TeSys T LTMR

Motor Management Controller
DeviceNet Communication Guide

07/2018

DOCA0133EN-01

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.

You agree not to reproduce, other than for your own personal, noncommercial use, all or part of this document on any medium whatsoever without permission of Schneider Electric, given in writing. You also agree not to establish any hypertext links to this document or its content. Schneider Electric does not grant any right or license for the personal and noncommercial use of the document or its content, except for a non-exclusive license to consult it on an "as is" basis, at your own risk. All other rights are reserved.

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.

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, 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 Book

At a Glance

Document Scope

This guide describes the DeviceNet network protocol version of the TeSys™ T LTMR motor management controller and LTME expansion module.

The purpose of this guide is to:
- Describe and explain the monitoring, protection, and control functions of the LTMR controller and LTME expansion module
- Provide all the information necessary to implement and support a solution that best meets your application requirements

The guide describes the four key parts of a successful system implementation:
- Installing the LTMR controller and LTME expansion module
- Commissioning the LTMR controller by setting essential parameter values
- Using the LTMR controller and LTME expansion module, both with and without additional human-machine interface devices
- Maintaining the LTMR controller and LTME expansion module

This guide is intended for:
- Design engineers
- System integrators
- System operators
- Maintenance engineers

Validity Note

This guide is valid for LTMR DeviceNet controllers. Some functions are available depending on the software version of the controller.

Related Documents

Title of Documentation	Description	Reference Number
TeSys T LTMR - Motor Management Controller - User Guide	This guide introduces the complete TeSys T range and describes the main functions of the TeSys T LTMR motor management controller and LTME expansion module.	*DOCA0127EN*
TeSys T LTMR - Motor Management Controller - Installation Guide	This guide describes the installation, commissioning, and maintenance of the TeSys T LTMR motor management controller and LTME expansion module.	*DOCA0128EN*
TeSys T LTMR - Motor Management Controller - Ethernet Communication Guide	This guide describes the Ethernet network protocol version of the TeSys T LTMR motor management controller.	*DOCA0129EN*
TeSys T LTMR - Motor Management Controller - Modbus Communication Guide	This guide describes the Modbus network protocol version of the TeSys T LTMR motor management controller.	*DOCA0130EN*
TeSys T LTMR - Motor Management Controller - PROFIBUS DP Communication Guide	This guide describes the PROFIBUS-DP network protocol version of the TeSys T LTMR motor management controller.	*DOCA0131EN*
TeSys T LTMR - Motor Management Controller - CANopen Communication Guide	This guide describes the CANopen network protocol version of the TeSys T LTMR motor management controller.	*DOCA0132EN*
TeSys® T LTM CU - Control Operator Unit - User Manual	This manual describes how to install, configure, and use the TeSys T LTMCU Control Operator Unit.	*1639581EN*
Compact Display Units - Magelis XBT N/XBT R - User Manual	This manual describes the characteristics and presentation of the XBT N/XBT R display units.	*1681029EN*
TeSys T LTMR Ethernet/IP with a Third-Party PLC - Quick Start Guide	This guide provides a single reference for configuring and connecting the TeSys T and the Allen-Bradley programmable logic controller (PLC).	*DOCA0119EN*
TeSys T LTM R Modbus - Motor Management Controller - Quick Start Guide	This guide uses an application example to describe the different steps to quickly install, configure, and use TeSys T for Modbus network.	*1639572EN*

Title of Documentation	Description	Reference Number
TeSys T LTM R Profibus-DP - Motor Management Controller - Quick Start Guide	This guide uses an application example to describe the different steps to quickly install, configure, and use TeSys T for PROFIBUS-DP network.	*1639573EN*
TeSys T LTM R CANopen - Motor Management Controller - Quick Start Guide	This guide uses an application example to describe the different steps to quickly install, configure, and use TeSys T for CANopen network.	*1639574EN*
TeSys T LTM R DeviceNet - Motor Management Controller - Quick Start Guide	This guide uses an application example to describe the different steps to quickly install, configure, and use TeSys T for DeviceNet network.	*1639575EN*
Electromagnetic Compatibility - Practical Installation Guidelines	This guide provides an insight to the electromagnetic compatibility.	*DEG999EN*
TeSys T LTM R•• - Instruction Sheet	This document describes the mounting and connection of the TeSys T LTMR motor management controller.	*AAV7709901*
TeSys T LTM E•• - Instruction Sheet	This document describes the mounting and connection of the TeSys T LTME expansion module.	*AAV7950501*
Magelis Compact Terminals XBT N/R/RT - Instruction Sheet	This document describes the mounting and connection of the Magelis XBT-N display units.	*1681014*
TeSys T LTM CU• - Instruction Sheet	This document describes the mounting and connection of the TeSys T LTMCU control unit	*AAV6665701*
TeSys T DTM for FDT Container - Online Help	This online help describes the TeSys T DTM and the custom logic editor embedded in the TeSys T DTM which allows the customization of the control functions of the TeSys T motor management system.	*1672614EN*
TCSMCNAM3M002P USB to RS485 Converter - Quick Reference Guide	This instruction guide describes the configuration cable between computer and TeSys T: USB to RS485	*BBV28000*
Electrical Installation Guide (Wiki version)	The aim of the Electrical Installation Guide (and now Wiki) is to help electrical designers and contractors to design electrical installations according to standards such as the IEC60364 or other relevant standards.	*www.electrical-installation.org*

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

Trademark Notice

All trademarks are owned by Schneider Electric Industries SAS or its affiliated companies.

Chapter 1
Introducing the TeSys T Motor Management System

Presentation of the TeSys T Motor Management System

Aim of the Product

The TeSys T motor management system offers protection, control, and monitoring capabilities for single-phase and three-phase AC induction motors.

The system is flexible, modular, and can be configured to meet the requirements of applications in industry. The system is designed to meet the needs for integrated protection systems with open communications and a global architecture.

Highly accurate sensors and solid-state full motor protection provide better utilization of the motor. Complete monitoring functions enable analysis of motor operating conditions and faster responses to prevent system downtime.

The system offers diagnostic and statistics functions and configurable warnings and faults, allowing better prediction of component maintenance, and provides data to continuously improve the entire system.

For more details on the product, refer to the *TeSys T LTMR Motor Management Controller User Guide*.

Chapter 2
Wiring of the DeviceNet Network

Overview

This chapter describes how to connect the LTMR controller to a DeviceNet network with an open-style connector.

It presents an example of DeviceNet network topology and list cable specifications.

⚠ WARNING

LOSS OF CONTROL

- The designer of any control scheme must consider the potential failure modes of control paths and, for certain critical 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.
- 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 anticipated transmission delays or failures of the link.(1)
- Each implementation of an LTMR controller 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*.

What Is in This Chapter?

This chapter contains the following topics:

DeviceNet Network Characteristics

Overview

The LTMR DeviceNet controller complies with the standard DeviceNet specification.

Physical Layer

DeviceNet's data link layer is defined by the CAN (Controller Area Network) specification and by the implementation of widely available CAN controller chips. CAN also implements a differentially driven (common return), two-wire bus line.

DeviceNet's physical layer contains two twisted pairs of shielded wires. One twisted pair is for transferring data and one is for supplying power. This results in simultaneous support for devices that receive power from the network (like sensors) and those that are self-powered (like actuators). Devices can be added or removed from the bus line without powering down the fieldbus.

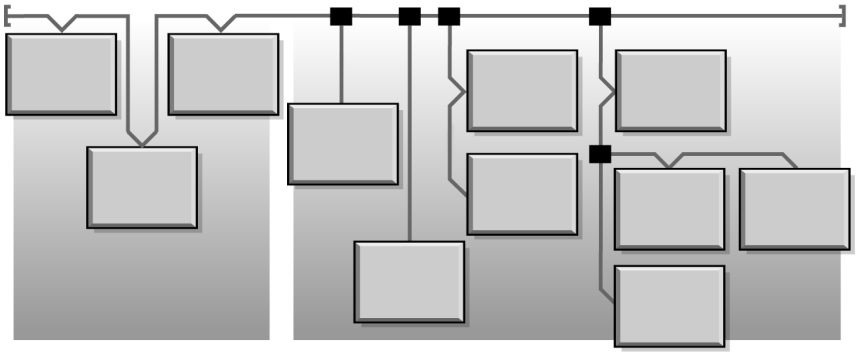
Network Topology

DeviceNet supports a trunk line/drop line network configuration. The implementation of multiple, branched, zero, and daisy chained drops should be established during system design.

The maximum number of slaves connected to one master is 63.

The network must be terminated at each end with 120 Ω resistors.

