
Variable Name	Data Type	Description
i_xE_Stop_Line_02_Circuit_02_Conv	BOOL	Emergency stop state line 2 circuit 2 TRUE: no emergency stop FALSE: emergency stop pending
i_xPlant_On_Conv	BOOL	Machine state TRUE: running mode FALSE: stop mode
i_xRemote_Alrm_Line_01_Conv	BOOL	Alarm line 1 TRUE: alarm pending FALSE: no alarm pending
i_xRemote_Alrm_Line_02_Conv	BOOL	Alarm line 2 TRUE: alarm pending FALSE: no alarm pending
i_xAlrmRele_Conv	BOOL	Conveyor alarm is released
i_xStat_Auto_Mode_Conv	BOOL	Machine state for automatic mode TRUE: automatic mode FALSE: no automatic mode
i_xStat_Manual_Mode_Conv	BOOL	Machine state for manual mode TRUE: manual mode FALSE: no manual mode

WRITE Data to M241 Logic Controller: Conveyor Application

Variable Name	Data Type	Description
q_xHMI_Switch_to_Auto_Mode_Conv	BOOL	HMI command to switch the conveyor to automatic mode TRUE: switch FALSE: no action
q_xHMI_Switch_to_Manual_Mode_Conv	BOOL	Command to switch the conveyor to manual mode TRUE: switch FALSE: no action
q_xCmd_RstAlrm_Conv	BOOL	Reset alarm state TRUE: reset FALSE: no action

Modbus TCP Slave I/O Mapping for the M221 Logic Controller

Inputs:

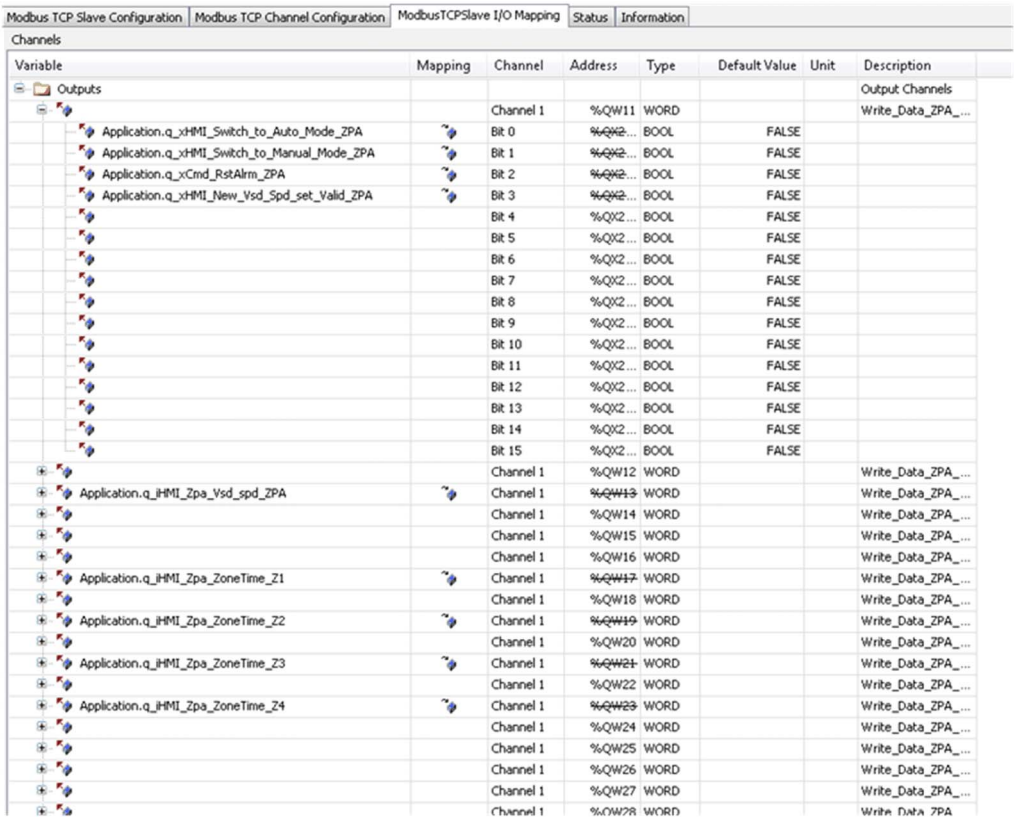


Modbus TCP Slave Configuration | Modbus TCP Channel Configuration | ModbusTCPSlave I/O Mapping | Status | Information

Channels

Variable	Mapping	Channel	Address	Type	Default Value	Unit	Description
Inputs							Input Channels
		Channel 0	%IW16	WORD			Read_Data_ZPA_...
		Channel 0	%IW17	WORD			Read_Data_ZPA_...
		Channel 0	%IW18	WORD			Read_Data_ZPA_...
Application.i_xZone_OK_ZPA		Bit 0	%IX3...	BOOL	FALSE		
Application.i_xRemote_Alrm_Line_01_ZPA		Bit 1	%IX3...	BOOL	FALSE		
Application.i_xE_Stop_Line_01_Circuit_01_ZPA		Bit 2	%IX3...	BOOL	FALSE		
		Bit 3	%IX3...	BOOL	FALSE		
		Bit 4	%IX3...	BOOL	FALSE		
		Bit 5	%IX3...	BOOL	FALSE		
		Bit 6	%IX3...	BOOL	FALSE		
		Bit 7	%IX3...	BOOL	FALSE		
Application.i_xStat_Auto_Mode_ZPA		Bit 8	%IX3...	BOOL	FALSE		
Application.i_xStat_Manual_Mode_ZPA		Bit 9	%IX3...	BOOL	FALSE		
Application.i_xStat_Remote_Mode_ZPA		Bit 10	%IX3...	BOOL	FALSE		
		Bit 11	%IX3...	BOOL	FALSE		
		Bit 12	%IX3...	BOOL	FALSE		
		Bit 13	%IX3...	BOOL	FALSE		
		Bit 14	%IX3...	BOOL	FALSE		
		Bit 15	%IX3...	BOOL	FALSE		
		Channel 0	%IW19	WORD			Read_Data_ZPA_...
		Channel 0	%IW20	WORD			Read_Data_ZPA_...
		Channel 0	%IW21	WORD			Read_Data_ZPA_...
		Channel 0	%IW22	WORD			Read_Data_ZPA_...
		Channel 0	%IW23	WORD			Read_Data_ZPA_...
		Channel 0	%IW24	WORD			Read_Data_ZPA_...
		Channel 0	%IW25	WORD			Read_Data_ZPA_...
		Channel 0	%IW26	WORD			Read_Data_ZPA_...
		Channel 0	%IW27	WORD			Read_Data_ZPA_...
		Channel 0	%IW28	WORD			Read_Data_ZPA_...
		Channel 0	%IW29	WORD			Read_Data_ZPA_...
		Channel 0	%IW30	WORD			Read_Data_ZPA_...
		Channel 0	%IW31	WORD			Read_Data_ZPA_...
		Channel 0	%IW32	WORD			Read_Data_ZPA_...
		Channel 0	%IW33	WORD			Read_Data_ZPA_...