A sample DeviceNet network topology is shown in the following figure:



1 Trunk line
2 Drop line (0...6 m / 0...20 ft)
3 Daisy chain drop-off
4 Branched drop-off
5 Network node
6 Trunk line tap junction
7 Terminating resistor
8 Zero drop
9 Short drops

Transmission Media

Your implementation of thick, thin, or flat cables for trunk lines and drop lines should be established during system design. Thick cables are generally used for trunk lines. Thin cables can be used for trunk or drop lines.

Maximum Network Lengths

End-to-end network distance varies with data rate and cable size. The following table shows the range of bauds that the Controller supports for CAN devices and the resulting maximum length of the DeviceNet network.

Cable Type	125 kBaud	250 kBaud	500 kBaud
Thick Trunk	500 m (1,640 ft)	250 m (820 ft)	100 m (328 ft)
Thin Trunk	100 m (328 ft)	100 m (328 ft)	100 m (328 ft)
Flat Trunk	420 m (1,378 ft)	200 m (565 ft)	75 m (246 ft)
Maximum Drop Length	6 m (20 ft)	6 m (20 ft)	6 m (20 ft)
Cumulative Drop Length (sum of the length of all drop lines)	156 m (512 ft)	78 m (256 ft)	39 m (128 ft)

Network Model

Like any broadcast communications network, DeviceNet operates within a producer/consumer model. Each data packet's identifier field defines the data priority and allows for efficient data transfer among multiple users. All nodes *listen* on the network for messages with identifiers that apply to their functionality. Messages sent by producer devices are accepted only by designated consumer devices.

DeviceNet supports polled, cyclic, change of state, and explicit data exchange.

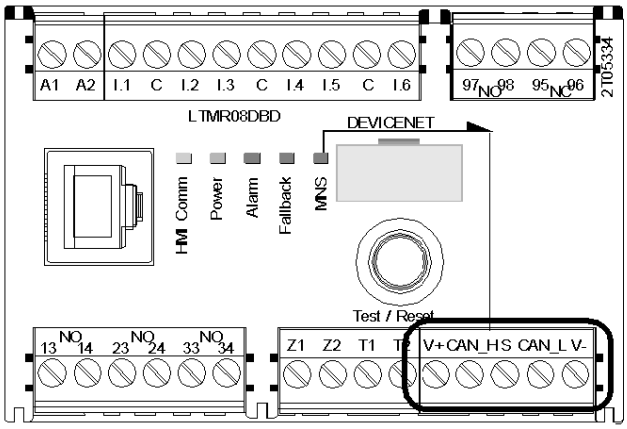
DeviceNet allows users to implement a master/slave, or multi-master network architecture (or some combination thereof), depending on the device's flexibility and your application requirements.

For more information, refer to the Using the DeviceNet Communication Network *(see page 17)*.

DeviceNet Communication Port Wiring Terminal Characteristics

Physical Interface and Connector

The LTMR Controller front face is equipped with one open-style, pull-apart, terminal block for DeviceNet communication.



The DeviceNet communication drivers are powered internally.

Open-Style Terminal Block

The LTMR controller has the following DeviceNet network plug-in terminals and pin assignments.

Pin	Signal	Description
1	V+	Not connected
2	CAN_L	CAN_L bus line (high dominant)
3	S	Shield
4	CAN_H	CAN_H bus line (low dominant)
5	V-	Ground

Open-Style Terminal Block Characteristics

Connector	5 pins
Pitch	5.08 mm (0.2 in.)
Tightening torque	0.5...0.6 N•m (5 lb-in)
Flat screwdriver	3 mm (0.10 in.)

Wiring of the DeviceNet Network

Overview

This section describes the connection of LTMR controllers installed in withdrawable drawers.

DeviceNet Wiring Rules

The following wiring rules must be respected in order to reduce disturbance due to EMC on the behavior of the LTMR controller:

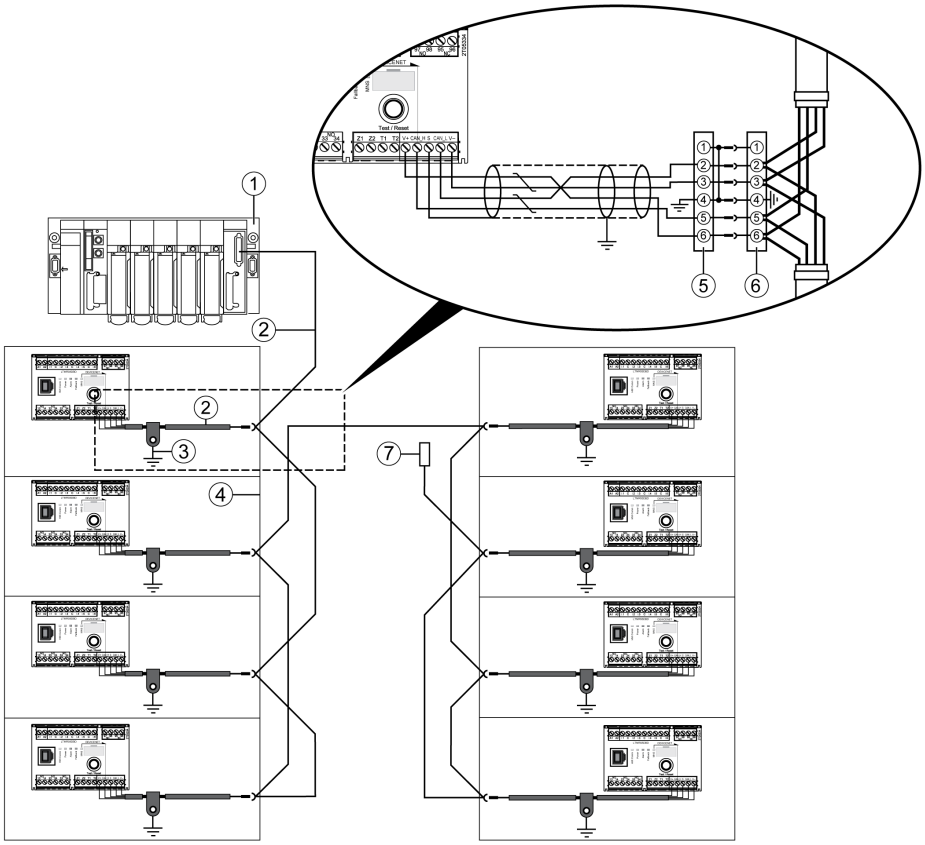
- Keep a distance as large as possible between the communication cable and the power or control cables (minimum 30 cm or 11.8 in.).
- Cross over the DeviceNet cables and the power cables at right angles, if necessary.
- Install the communication cables as close as possible to the grounded plate.
- Do not bend or damage the cables. The minimum bending radius is 10 times the cable diameter.
- Avoid sharp angles of paths or passage of the cable.
- Use the recommended cables only.
- A DeviceNet cable must be shielded:
  - The cable shield must be connected to a protective ground.
  - The connection of the cable shield to the protective ground must be as short as possible.
  - Connect together all the shields, if necessary.
  - Perform the grounding of the shield with a collar.
- When the LTMR controller is installed in a withdrawable drawer:
  - Connect together all the shield contacts of the withdrawable drawer part of the auxiliary connector to the ground of the withdrawable drawer to create an electromagnetic barrier. Refer to the *Okken Communications Cabling & Wiring Guide* (available on request).
  - Do not connect the cable shield at the fixed part of the auxiliary connector.
- Place a line terminator at each end of the bus to avoid malfunctions on the communication bus. A line terminator is already integrated in the master.
- Wire the bus between each connector directly, without intermediate terminal blocks.
- The common polarity (0 V) must be connected directly to protective ground, preferably at one point only for the entire bus. In general, this point is chosen either on the master device or on the polarization device.

For more information, refer to the *Electrical Installation Guide* (available in English only), chapter *ElectroMagnetic Compatibility (EMC)*.

NOTICE
COMMUNICATION MALFUNCTION Respect all the wiring and grounding rules in order to avoid communication malfunctions due to EMC disturbance. **Failure to follow these instructions can result in equipment damage.**

LTMR Controllers Installed in Withdrawable Drawers

The wiring diagram for connection of LTMR controllers installed in withdrawable drawers to the DeviceNet bus is as follows:



1 Master (PLC, PC, or communication module) with line terminator
2 DeviceNet shielded cable
3 Grounding of the DeviceNet cable shield
4 Withdrawable drawer
5 Withdrawable drawer part of the auxiliary connector
6 Fixed part of the auxiliary connector
7 Line terminator VW3 A8 306 DR (120 Ω)

Chapter 3
Using the DeviceNet Communication Network

Overview

This chapter describes how to use the LTMR controller via the network port using the DeviceNet protocol.

⚠ WARNING

LOSS OF CONTROL

- The designer of any control scheme must consider the potential failure modes of control paths and, for certain critical 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.
- 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 anticipated transmission delays or failures of the link.(1)
- Each implementation of an LTMR controller 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".

⚠ WARNING

UNEXPECTED RESTART OF THE MOTOR

Check that the PLC application software:

- Considers the change from local to remote control,
- Manages appropriately the motor control commands during those changes.

When switching to the Network control channels, depending on the communication protocol configuration, the LTMR controller can take into account the latest known state of the motor control commands issued from the PLC and restart automatically the motor.

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

What Is in This Chapter?

This chapter contains the following topics:

DeviceNet Protocol Principles

Overview

The DeviceNet low-level controller area network (CAN) provides a communication link between simple industrial devices (such as actuators and sensors) and controlling devices.

The network carries control data and the properties of the device being controlled. It enables you to operate either in a master/slave mode or a peer-to-peer mode.

The DeviceNet powered 4-wire network operates in a trunk line/drop line configuration and supports up to 64 nodes.

Two main types of messages can be exchanged:
- I/O messaging, dedicated to fast exchanges of process data.
- Explicit messaging, dedicated to slower exchanges such as configuration, settings, or diagnostics data.

Connections and Data Exchange

I/O Messaging

I/O messages contain application-specific data. They are communicated across single or multicast connections between an application producer and its corresponding consuming application. Because I/O messages carry time-critical messages, they have high-priority identifiers.

An I/O Message consists of a Connection ID and associated I/O data. The meaning of the data within an I/O Message is implied by the associated Connection ID. The connection endpoints are assumed to have knowledge of the intended use or meaning of the I/O Message.

I/O Message Types

Slave devices can produce data using one or more of the following I/O message types, depending on how the device is configured and the requirements of the application:

Type	Description of Operation
Polled	A slave configured for polled I/O receives output data from the master device. This data is received in a sequential order that is defined by the master's scan list. The master's polling rate is determined by the number of nodes in the scan list, the DeviceNet baud rate, the size of messages produced by the master and each node in its scan list, and the internal timing of the master device.
Cyclic	A device configured to produce a cyclic I/O message will produce its data at a precisely defined interval. This type of I/O messaging allows the user to configure the system to produce data at a rate appropriate for the application. Depending on the application this can reduce the amount of traffic on the wire and more efficiently use the available bandwidth.
Change-of-state	A device configured to produce a change-of-state (COS) message will produce data whenever it changes, or at a base heartbeat rate. This adjustable heartbeat rate enables the consuming device to verify that the producer is still present and active on the network. DeviceNet also defines a user-configurable Production Inhibit Time that limits how often COS messages are produced to prevent nodes from flooding the bandwidth. Users can adjust these parameters to provide optimum bandwidth utilization in a given application.

Explicit Messaging

Explicit messaging connections provide multipurpose point-to-point communication paths between two particular devices. Explicit messages are used to command the performance of a particular task and to report the results of performing the task. You can therefore, use explicit messaging connections to configure nodes and diagnose problems.

DeviceNet defines an explicit messaging protocol that states the meaning or intended use of an explicit message within the CAN (Controller Area Network) data field. The message consists of a Connection ID and associated messaging protocol information.

Idle Message Management

When the LTMR controller receives an Idle Message sent by the DeviceNet network master, it generates a communication loss and the LTMR controller is in fallback condition.

The conditions to exit the idle mode are the same as to exit the fallback condition.

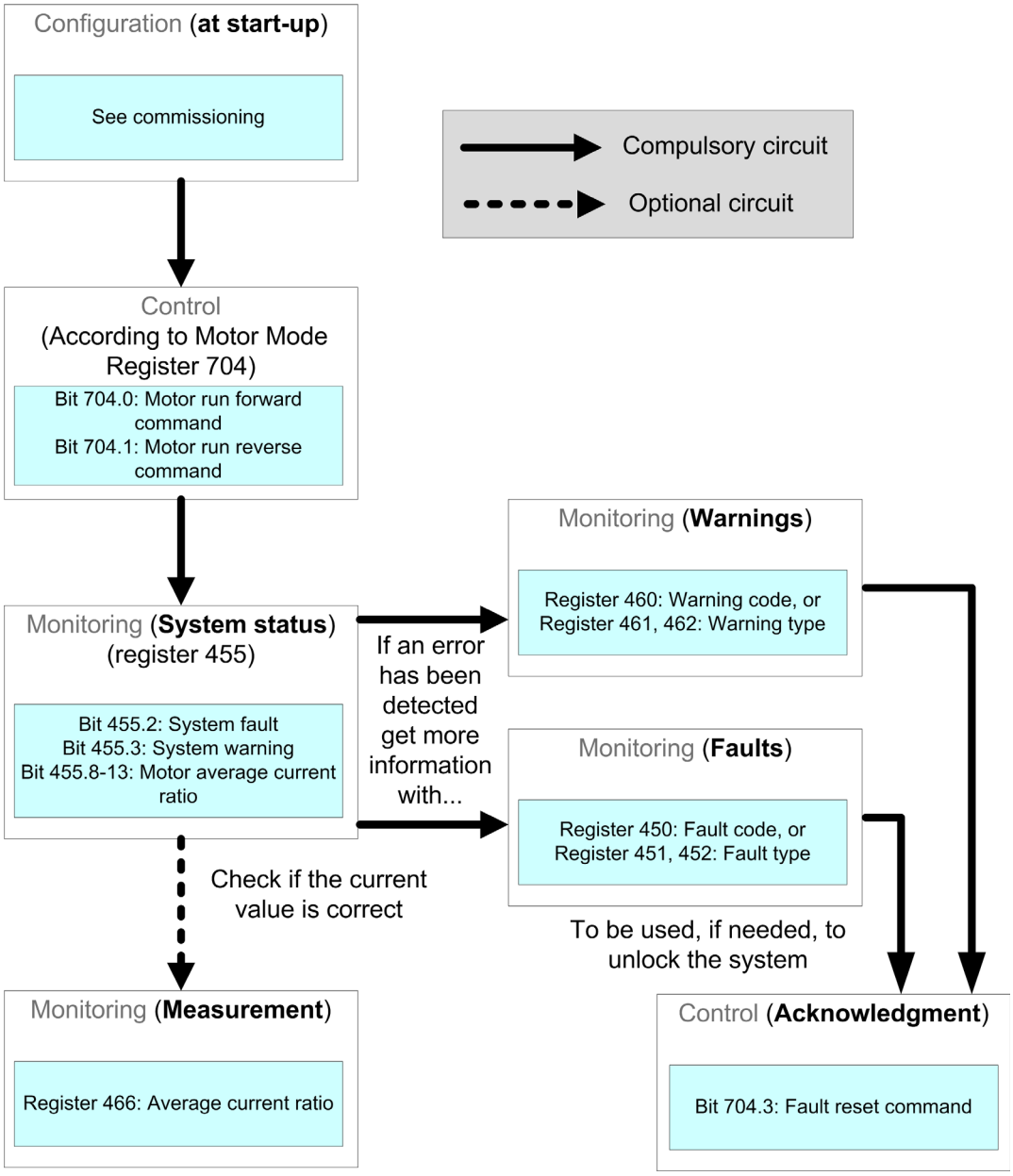
Simplified Control and Monitoring

Overview

This is a simplified example of the main registers which control and monitor a Motor Management Controller.

DeviceNet Registers for Simplified Operation

The following illustration provides basic setup information, using the following registers: configuration, control and monitoring (system status, measurements, faults and warnings, acknowledgement).



Configuration of the LTMR DeviceNet Network Port

Communication Parameters

Use the TeSys T DTM or the HMI to configure the DeviceNet communication parameters:

- Network Port Address Setting
- Network Port Baud Rate Setting
- Config Via Network Port Enable

Setting the MAC-ID

The MAC-ID is the address of the module on the DeviceNet bus. A DeviceNet network is limited to 64 addressable nodes (node IDs 0 to 63). This means that you can assign a MAC-ID of 0-63.

You must set the MAC-ID before any communication can start. To do this, use the TeSys T DTM or the HMI to configure the communication parameter Network Port Address Setting. The factory setting for the address is 63.

Setting the Baud Rate

You can also set a baud rate of the following speeds:

- 125 kBaud
- 250 kBaud
- 500 kBaud

To set the baud rate, use the TeSys T DTM or the HMI to configure the communication parameter Network Port Baud Rate Setting.