Outputs:



Modbus TCP Slave Configuration | Modbus TCP Channel Configuration | ModbusTCPSlave I/O Mapping | Status | Information

Channels

Variable	Mapping	Channel	Address	Type	Default Value	Unit	Description
Outputs							Output Channels
		Channel 1	%QW11	WORD			Write_Data_ZPA_...
Application.q_xHMI_Switch_to_Auto_Mode_ZPA		Bit 0	%QX2...	BOOL	FALSE		
Application.q_xHMI_Switch_to_Manual_Mode_ZPA		Bit 1	%QX2...	BOOL	FALSE		
Application.q_xCmd_RstAlrm_ZPA		Bit 2	%QX2...	BOOL	FALSE		
Application.q_xHMI_New_Vsd_Spd_set_Valid_ZPA		Bit 3	%QX2...	BOOL	FALSE		
		Bit 4	%QX2...	BOOL	FALSE		
		Bit 5	%QX2...	BOOL	FALSE		
		Bit 6	%QX2...	BOOL	FALSE		
		Bit 7	%QX2...	BOOL	FALSE		
		Bit 8	%QX2...	BOOL	FALSE		
		Bit 9	%QX2...	BOOL	FALSE		
		Bit 10	%QX2...	BOOL	FALSE		
		Bit 11	%QX2...	BOOL	FALSE		
		Bit 12	%QX2...	BOOL	FALSE		
		Bit 13	%QX2...	BOOL	FALSE		
		Bit 14	%QX2...	BOOL	FALSE		
		Bit 15	%QX2...	BOOL	FALSE		
		Channel 1	%QW12	WORD			Write_Data_ZPA_...
Application.q_iHMI_Zpa_Vsd_spd_ZPA		Channel 1	%QW13	WORD			Write_Data_ZPA_...
		Channel 1	%QW14	WORD			Write_Data_ZPA_...
		Channel 1	%QW15	WORD			Write_Data_ZPA_...
		Channel 1	%QW16	WORD			Write_Data_ZPA_...
Application.q_iHMI_Zpa_ZoneTime_Z1		Channel 1	%QW17	WORD			Write_Data_ZPA_...
		Channel 1	%QW18	WORD			Write_Data_ZPA_...
Application.q_iHMI_Zpa_ZoneTime_Z2		Channel 1	%QW19	WORD			Write_Data_ZPA_...
		Channel 1	%QW20	WORD			Write_Data_ZPA_...
Application.q_iHMI_Zpa_ZoneTime_Z3		Channel 1	%QW21	WORD			Write_Data_ZPA_...
		Channel 1	%QW22	WORD			Write_Data_ZPA_...
Application.q_iHMI_Zpa_ZoneTime_Z4		Channel 1	%QW23	WORD			Write_Data_ZPA_...
		Channel 1	%QW24	WORD			Write_Data_ZPA_...
		Channel 1	%QW25	WORD			Write_Data_ZPA_...
		Channel 1	%QW26	WORD			Write_Data_ZPA_...
		Channel 1	%QW27	WORD			Write_Data_ZPA_...
		Channel 1	%QW28	WORD			Write_Data_ZPA_...

NOTE: All data exchanged with the Modicon M221 Logic Controller are mapped to the %IWxx and %QWxx register listed in ModbusTCPSlave I/O Mapping.

READ Data from M221 Logic Controller: ZPA Application

Variable Name	Data Type	Description
`i_xE_Stop_Line_01_Circuit_01_ZPA`	BOOL	Emergency stop state of all 4 zones TRUE: no emergency stop FALSE: emergency stop pending
`i_xZone_OK_ZPA`	BOOL	All 4 zones of the ZPA state TRUE: 4 zones are running FALSE: minimum 1 zone is stopped

Variable Name	Data Type	Description
i_xStat_Auto_Mode_ZPA	BOOL	Machine state for automatic mode TRUE: automatic mode FALSE: no automatic mode
i_xStat_Manual_Mode_ZPA	BOOL	Machine state for manual mode TRUE: manual mode FALSE: no manual mode
i_xStat_Remote_Mode_ZPA	BOOL	Command level priority TRUE: command level remote HMI FALSE: command level local HMI or I/O

WRITE Data to M221 Logic Controller: ZPA Application

Variable Name	Data Type	Description
q_xHMI_Switch_to_Auto_Mode_ZPA	BOOL	HMI command to switch the conveyor to automatic mode TRUE: switch FALSE: no action
q_xHMI_Switch_to_Manual_Mode_ZPA	BOOL	Command to switch the conveyor to manual mode TRUE: switch FALSE: no action
q_xHMI_New_Vsd_Spd_set_Valid_ZPA	BOOL	Set speed value active for the variable speed drives
q_xCmd_RstAlrm_ZPA	BOOL	Reset alarm state TRUE: reset FALSE: no action
q_iHMI_Zpa_Vsd_spd_ZPA	WORD	Speed value to be set for all used variable speed drives (no scaling executed in ZPA application)
q_iHMI_Zpa_ZoneTime_Z1	WORD	TIMER set value for zone 1 (s)
q_iHMI_Zpa_ZoneTime_Z2	WORD	TIMER set value for zone 2 (s)
q_iHMI_Zpa_ZoneTime_Z3	WORD	TIMER set value for zone 3 (s)
q_iHMI_Zpa_ZoneTime_Z4	WORD	TIMER set value for zone 4 (s)
q_xHMI_Set_ZPA_ZoneTime_Z1	BOOL	Set timer value active for zone 1
q_xHMI_Set_ZPA_ZoneTime_Z2	BOOL	Set timer value active for zone 2
q_xHMI_Set_ZPA_ZoneTime_Z3	BOOL	Set timer value active for zone 3
q_xHMI_Set_ZPA_ZoneTime_Z4	BOOL	Set timer value active for zone 4

Data Exchange with the Magelis HMIGT04310

READ data from M251 Logic Controller: Conveyor Application

The Magelis HMIGTO4310 device communicates with the Modicon M251 Logic Controller as front end processor.The Modicon M251 Logic Controller communicates with the Modicon M241 Logic Controller. The data read by the Modbus TCP IOScanner (monitoring) from the Modicon M241 Logic Controller are mapped to the following variables. The HMI application reads these data using SoMachine protocol.

NOTE: The variables are mapped in a POU. For data exchange, only BOOL and WORD data type are used.