The parameter has the following possible settings:

Network Port Baud Rate Setting	Baud Rate
0	125 kBaud (factory setting)
1	250 kBaud
2	500 kBaud
3	Autobaud

Autobaud automatically detects the baud rate required.

NOTE: The Autobaud functionality can only be used if a valid communication is already present on the network, that is to say, that at least one master and one slave are already communicating.

Setting the Configuration Channel

The LTMR configuration can be managed via two different modes:

- Locally through the HMI port using the TeSys T DTM or the HMI
- Remotely through the network

To manage the configuration locally, parameter Config Via Network Port Enable must be disabled to prevent overwriting of the configuration through the network.

To manage the configuration remotely, parameter Config Via Network Port Enable must be enabled (factory setting).

Device Profiles and EDS Files

Device Profiles

DeviceNet's device models define the physical connections and promote interoperability among standard devices.

Devices that implement the same device model must support common identity and communications status data. Device-specific data is contained in *device profiles* that are defined for various device types. Typically, a device profile defines the device's:

- Object model
- I/O data format
- Configurable parameters

The above information is made available to other vendors through the device's EDS (electronic data sheet).

For a full description of the objects in the LTMR device profile, refer to the Object Dictionary *(see page 36)*.

What's an EDS?

The EDS is a standardized ASCII file that contains information about a network device's communications functionality and the contents of its object dictionary *(see page 36)*, as defined by ODVA (Open DeviceNet Vendor Association). The EDS also defines device-specific and manufacturer-specific objects.

Using the EDS, you can standardize tools to:

- Configure DeviceNet devices
- Design networks for DeviceNet devices
- Manage project information on different platforms

The parameters of a particular device depend on those objects (parameter, application, communications, emergency, and other objects) that reside on the device.

LTMR Controller EDS Files

EDS files and associated icons that describe the various configurations of the LTMR Controller can be downloaded from www.schneider-electric.com website (**Products and Services → Automation and Control → Product offers → Motor Control → TeSys T → Downloads → Software/Firmware → EDS&GSD**).

EDS files and icons are grouped in a single compressed Zip file that you must unzip to a single directory on your hard disk drive.

For information on how to register these EDS files in the RSNetworx's EDS library System, refer to the Register the Controller's EDS *(see page 25)*.

Configuring the DeviceNet network

Introduction

Use these sample instructions to configure for example a Rockwell Automation® SLC-500 PLC (1747-SDN) with a DeviceNet controller at the head of a TeSys T Motor Management system. The configuration software is RSNetworx for DeviceNet configuration software. The stages of this process are described in the following table:

Stage	Description
1	Assemble the DeviceNet network *(see page 25)*
2	Register the controller's EDS files *(see page 25)*
3	Connect devices to your network *(see page 26)*
4	Upload the controller configuration *(see page 30)*
5	Add the controller to the Scanlist *(see page 30)*
6	Edit the I/O parameters *(see page 31)*
7	Save the configuration *(see page 33)*

Before You Begin

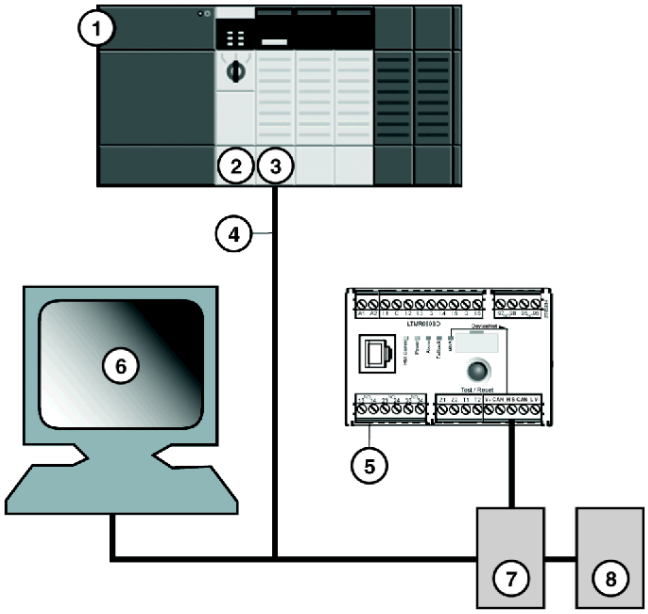
Before you begin, make sure:

- The TeSys T Motor Management system is fully assembled, installed, and powered according to your particular system, application, and network requirements.
- You have properly set the network port *(see page 22)* of the controller.
- You have the basic EDS files *(see page 23)* and corresponding .ico files that are available at *www.schneider-electric.com*), or you have generated an EDS that is specific to the system assembly.

To configure the controller using RSNetWorx, you must have a working familiarity with both the DeviceNet fieldbus protocol and RSNetWorx for DeviceNet. (The described procedures cannot practically anticipate every prompt or option you may encounter during configuration.)

Connection Figure

Before assembling the network, familiarize yourself with the required hardware connections. The following figure shows the DeviceNet network connections between an Allen-Bradley PLC, the controller, and RSNetWorx:



1 Allen-Bradley SLC-500 PLC
2 PLC processor module
3 1747-SDN DeviceNet scanner module
4 DeviceNet network cable
5 LTMR controller
6 PC running RSNetWorx (properly connected to your network)
7 Power tap

8 DeviceNet power supply 24 Vdc

The scanner module is the control mechanism for all network traffic. It reads and writes every piece of I/O data that is moved on the network.

Assemble the Physical Network

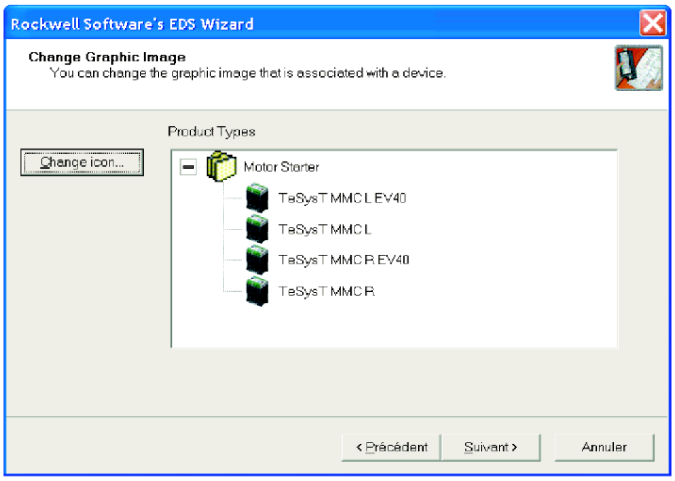
The following procedure describes the connections required to construct a physical DeviceNet network.

Step	Action	Comment
	⚠ CAUTION **EQUIPMENT DAMAGE IF VOLTAGE IS PRESENT** Read and understand this guide and the Allen-Bradley PLC users manual before installing or operating this equipment. This equipment must be installed, adjusted, repaired, and maintained by qualified personnel only. ● Disconnect all power to the PLC before making the network connection. ● Place a DO NOT TURN ON sign on the system power disconnect. ● Lock the disconnect in the open position. You are responsible for conforming to all applicable code requirements with respect to grounding all equipment. **Failure to follow these instructions can result in injury or equipment damage.**	
1	Install the DeviceNet scanner module in the desired PLC slot.	The connection figure *(see page 24)* above shows the scanner in slot 2 of the PLC.
2	Check that the desired DeviceNet network node address *(see page 22)* and baud rate *(see page 22)* have been correctly set.	This example uses an address of *4*.
3	Make connections with DeviceNet network cable and end connectors, manufactured in accordance with ODVA specifications.	The cable and end connectors are not supplied.
4	Place the system on the network by connecting the PLC to the LTMR controller with the DeviceNet cable.	
5	Connect the RSNetWorx PC to the network using the DeviceNet cable.	

Register the Controller's EDS

To register the controller's EDS in RSNetWorx's EDS library:

Step	Action	Comment
1	From the RSNetWorx Tools menu, select EDS Wizard.	The Wizard's welcome screen appears.
2	Click Next.	The Options screen appears.
3	Select Register an EDS files and click Next.	The Registration screen appears.
4	Select Register a directory of EDS files and browse to the controller's EDS file.	You must have already unzipped the Zip file containing the EDS files and corresponding icons into a single directory.
5	Click Next.	The EDS File Installation Test Results screen appears.

Step	Action	Comment
6	Click Next.	The Change Graphic Image screen appears. The controller should be listed in the Product Types field as a Motor Starter: 
7	Click Next.	The Final Task Summary screen appears.
8	Verify that the controller is to be registered and click Next.	The completion screen appears.
9	Click Finish.	The EDS Wizard closes.