Data exchange with HMI:

SoMachine	
Data exchanged between the HMI and the Conveyor application running on Modicon M241 Logic Controller.	

```
//******* Conveying-controller M241 ************************************

// The READ Data from M241 Conveying controller

g_xHMI_Plant_On_Conv                := i_xPlant_On_Conv;

g_xHMI_Remote_Alrm_Line_01_Conv     := i_xRemote_Alrm_Line_01_Conv;
g_xHMI_EStopLine01Circuit01_Conv    := i_xE_Stop_Line_01_Circuit_01_Conv;
g_xHMI_EStopLine01Circuit02_Conv    := i_xE_Stop_Line_01_Circuit_02_Conv;

g_xALRM_Remote_Alrm_Line_01_Con     := i_xRemote_Alrm_Line_01_Conv;
g_xALRM_EStopLine01Circuit01_Con    := NOT i_xE_Stop_Line_01_Circuit_01_Conv;
g_xALRM_EStopLine01Circuit02_Con    := NOT i_xE_Stop_Line_01_Circuit_02_Conv;

g_xHMI_Remote_Alrm_Line_02_Conv     := i_xRemote_Alrm_Line_02_Conv;
g_xHMI_EStopLine02Circuit01_Conv    := i_xE_Stop_Line_02_Circuit_01_Conv;
g_xHMI_EStopLine02Circuit02_Conv    := i_xE_Stop_Line_02_Circuit_02_Conv;

g_xALRM_Remote_Alrm_Line_02_Con     := i_xRemote_Alrm_Line_02_Conv;
g_xALRM_EStopLine02Circuit01_Con    := NOT i_xE_Stop_Line_02_Circuit_01_Conv;
g_xALRM_EStopLine02Circuit02_Con    := NOT i_xE_Stop_Line_02_Circuit_02_Conv;

g_xHMI_AlrmRele_Conv                := i_xAlrmRele_Conv;

g_xHMI_Stat_Auto_Mode_Conv          := i_xStat_Auto_Mode_Conv;
g_xHMI_Stat_Manual_Mode_Conv        := i_xStat_Manual_Mode_Conv;

// The WRITE Data to M241 Conveying controller
q_xHMI_Switch_to_Auto_Mode_Conv     := g_xHMI_Switch_to_Auto_Mode_Conv;
q_xHMI_Switch_to_Manual_Mode_Conv   := g_xHMI_Switch_to_Manual_Mode_Con;

q_xCmd_RstAlrm_Conv                 := g_xHMI_RstAlrm_Conv;
```

SoMachine	
Data exchanged between the HMI and the ZPA application running on Modicon M221 Logic Controller.	

```
//******* ZPA-controller M221 ***********************************

// The READ Data from M221 ZPA controller
g_xHMI_Zone_OK_ZPA                  := i_xZone_OK_ZPA;

g_xHMI_Remote_Alrm_Line_01_ZPA      := NOT i_xZone_OK_ZPA;
g_xALRM_Remote_Alrm_Line_01_ZPA     := NOT i_xZone_OK_ZPA;

g_xHMI_EStopLine01Circuit01_ZPA     := NOT i_xE_Stop_Line_01_Circuit_01_ZPA;

g_xALRM_EStopLine01Circuit01_ZPA    := i_xE_Stop_Line_01_Circuit_01_ZPA;

g_xHMI_Stat_Auto_Mode_ZPA           := i_xStat_Auto_Mode_ZPA;
g_xHMI_Stat_Manual_Mode_ZPA         := i_xStat_Manual_Mode_ZPA;
g_xHMI_Stat_Remote_Mode_ZPA         := i_xStat_Remote_Mode_ZPA;

g_iHMI_Zpa_Vsd_spd_act_ZPA          := i_iHMI_Zpa_Vsd_spd_ZPA;

// The WRITE Data to M221 ZPA controller
q_xHMI_Switch_to_Auto_Mode_ZPA      := g_xHMI_Switch_to_Auto_Mode_ZPA;
q_xHMI_Switch_to_Manual_Mode_ZPA    := g_xHMI_Switch_to_Manual_Mode_ZPA;
q_xCmd_RstAlrm_ZPA                  := g_xHMI_RstAlrm_ZPA;
q_xHMI_New_Vsd_Spd_set_Valid_ZPA    := g_xHMI_New_Vsd_Spd_set_Valid_ZPA;

q_iHMI_Zpa_Vsd_spd_ZPA              :=g_iHMI_Zpa_Vsd_spd_set_ZPA;

q_iHMI_Zpa_ZoneTime_Z1              := g_iHMI_Zpa_ZoneTime_Z1;
q_iHMI_Zpa_ZoneTime_Z2              := g_iHMI_Zpa_ZoneTime_Z2;
q_iHMI_Zpa_ZoneTime_Z3              := g_iHMI_Zpa_ZoneTime_Z3;
q_iHMI_Zpa_ZoneTime_Z4              := g_iHMI_Zpa_ZoneTime_Z4;

q_xHMI_Set_ZPA_ZoneTime_Z1          := g_xHMI_Set_ZPA_ZoneTime_Z1;
q_xHMI_Set_ZPA_ZoneTime_Z2          := g_xHMI_Set_ZPA_ZoneTime_Z2;
q_xHMI_Set_ZPA_ZoneTime_Z3          := g_xHMI_Set_ZPA_ZoneTime_Z3;
q_xHMI_Set_ZPA_ZoneTime_Z4          := g_xHMI_Set_ZPA_ZoneTime_Z4;
```

Variable Name	Data Type	Description
g_xHMI_EStopLine01Circuit01_Conv	BOOL	Emergency stop state line 1 circuit 1 TRUE: no emergency stop FALSE: emergency stop pending
g_xHMI_EStopLine01Circuit02_Conv	BOOL	Emergency stop state line 1 circuit 2 TRUE: no emergency stop FALSE: emergency stop pending
g_xHMI_EStopLine02Circuit01_Conv	BOOL	Emergency stop state line 2 circuit 1 TRUE: no emergency stop FALSE: emergency stop pending

Variable Name	Data Type	Description
g_xHMI_EStopLine02Circuit02_Conv	BOOL	Emergency stop state line 2 circuit 2 TRUE: no emergency stop FALSE: emergency stop pending
g_xHMI_Plant_On_Conv	BOOL	Machine state TRUE: running mode FALSE: stop mode
g_xHMI_Remote_Alrm_Line_01_Con	BOOL	Alarm line 1 TRUE: alarm pending FALSE: no alarm pending
g_xHMI_Remote_Alrm_Line_02_Con	BOOL	Alarm line 2 TRUE: alarm pending FALSE: no alarm pending
g_xHMI_AlrmRele_Conv	BOOL	Conveyor alarm is released
g_xHMI_Stat_Auto_Mode_Conv	BOOL	Machine state for automatic mode TRUE: automatic mode FALSE: no automatic mode
g_xHMI_Stat_Manual_Mode_Conv	BOOL	Machine state for manual mode TRUE: manual mode FALSE: no manual mode
g_xALRM_EStopLine01Circuit01_Con	BOOL	Alarm, emergency stop state line 1 circuit 1 TRUE: no emergency stop FALSE: emergency stop pending
g_xALRM_EStopLine01Circuit02_Con	BOOL	Alarm, emergency stop state line 1 circuit 2 TRUE: no emergency stop FALSE: emergency stop pending
g_xALRM_EStopLine02Circuit01_Con	BOOL	Alarm, emergency stop state line 2 circuit 1 TRUE: no emergency stop FALSE: emergency stop pending
g_xALRM_EStopLine02Circuit02_Con	BOOL	Alarm, emergency stop state line 2 circuit 2 TRUE: no emergency stop FALSE: emergency stop pending
g_xALRM_Remote_Alrm_Line_01_Con	BOOL	Alarm line 1 TRUE: alarm pending FALSE: no alarm
g_xALRM_Remote_Alrm_Line_02_Con	BOOL	Alarm line 2 TRUE: alarm pending FALSE: no alarm

WRITE Data to M251 Logic Controller: Conveyor Application

The Magelis HMIGTO4310 device communicates with the Modicon M251 Logic Controller as front end processor. The Modicon M251 Logic Controller communicates with the Modicon M241 Logic Controller. The data to be written by the Modbus TCP IOScanner of the Modicon M251 Logic Controller (command direction) to the Modicon M241 Logic Controller are mapped to the following variables. The HMI application writes these data using SoMachine protocol.