Selection Criteria for TeSys T LTMR Controller Variants

There are four EDS files corresponding to the four possible configurations of the TeSys T Motor Management controller system:

Choose...	When You Want to Use...
TeSys T MMC L	A TeSys T Motor Management controller system without an expansion module, configurable via the HMI port. This variant enables you to preserve your local configuration.
TeSys T MMC L EV40	A TeSys T Motor Management controller system with expansion module, configurable via the HMI port. This variant enables you to preserve your local configuration.
TeSys T MMC R	A TeSys T Motor Management controller system without expansion module configurable via the network.
TeSys T MMC R EV40	A TeSys T Motor Management controller system with expansion module configurable via the network.

In **local** configuration mode, the parameter Config via Network Port Enable must be disabled. This mode preserves the local configuration made using the Magelis XBT or SoMove with the TeSys T DTM through the HMI port and prevents PLC configuration via the network.

In **remote** configuration mode, the parameter Config via Network Port Enable must be enabled. This enables the PLC to remotely configure the L R controller.

NOTE: The parameters overwritten by the PLC will be lost. Remote mode is useful when replacing faulty devices.

The Config via Network Port Enable parameter is set by default.

Connect Devices to Your Network

This example requires you to add two devices to your project view:

- An LTMR controller without expansion module configured in remote mode with the address 4.
- A DeviceNet scanner in PLC slot 2 with the address 1

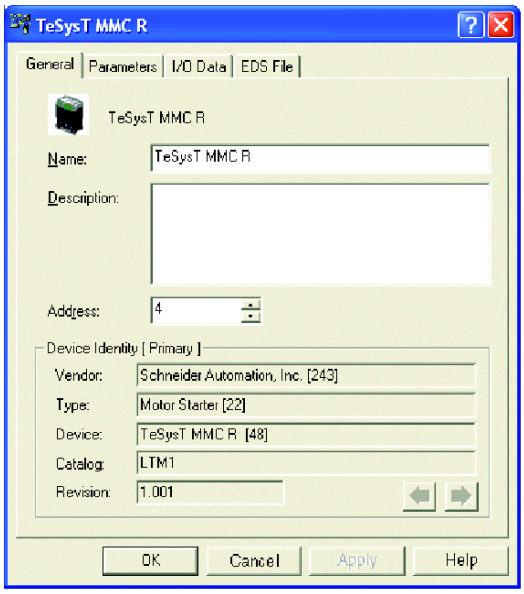
You can use RSNetWorx to configure the devices in either *offline* or *online* mode:

- *Offline*-The configuration tool and the physical network are not connected.
- *Online*-The configuration tool is connected to the physical network. Build the network using the parameters transferred from devices on the physical network.

Connect to network connections using either the offline or online procedures in the tables that follow. (These are standard RSNetWorx procedures.)

Offline Device Connection

Use this procedure for adding devices to your network when the configuration tool is offline:

Step	Action	Comment
1	From the Hardware list, double-click on the controller EDS named TeSys T MMC R under Schneider Automation, Inc.\Motor Starter.	The new device appears in the project view. The lowest available MAC ID has been assigned to it, even if that ID is inappropriate.
2	Double-click on the controller graphic.	The controller's properties window appears.
3	Change the MAC ID in the Address text field to 4.	*4* is the MAC ID used throughout this example.
4	Click OK.	Note that the MAC ID of the controller is now *4* in the project view. 
5	Repeat steps 1 to 4 to add the 1747-SDN Scanner Module to the network with MAC ID *00*.	The scanner's EDS is in the *Hardware* list at Rockwell Automation - Allen Bradley/Communication Adapter.
6	Save your configuration by choosing Save as from the File menu.	Save offline configurations for later use.

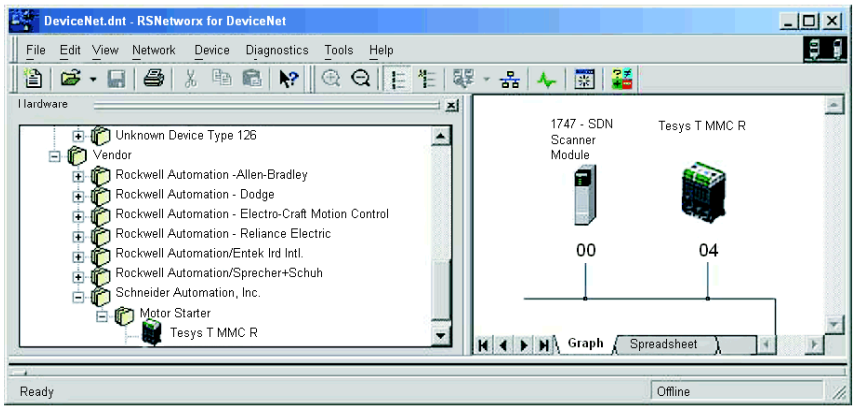
Online Device Connection

Use this procedure for adding devices to your network when the DeviceNet network is already assembled and the configuration tool is online:

Step	Action	Comment
1	From the Network menu, select Online.	The Browse for network screen appears.
2	Set a communication path to select a path, based on your system and application requirements. Click OK.	When the Browsing network screen finishes, the physically connected devices will appear in the project view.
3	Save your configuration by choosing Save as from the File menu.	Save the configuration for later use.

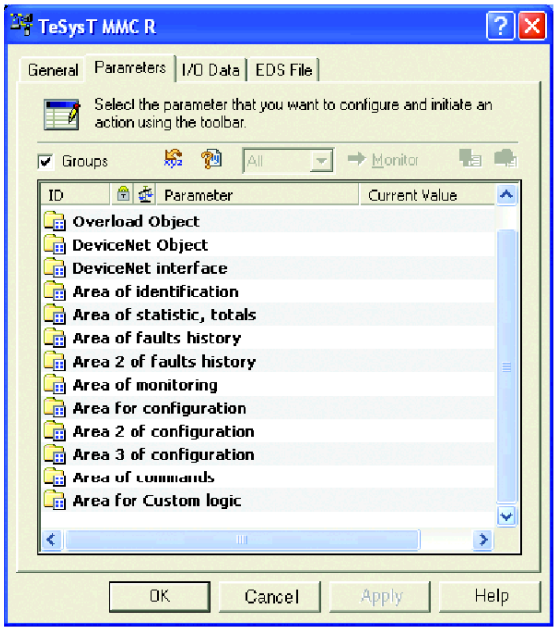
The RSNetWorx Project View

The RSNetWorx project view should resemble the following figure after you have added the controller and the master scanner to your network configuration (using either the online or offline connection procedure):



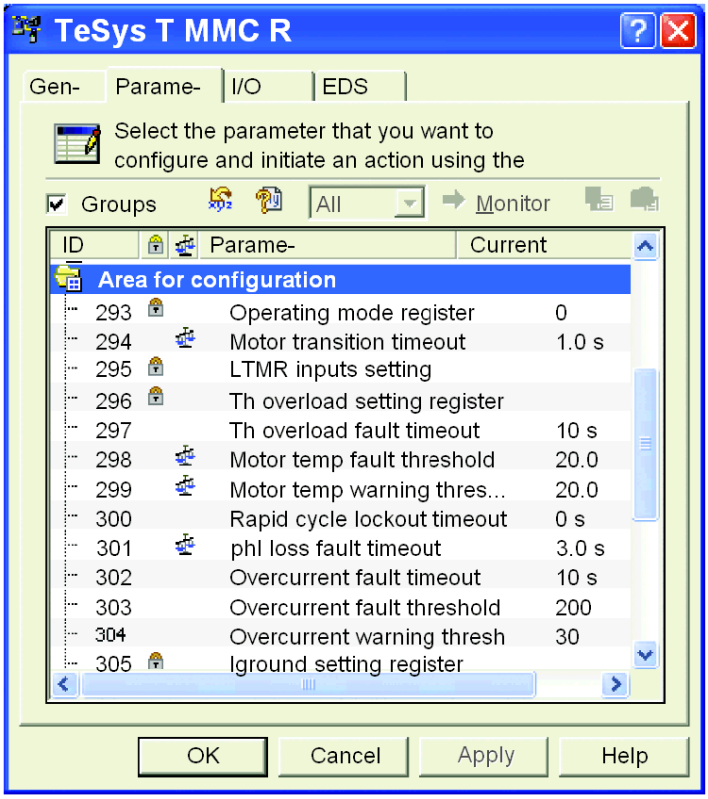
Read and Write LTMR Controller Parameters

To read and write to the controller's parameters:

Step	Action	Comment
1	From the project view, double-click on the controller icon.	The controller configuration screen appears.
2	Select the Parameter tab.	The parameter list appears.
3	Select Group View.	The parameter groups appear. 
4	Select configuration group 1, 2, or 3 to access controller configuration parameters.	For controllers used without expansion modules: • **Area for configuration** includes registers 540 to 564 without expansion module, or 540 to 595 with expansion module • **Area 2 of configuration** includes registers 600 to 645 • **Area 3 of configuration** includes registers 650 to 596 For more details, refer to Register Map - Organization of Communication Variables *(see page 51)* for a complete list of communication variables.
5	Select the parameter you want to access and read from or write to it.	Write access to parameters is only available with TeSys T MMC R and TeSys T MMC R EV40.