Variable Name	Data Type	Description
g_xHMI_Switch_to_Auto_Mode_Conv	BOOL	HMI command to switch the conveyor to automatic mode TRUE: switch FALSE: no action
g_xHMI_Switch_to_Manual_Mode_Con	BOOL	Command to switch the conveyor to manual mode TRUE: switch FALSE: no action
g_xHMI_RstAlrm_Conv	BOOL	Reset alarm state TRUE: reset FALSE: no action

READ Data from M251 Logic Controller: ZPA Application

The Magelis HMIGTO4310 device communicates with the Modicon M251 Logic Controller as front end processor.The Modicon M251 Logic Controller communicates with the Modicon M221 Logic Controller. The data read by the Modbus TCP IOScanner (monitoring) from the Modicon M221 Logic Controller are mapped to the followingvariables. The HMI application reads these data using SoMachine protocol.

Variable Name	Data Type	Description
g_xHMI_EStopLine01Circuit01_ZPA	BOOL	Emergency stop state for all 4 zones TRUE: no emergency stop FALSE: emergency stop pending
g_xHMI_Zone_OK_ZPA	BOOL	Machine state TRUE: running mode FALSE: stop mode
g_xHMI_Remote_Alrm_Line_01_ZPA	BOOL	Alarm Zero Pressure Accumulation (ZPA) TRUE: alarm pending FALSE: no alarm pending
g_xHMI_Stat_Auto_Mode_ZPA	BOOL	Machine state for automatic mode TRUE: automatic mode FALSE: no automatic mode
g_xHMI_Stat_Manual_Mode_ZPA	BOOL	Machine state for manual mode TRUE: manual mode FALSE: no manual mode

Variable Name	Data Type	Description
g_xHMI_Stat_Remote_Mode_ZPA	BOOL	Command level priority TRUE: command level remote HMI FALSE: command level local HMI or I/O
g_xALRM_EStopLine01Circuit01_ZPA	BOOL	Emergency stop state of zone 1 TRUE: no emergency stop FALSE: emergency stop pending
g_xALRM_Remote_Alrm_Line_01_ZPA	BOOL	Alarm ZPA TRUE: alarm pending FALSE: no alarm pending

WRITE Data to M251 Logic Controller: ZPA Application

The Magelis HMIGTO4310 device communicates with the Modicon M251 Logic Controller as front end processor. The Modicon M251 Logic Controller communicates with the Modicon M221 Logic Controller. The data to be written by the Modbus TCP IOScanner of the Modicon M251 Logic Controller (command direction) to the Modicon M221 Logic Controller are mapped to the following variables. The HMI application writes these data using SoMachine protocol.

Variable Name	Data Type	Description
g_xHMI_Switch_to_Auto_Mode_ZPA	BOOL	HMI command to switch the conveyor to automatic mode TRUE: switch FALSE: no action
g_xHMI_Switch_to_Manual_Mode_ZPA	BOOL	Command to switch the conveyor to manual mode TRUE: switch FALSE: no action
g_xHMI_RstAlrm_ZPA	BOOL	Reset alarm state TRUE: reset FALSE: no action
g_xHMI_New_Vsd_Spd_set_Valid_ZPA	BOOL	Set speed value active for the variable speed drives
g_iHMI_Zpa_Vsd_spd_set_ZPA	WORD	Speed value to be set for all used variable speed drives (no scaling executed in ZPA application)
g_iHMI_Zpa_ZoneTime_Z1	WORD	TIMER set value for zone 1 (s)
g_iHMI_Zpa_ZoneTime_Z2	WORD	TIMER set value for zone 2 (s)
g_iHMI_Zpa_ZoneTime_Z3	WORD	TIMER set value for zone 3 (s)
g_iHMI_Zpa_ZoneTime_Z4	WORD	TIMER Set Value for zone 4 (s)
g_xHMI_Set_ZPA_ZoneTime_Z1	BOOL	Set timer value active for zone 1
g_xHMI_Set_ZPA_ZoneTime_Z2	BOOL	Set timer value active for zone 2
g_xHMI_Set_ZPA_ZoneTime_Z3	BOOL	Set timer value active for zone 3
g_xHMI_Set_ZPA_ZoneTime_Z4	BOOL	Set timer value active for zone 4

Handling of Energy Values Provided by the Power Meter

Overview

This architecture includes a iEM3150 power meter, measuring the energy consumption of the application.

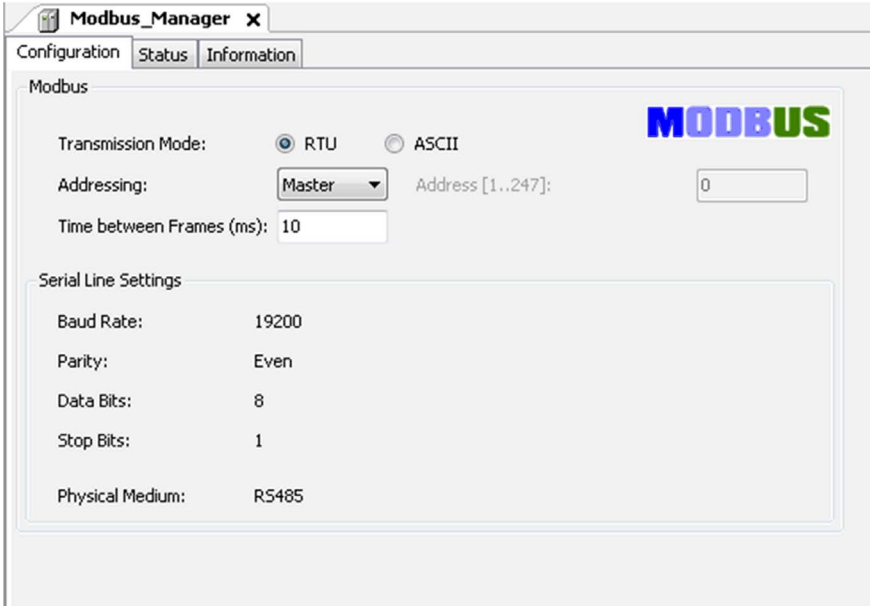
The communication with the power meter is established through the Modbus serial line of the Modicon M251 Logic Controller.

To monitor the energy values on the HMI, follow the steps below:

- Use the application function block (AFB) to read the energy values from the power meter.
- Provide the values read from the power meter to the selected AFBs of the Machine Energy Dashboard library.
- Select the dedicated graphical object linked to the selected AFB in Vijeo Designer.