The TeSys T MMC R Parameter Screen

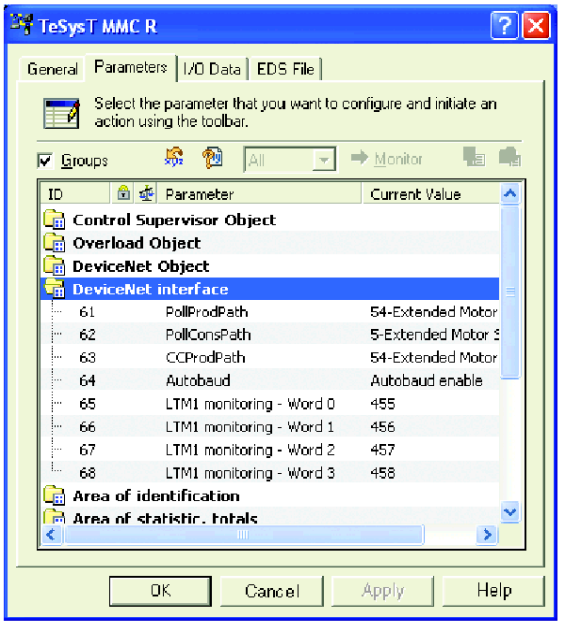
The TeSys T MMC R parameter screen should resemble the following figure:



Select Data Exchanged via I/O Messaging

To select data exchanged through I/O messaging

Step	Action	Comment
1	In the TeSys T MMC R parameter screen, select DeviceNet Interface Group.	The parameter list appears.
2	For the PollProdPath parameter, select the input assembly object you want the controller to produce.	PollProdPath consists of data produced by the controller on polling sent by the scanner.
3	For the PollConsPath parameter, select the output assembly object you want the controller to consume.	PollConsPath consists of data sent by polling by the scanner and consumed by the controller.
4	For the COSProdPath parameter, select the Input Assembly object you want the controller to produce.	COSProdPath consists of data produced by the controller on Change-of-State (COS).

Step	Action	Comment
5	If you selected Input Assembly object 110 or 113 in steps 2 or 4, adjust the LTMR Monitoring Word 0 to 3 to the register you want the controller to produce.	The TeSys T MMC R parameter screen should resemble the following figure:  Only used with instances 110 and 113.

Upload and Download Device Configurations

After the online connection of devices, you must transfer the required device information.

Use the following options from the Device menu to transfer the configurations of only selected devices:

- Download to Device-Transfer the offline configuration from the PC to the device.
- Upload from Device-Transfer the configuration from the device to the PC.

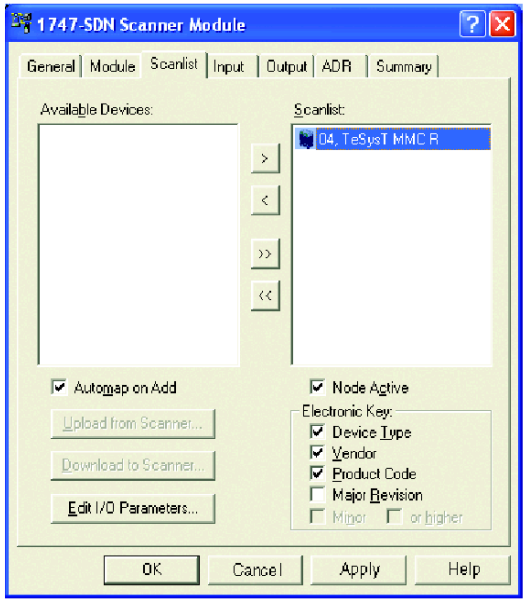
Use the following options from the Network menu to transfer the configurations of all online devices in the project view:

- Download to Network-Transfer the offline configurations from the PC to all online devices.
- Upload from Network-Transfer the configurations of all online devices to the PC.

Add the Controller to the Scanlist

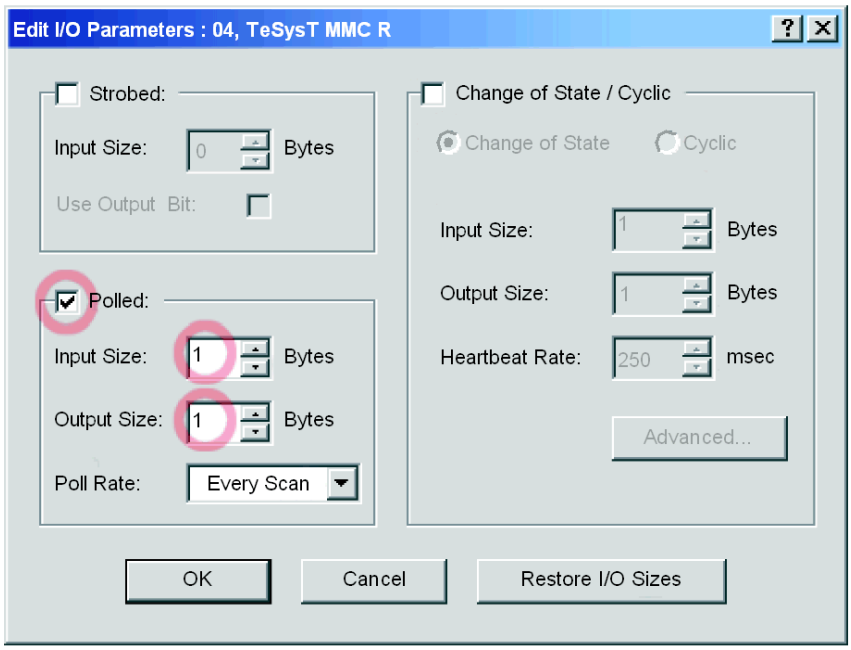
To be recognized on the network, the controller must be added to the master scanner's Scanlist using the online procedure in the following table:

Step	Action	Comment
1	From the project view, double-click the scanner icon.	The scanner configuration screen appears.
2	Select the Scanlist tab.	The Scanner Configuration Applet screen appears.
3	Select Upload from scanner.	Wait for the Uploading from Scanner timer to finish.

Step	Action	Comment
4	At the Scanlist tab, highlight the controller (at MAC ID 4) in the Available Devices list, and click the right arrow.	The controller now appears in the Scanlist. 
5	With the controller selected, click the Edit I/O Parameters button.	The Edit I/O Parameters window appears.
6	Check Polled and enter the correct input size and correct output size (depending on assembly objects previously selected).	The determination of the controller input and output data sizes is described in the next paragraph.
7	Click OK.	The Edit I/O Parameters window closes.
8	Click Download to scanner.	The Downloading Scanlist from Scanner window appears.
9	Click Download.	Wait for the Downloading to Scanner timer to finish.
10	Click OK.	The scanner properties window closes.

The Edit I/O Parameters Screen

The controller's Edit I/O Parameters screen should resemble the following figure after you have customized it as described above:



Depending on your requirements, you can select one of three transmission modes:

- Polled
- Change of State
- Cyclic

NOTE: The controller does not support Strobed I/O messages used for very simple I/O devices.

You have to enter the number of input and output bytes produced by the controller. The master device needs this information to allocate data space for each network node.

The number of input and output bytes the controller produces depends on the instances you select for the DeviceNet Interface object.

The following tables show the byte size of each assembly object you can select for I/O messaging.

Output Assembly data size (consumed by the controller):

Instance	Name	Number of Bytes
2	Basic Overload	1
3	Basic Motor Starter	1
4	Extended Contactor	1
5	Extended Motor Starter	1
100	LTMR Control Registers	6
101	PKW Request Object	8
102	PKW Request and Extended Motor Starter	10
103	PKW Request and LTMR Control Registers	14

Input Assembly data size (produced by the controller):

Instance	Name	Number of Bytes
50	Basic Overload	1
51	Extended Overload	1
52	Basic Motor Starter	1
53	Extended Motor Starter 1	1
54	Extended Motor Starter 2	1
110	LTMR Monitoring Registers (with dynamic configuration)	8
111	PKW Response Object	8
112	PKW Response and Extended Motor Starter	10
113	PKW Response and LTMR Monitoring Registers	16

Create an EDS for the Controller

Devices that do not correspond to specific EDS files during online network browsing will appear in the project view as Unrecognized Devices. If your controller is not recognized, you must create an EDS using the following procedure:

Step	Action	Comment
1	In the project view, double-click the controller.	You will be asked if you want to register the controller with the EDS Wizard.
2	Click Yes.	The Wizard's welcome screen appears.
3	Click Next.	The Options screen appears.
4	Select Create an EDS file and click Next.	RSNetWorx will upload the controller's identity information, displayed in the Device Description screen.
5	Record the product name string, *LTM1*, and click Next.	The Input/Output screen appears.
6	Check Polled and enter the appropriate values for input and output sizes. Also check COS and enter an input size value of 1. Click Next.	
7	Change the icon, if you wish, at the Change Graphic Image and click Next.	The Final Task Summary screen appears.
8	Verify that the controller is to be registered and click Next.	The completion screen appears.
9	Click Finish.	The EDS Wizard closes.
10	Add the controller to the Scanlist *(see page 30)*.	

Saving the Configuration

Save your configuration by selecting **File** → **Save** from the RSNetworx menu. This is a standard Windows command.

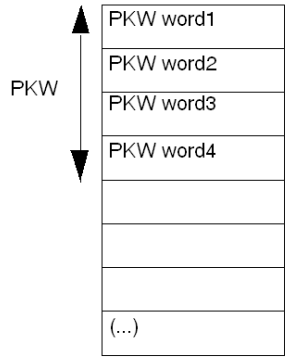
PKW Objects

Overview

The LTMR controller supports PKW (**P**eriodically **K**ept in acyclic **W**ords). The PKW feature consists of:
- 4 input words mapped in input assembly objects 111, 112, and 113
- 4 output words mapped in output assembly objects 101, 102, and 103

These 4 words tables enable a DeviceNet scanner to read or write any register using I/O messaging.

As shown in the following table, the PKW area is located at the beginning of the corresponding assembly objects 112, 113, 102, and 103.



PKW OUT Data

PKW OUT data requests from the DeviceNet scanner to the LTMR are mapped in assembly objects 101, 102, and 103.

To access registers, select one of the following function codes:
- R_REG_16 (0x25) to read 1 register
- R_REG_32 (0x26) to read 2 registers
- W_REG_16 (0x2A) to write 1 register
- W_REG_32 (0x2B) to write 2 registers

Word 1	**Word 2**			**Word 3**	**Word 4**
	MSB		**LSB**		
Register address	Toggle bit (bit 15)	Function bits (bits 8 to 14)	Not used (bits 0 to 7)	Data to write	
Register number	0/1	R_REG_16 Code 0x25	0x00	_	_
		R_REG_32 Code 0x26		_	_
		W_REG_16 Code 0x2A		Data to write in register	_
		W_REG_32 Code 0x2B		Data to write in register 1	Data to write in register 2

Any changes in the function code will trigger the handling of the request (unless Function code [bit 8 to 14] = 0x00).

NOTE: The highest bit of function code (bit 15) is a toggle bit. It is changed for each consecutive request.

This mechanism enables the request initiator to detect that a response is ready by polling bit 15 of the function code in word 2. When this bit in the OUT data becomes equal to the response emitted toggle bit in the IN data (when starting the request), then the response is ready.

PKW IN Data

PKW IN data response from the LTMR to the DeviceNet scanner are mapped in assembly objects 111, 112, and 113.

The LTMR echoes the same register address and function code or, eventually, an error code.

Word 1	Word 2			Word 3	Word 4
	MSB		LSB		
Register address	Toggle bit (bit 15)	Function bits (bits 8 to 14)	Not used (bits 0 to 7)	Data to write	
Same register number as in the request	Same as in the request	Error Code 0x4E	0x00	Error code	
		R_REG_16 Code 0x25		Data read in register	_
		R_REG_32 Code 0x26		Data read in register 1	Data read in register 2
		W_REG_16 Code 0x2A		_	_
		W_REG_32 Code 0x2B		_	_

If the initiator tries to write a TeSys T object or register to an unauthorized value, or tries to access an inaccessible register, an error code is returned (Function code = toggle bit + 0x4E). The exact error code can be found in words 3 and 4. The request is not accepted and the object/register remains at the old value.

To re-trigger exactly the same command, you need to:

1. Reset the Function code to 0x00,
2. Wait for the response frame with the function code equal to 0x00,
3. Reset it to its previous value.

This is useful for a limited master like an HMI.

Another way of re-triggering exactly the same command is to invert the toggle bit in the function code byte.

The response is valid when the toggle bit of the response is equal to the toggle bit written in the answer (this is a more efficient method, but it requires higher programming capabilities).

PKW Error Codes

Case of a write error:

Error Code	Error Name	Explanation
1	FGP_ERR_REQ_STACK_FULL	external request: sends back an error frame
3	FGP_ERR_REGISTER_NOT_FOUND	register not managed (or the request needs super user access rights)
4	FGP_ERR_ANSWER_DELAYED	external request: answer postponed
7	FGP_ERR_NOT_ALL_REGISTER_FOUND	one or both registers cannot be found
8	FGP_ERR_READ_ONLY	register not authorized to be written
10	FGP_ERR_VAL_1WORD_TOOHIGH	written value not in the range of the register (word value is too high)
11	FGP_ERR_VAL_1WORD_TOOLOW	written value not in the range of the register (word value is too low)
12	FGP_ERR_VAL_2BYTES_INF_TOOHIGH	written value not in the range of the register (MSB value is too high)
13	FGP_ERR_VAL_2BYTES_INF_TOOLOW	written value not in the range of the register (MSB value is too low)
16	FGP_ERR_VAL_INVALID	written value not a valid value
20	FGP_ERR_BAD_ANSWER	external request: sends back an error frame

Case of a read error:

Error Code	Error Name	Explanation
1	FGP_ERR_REQ_STACK_FULL	external request: sends back an error frame
3	FGP_ERR_REGISTER_NOT_FOUND	register not managed (or the request needs super user access rights)
4	FGP_ERR_ANSWER_DELAYED	external request: answer postponed
7	FGP_ERR_NOT_ALL_REGISTER_FOUND	one or both registers cannot be found

Object Dictionary

Overview

The DeviceNet protocol using object modeling. Object modeling organizes related data and procedures into one entity: the object.

An object is a collection of related services and attributes. Services are procedures an object performs. Attributes are characteristics of objects represented by values, which can vary. Typically, attributes provide status information or govern the operation of an object. The value associated with an attribute may or may not affect the behavior of an object. An object's behavior is an indication of how the object responds to particular events.

Objects within a class are called object instances. An object instance is the actual representation of a particular object within a class. Each instance of a class has the same set of attributes, but has its own set of attribute values, which makes each instance in the class unique. The Object Dictionary describes the attribute values of each object in the device profile.

LTMR Object Dictionary

The general breakdown of the LTMR DeviceNet brick object dictionary is the same for all DeviceNet devices:

Index	Object	Description
01h	Identity Object	Identifiers, such as device type, vendor ID, and serial number.
02h	Message Router Object	Provides a message connection point.
03h	DeviceNet Object	Maintains physical connection to the DeviceNet network; allocates and de-allocates the master/slave connection set.
04h	Assembly Object	Provides collection of other object's attributes (frequently used for I/O messaging).
05h	Connection Object	Allows explicit messaging to be conducted.
29h	Control Supervisor Object	Manages controller functions, operational states, and control.
2Ch	Overload Object	Implements overload behavior.
C6h	DeviceNet Interface Object	Enables I/O messaging data to be selected.
C5h	PKW: Periodic Registers Service Objects	Enables cyclic I/O messaging for manufacturer-specific registers.

These objects are described in detail in the following pages.

Identity Object

Description

This object, present in all DeviceNet products, provides identification of, and general information about, the device.