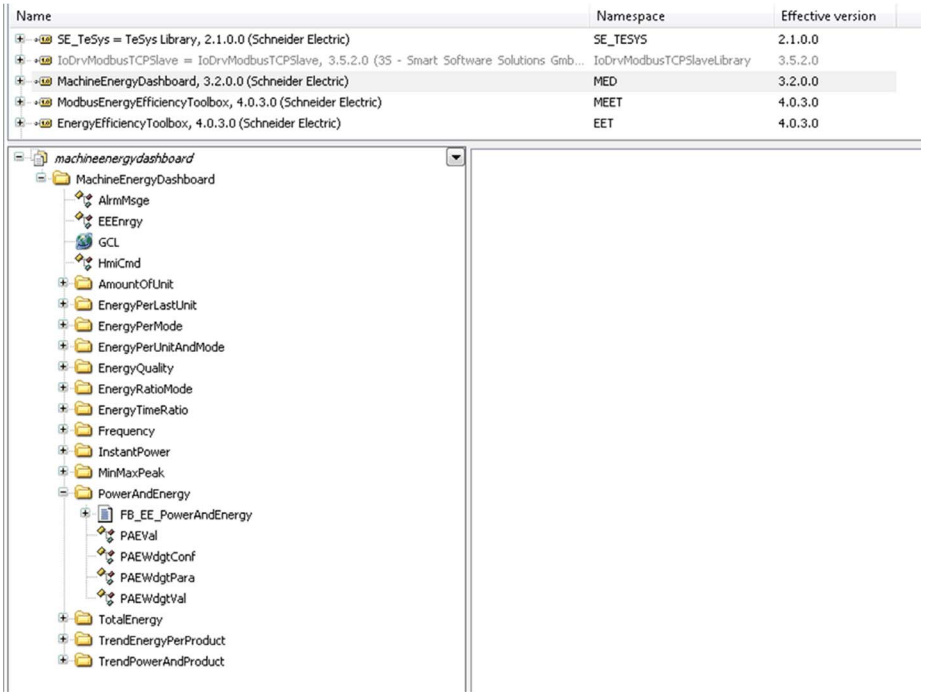
Serial Line Configuration

The following illustration presents the serial line configuration in SoMachine to read data from the power meter.



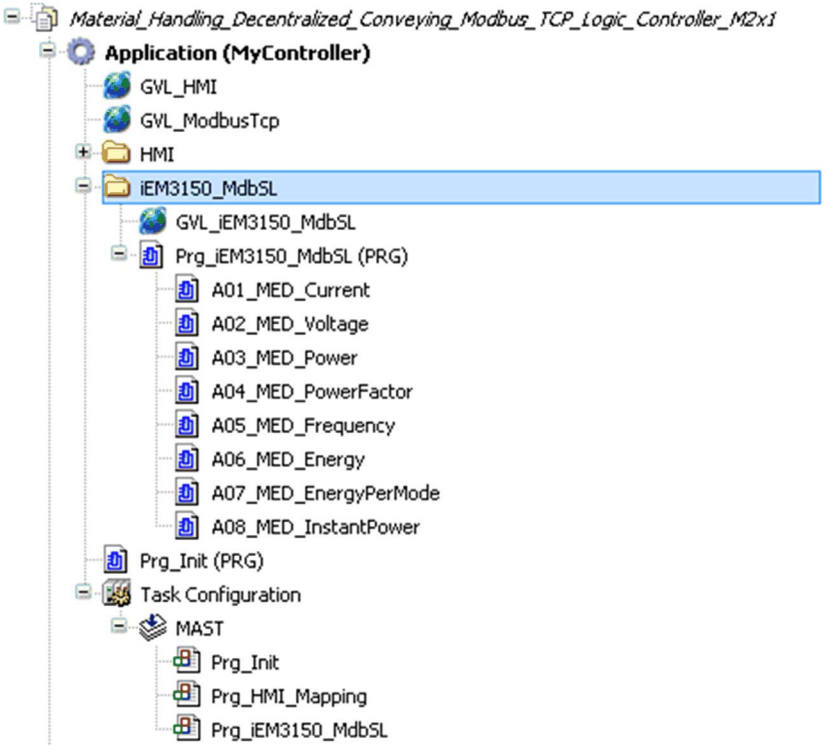
Used Libraries

The following illustration indicates the two libraries, the Machine Energy Dashbord library and the Modbus Energy Efficiency Toolbox libraries, used to monitor and present the energy values read from the iEM3150 power meter.

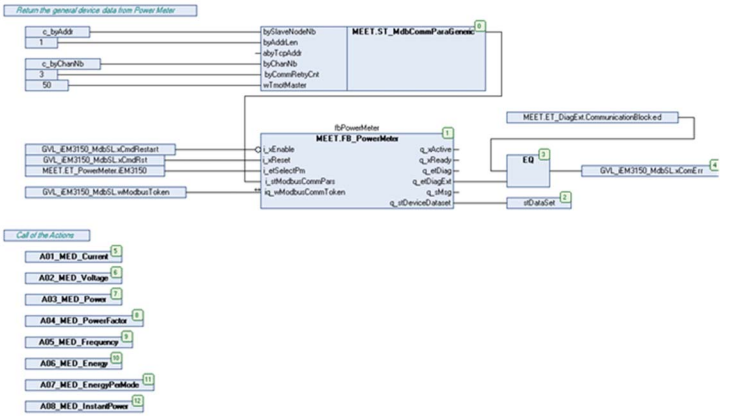
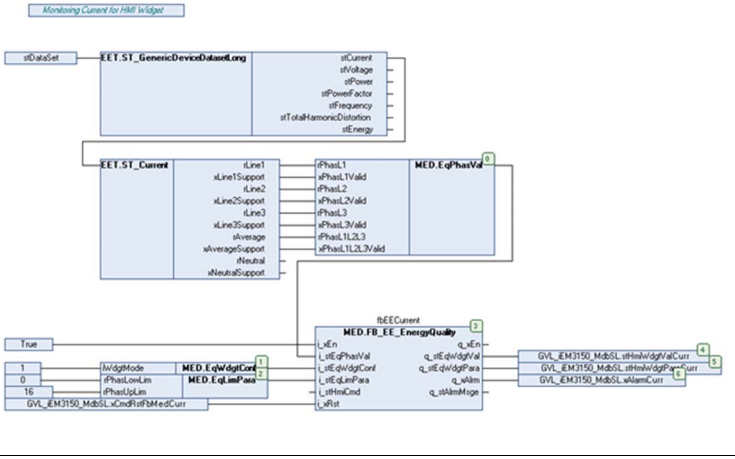


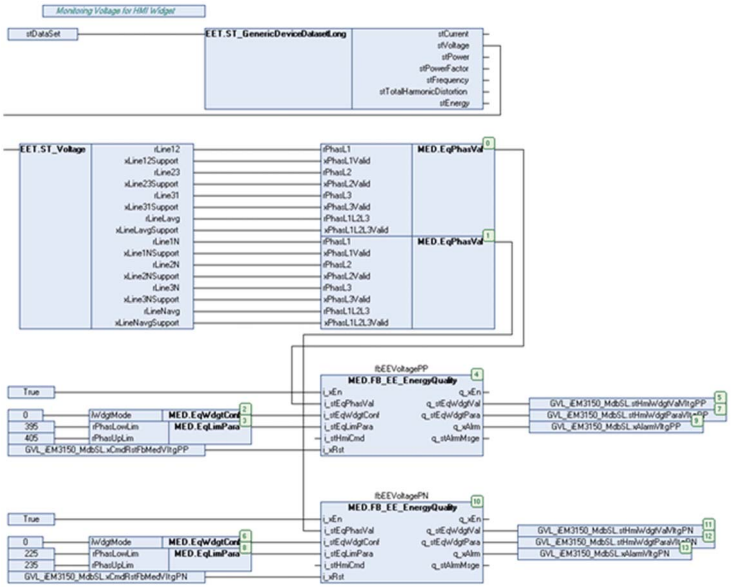
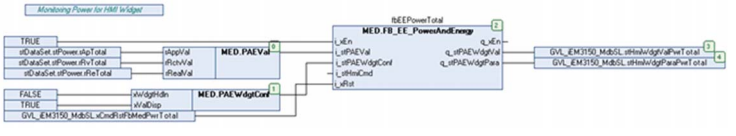
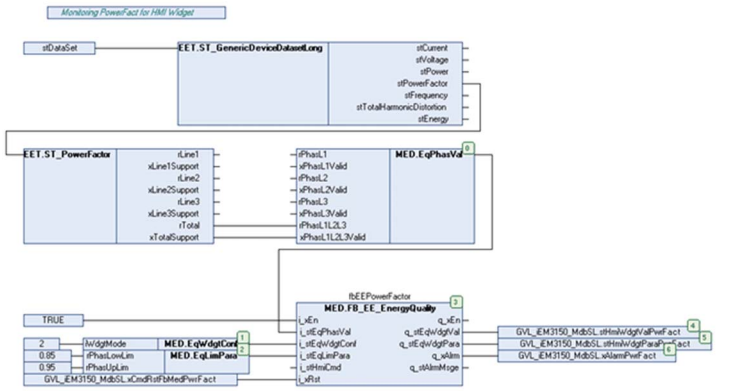
Energy Monitoring Application

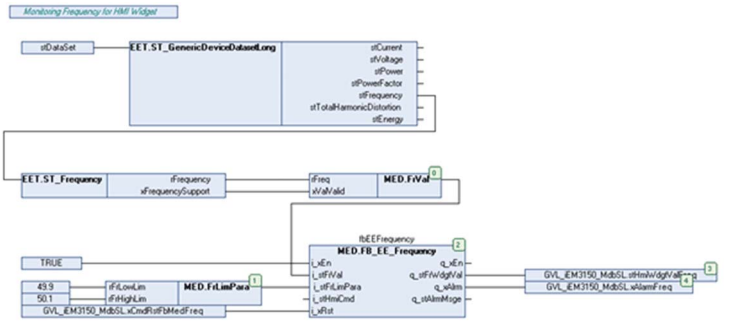
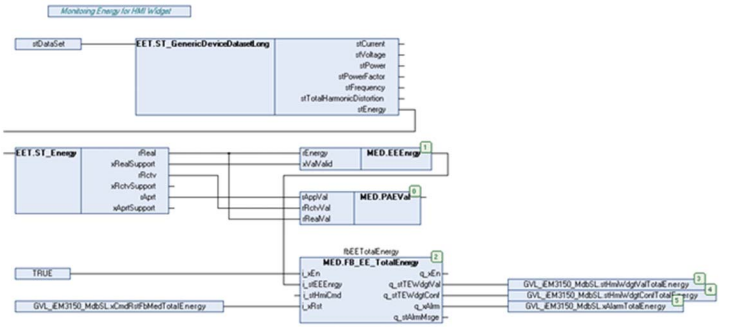
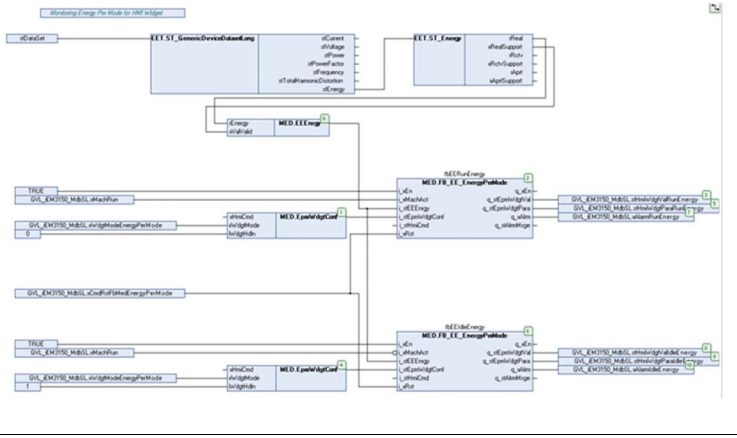
The following illustration presents the energy monitoring application.

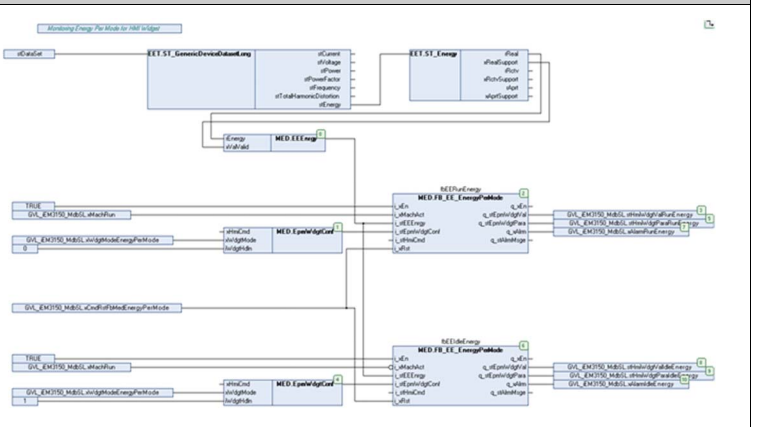


The program POU Prg_iEM3150_MdbSL reads energy values provided by the iEM3150 power meter and links them to the relevant application function blocks (AFBs) provided with Machine Energy Dashboard library.

SoMachine	
Monitor energy data from the iEM3150 power meter.	
Monitor current	

SoMachine	
Monitor voltage	
Monitor power	
Monitor power factor	

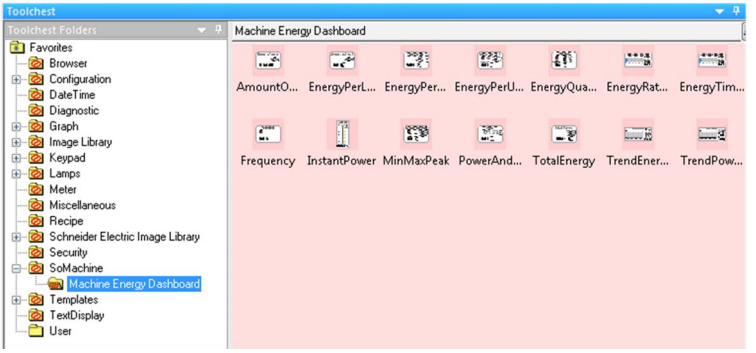
SoMachine	
Monitor frequency	
Monitor energy	
Monitor energy per mode	

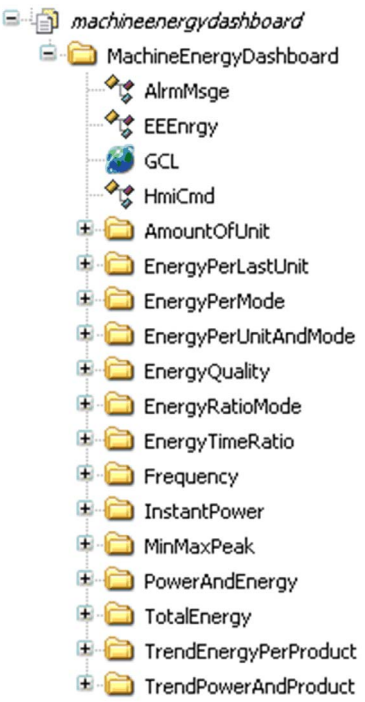
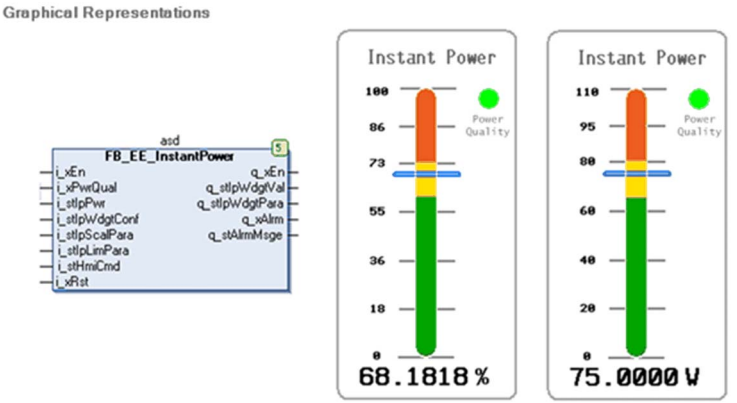
SoMachine	
Monitor instant power	

Magelis HMIGTO4310 Application

Graphical Object

In Vijeo Designer, the graphical objects for energy monitoring are stored in **Toolchest: Machine Energy Dashboard**. To animate these objects use and link the application function blocks (AFBs) of the Machine Energy Dashboard library to the graphical object. .

SoMachine	
Graphical objects of the Machine Energy Dashboard library in Vijeo Designer.	

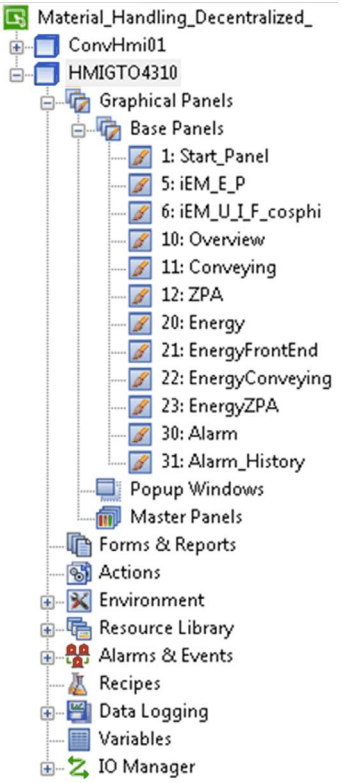
SoMachine	
AFBs of the Machine Energy Dashboard library.	
Example: Graphical representation for the energy value Instant Power.	

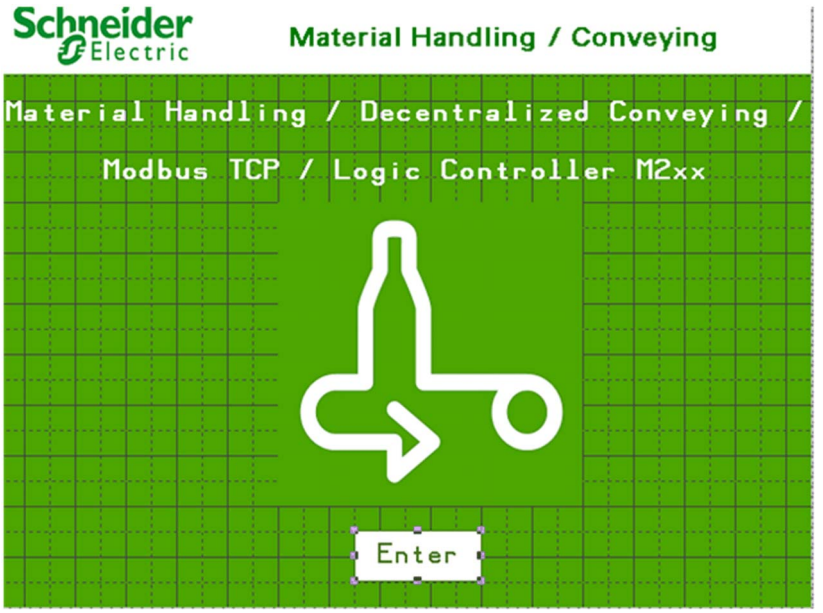
Application Overview

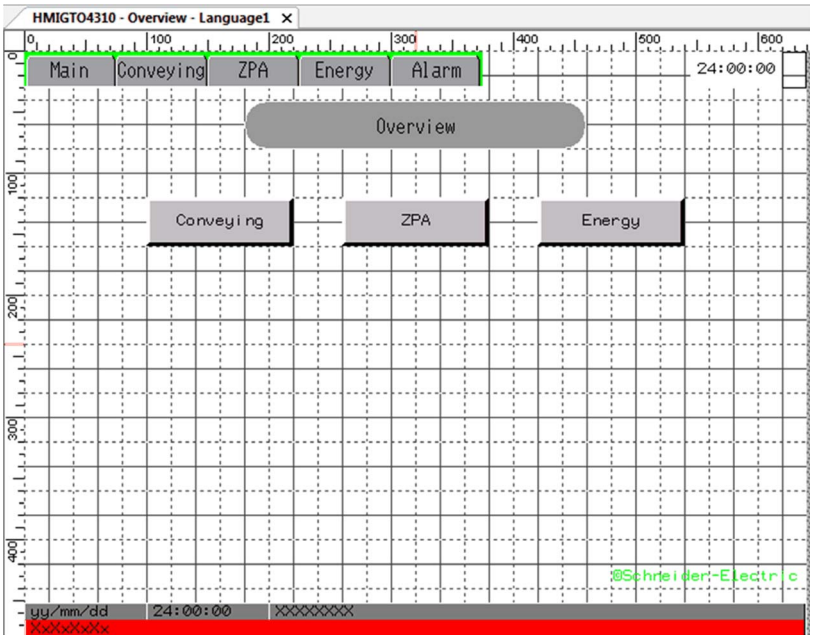
The two main objectives of this application are:

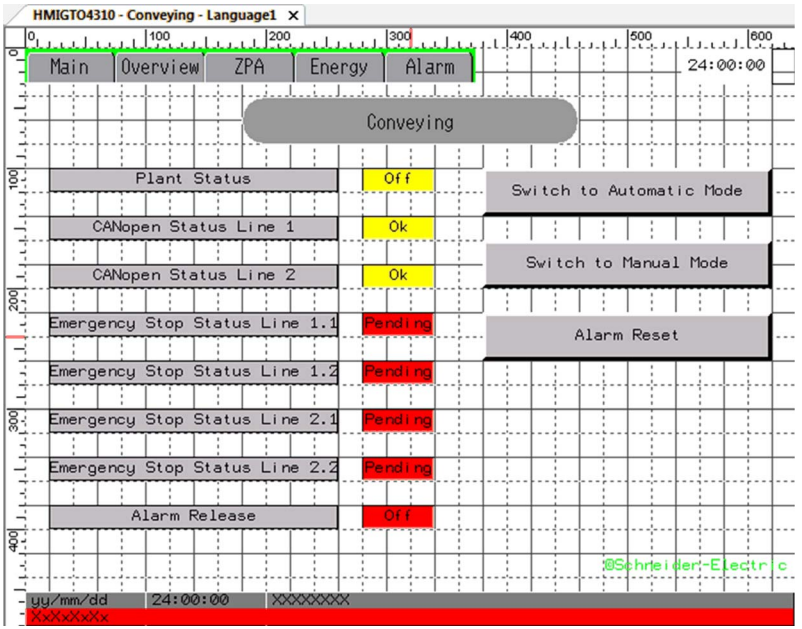
- To monitor and control the conveying zone and the zero pressure accumulation (ZPA) zone.
- To monitor the energy values provided by the iEM3150 power meter linked to the Modicon M251 Logic Controller.

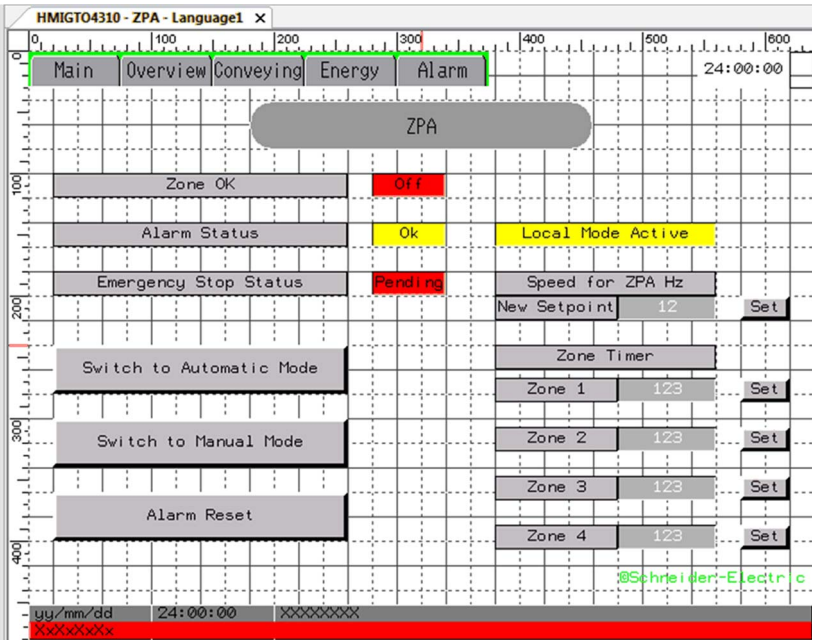
The following table presents the application overview in Vijeo Designer:

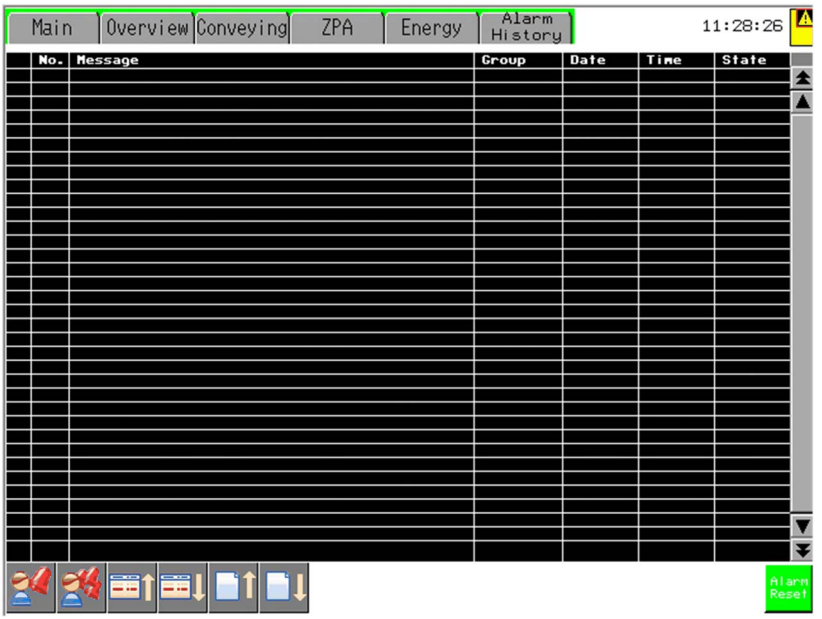
Elements	
HMI application tree	

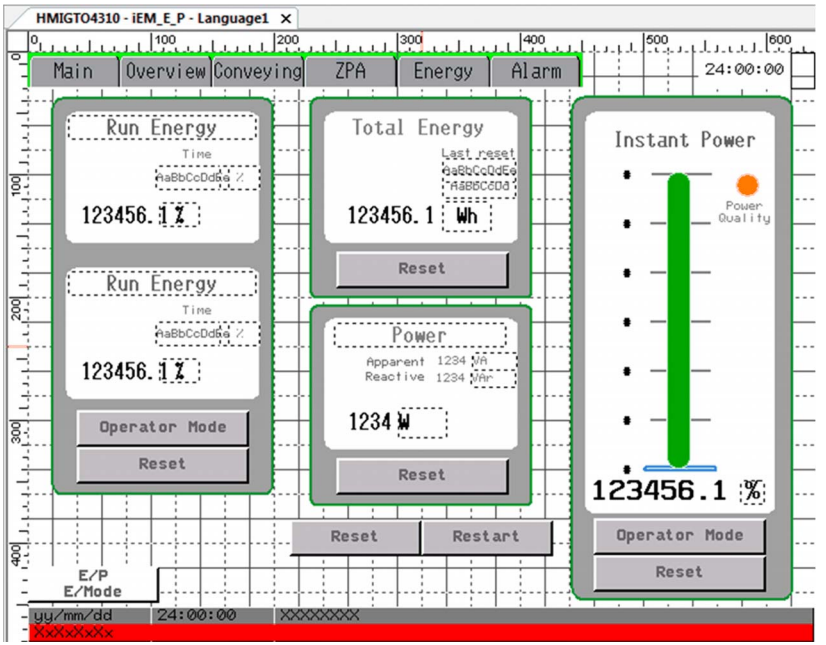
Elements	
Start panel	

Elements	
Overview panel for navigation. Use the main buttons: Conveying, ZPA, Energy to navigate.	

Elements	
Panel to monitor and control the conveying zone.	

Elements	
Panel to monitor and control the ZPA zone.	

Elements	
Panel for alarm management	

Elements	
Main panel for energy monitoring: Energy at RUN time Energy at IDLE time Total Energy Power Instant Power	

Elements	
Panel of additional monitoring of energy values Current Frequency Voltage per phase