Class Attributes

Attribute ID	Access	Name	Data Type	Value	Description
1	Get	Revision	UInt	01	-

Instance Attributes

Attribute ID	Access	Name	Data Type	Value	Description
1	Get	Vendor ID	UInt	243	243 -> "Schneider Automation Inc."
2	Get	Device type	UInt	16h	Motor Starter
3	Get	Product code	UInt	Product identification depends on the configuration	Remote mode: ● 0x30: Without expansion module ● 0x31: With expansion module ● 0x32 to 0x3F: Reserved Local mode: ● 0x130: Without expansion module ● 0x131: With expansion module
4	Get	Revision	Struct. of: UInt UInt	Product configuration	Product version
5	Get	Status	Word	01	See the following table.
6	Get	Serial number	UDInt	01	Read from the controller during start-up in registers [70] to [74]: *Control Unit Serial Number*
7	Get	Product name	Struct. of: USInt String	"LTM1"	Read from the controller during start-up in registers [64] to [69]: *Control Unit Identification*

Bit	Definition	Values
0	Owned by Master (predefined Master/Slave connection)	Provided by the stack
1	*Reserved*	0
2	Configured	NOT(Control Unit In Configuration Mode [456.9])
3	*Reserved*	0
4, 5, 6, 7	Vendor Specific: 4: Warning 5: Trip (Fault) 6: Contactor state 7: Reverser contactor state	 [455.3] [455.4] [455.1] & [704=1] [455.1] & [704=2]
8	Minor recoverable fault	0
9	Minor unrecoverable fault	0
10	Major recoverable fault	1 ≤ [451] ≤ 15
11	Major unrecoverable fault	[451] ≤ 15

Class and Instance Service

Service Code	Service Name	Description
0E hex	Get_Attribute_Single	Read 1 attribute
05 hex	Reset	Product reset

Message Router Object

Description

The Message Router Object provides a messaging connection point through which a Client may address a service to any object class or instance in the physical device.

Class Attributes

Attribute ID	Access	Name	Data Type	Value	Description
1	Get	Revision	UInt	01	-

Instance Attributes

Attribute ID	Access	Name	Data Type	Value	Description
1	Get	Object list: ● Number ● Classes	UInt		List of supported objects Number of supported classes List of supported classes
2	Get	Number available	UInt		Maximum number of connections supported
3	Get	Number active	UInt		Number of active connections
4	Get	Active connections	Struct. of: UInt UInt		List of active connections

Class and Instance Service

Service Code	Service Name	Description
0E hex	Get_Attribute_Single	Read 1 attribute

DeviceNet Object

Overview

The DeviceNet Object is used to provide the configuration and status of a physical attachment to the DeviceNet network. A product can support only one DeviceNet Object per physical connection to the DeviceNet communication terminals.

Class Attributes

Attribute ID	Access	Name	Data Type	Value	Description
1	Get	Revision	UInt	002	-

Instance Attributes

Attribute ID	Access	Name	Data Type	Value	Description
1	Get	MAC ID	USInt	0 - 63	Read-only attribute
2	Get	Baud rate	USInt	0 - 2	0: 125 k 1: 250 k 2: 500 k Read-only attribute
3	Get/Set	BOI (Bus OFF Interrupt)	Bool	-	Upon Bus-Off interrupt: 0: Hold the CAN chip in its bus-OFF state. 1: Reset the CAN chip and continue communicating.
4	Get/Set	BusOFF counter	USInt	0 - 255	Number of times CAN was in bus-OFF state
5	Get	Allocation information	Byte - USInt	0 - 63	Allocation choice Master Address (255 not allocated)

Class Service

Service Code	Service Name	Description
0E hex	Get_Attribute_Single	Read 1 attribute

Instance Service

Service Code	Service Name	Description
0E hex	Get_Attribute_Single	Read 1 attribute
19 hex	Set_AttributesSingle	Write 1 attribute
0E hex	Allocate Master/Slave Connection Set	Requests the use of the predefined Master/Slave Connection Set
0E hex	Release Master/Slave Connection Set	Indicates that the specified connections within the predefined Master/Slave Connection Set are no longer desired. These Connections are to be released (deleted).

Assembly Object

Description

The Assembly Object binds attributes of multiple objects, which enables each object's data to be sent or received over a single connection. Assembly objects can be used to bind input data or output data. The terms "input" and "output" are defined from the network's point of view. An input sends (produces) data on the network, and an output receives (consumes) data from the network.

Only static assemblies are supported.

Class Attributes

Attribute ID	Access	Name	Data Type	Value	Description
1	Get	Revision	UInt	02	-
2	Get	Max instance	UInt	13	-

Instance Attributes

Attribute ID	Access	Name	Data Type	Value	Description
3	Get	Data	See assembly data description below.		

Class and Instance Service

Service Code	Service Name	Description
0E hex	Get_Attribute_Single	Read 1 attribute

Output Assembly Data

Instance 2: Basic Overload

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	*Reserved*	*Reserved*	*Reserved*	*Reserved*	*Reserved*	FaultReset	*Reserved*	*Reserved*

Instance 3: Basic Motor Starter

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	*Reserved*	*Reserved*	*Reserved*	*Reserved*	*Reserved*	FaultReset	*Reserved*	Run 1

Instance 4: Extended Contactor

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	*Reserved*	*Reserved*	*Reserved*	*Reserved*	*Reserved*	*Reserved*	Run 2	Run 1

Instance 5: Extended Motor Starter

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	*Reserved*	*Reserved*	*Reserved*	*Reserved*	*Reserved*	FaultReset	Run 2	Run 1

NOTE:

- FaultReset = Register 704.3
- Run2 = Register 704.1
- Run1 = Register 704.0

Instance 100: LTMR Control Registers

This assembly contains several control registers commonly used with an LTMR device.

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
path: 6C : 01 : 05 *(Register {704})*		path: 6C : 01 : 04 *(Register {703})*		path: 6C : 01 : 01 *(Register {700})*	
LSB (least significant bit)	MSB (most significant bit)	LSB	MSB	LSB	MSB

Instance 101: PKW Request Object

This assembly is vendor specific. It is used to implement the request object of PKW protocol.

Byte 7	Byte 6	Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0
For details, refer to PKW Objects *(see page 34)*							

Instance 102: PKW Request and Extended Motor Starter

This assembly is vendor specific.

Bytes 0 to 7	Byte 8	Byte 9
See Instance 101 above.	Reserved (value = 0)	See Instance 5 above.

Instance 103: PKW Request and LTMR Control Registers

This assembly is vendor specific.

Bytes 0 to 7	Byte 8 to 13
See Instance 101 above.	See Instance 100 above.

Input Assembly Data

Instance 50: Basic Overload

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	*Reserved*	*Reserved*	*Reserved*	*Reserved*	*Reserved*	*Reserved*	*Reserved*	Faulted/Trip

Instance 51: Extended Overload

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	*Reserved*	*Reserved*	*Reserved*	*Reserved*	*Reserved*	FaultReset	Warning	Faulted/Trip

Instance 52: Basic Motor Starter

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	*Reserved*	*Reserved*	*Reserved*	*Reserved*	*Reserved*	Running1	*Reserved*	Faulted/Trip

Instance 53: Extended Motor Starter 1

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	*Reserved*	*Reserved*	CntrlfromNet	Ready	*Reserved*	Running1	Warning	Faulted/Trip

Instance 54: Extended Motor Starter 2

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	***Reserved***	***Reserved***	CntrlfromNet	Ready	Running2	Running1	Warning	Faulted/Trip

NOTE:

- CntrlfromNet = NOT (Register 455.14)
- Ready = Register 455.0
- Running2 = (Register 455.7) AND (Register 704.1)
- Running1 = (Register 455.7) AND (Register 704.0)
- Warning = Register 455.3
- Fault/Trip = (Register 455.2) OR (Register 455.4)

Instance 110: LTMR Monitoring Registers (with dynamic configuration)

This assembly contains several monitoring registers commonly used with an LTMR device. You can choose registers by setting attributes 5-8 of the DeviceNet interface object. For more details, refer to DeviceNet Interface Object *(see page 50)*.

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Register pointed using path: C6 : 01 : 05		Register pointed using path: C6: 01 : 06		Register pointed using path: C6 : 01 : 07		Register pointed using path: C6 : 01 : 08	
LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB

Instance 111: PKW Response Object

This assembly is vendor specific. It is used to implement the response object of PKW protocol.

Byte 7	Byte 6	Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0
For details, refer to PKW Objects *(see page 34)*							

Instance 112: PKW Request and Extended Motor Starter

This assembly is vendor specific.

Bytes 0 to 7	Byte 8	Byte 9
See Instance 111 above.	Reserved (value = 0)	See Instance 54 above.

Instance 113: PKW Request and LTMR Monitoring Registers

This assembly is vendor specific.

Bytes 0 to 7	Byte 8 to 15
See Instance 111 above.	See Instance 110 above.

Connection Object

Description

The Connection Object provides for and manages the run-time exchange of messages.

Class Attributes

Attribute ID	Access	Name	Data Type	Value	Description
1	Get	Revision	UInt	01	-

Instance 1 Attributes: Explicit Message Instance

Attribute ID	Access	Name	Data Type	Value	Description
1	Get	State	USInt	-	0: Non-existent 3: Established 5: Deferred Delete
2	Get	Instance_type	USInt	0	Explicit Message
3	Get	TransportClass_trigger	USInt	83h	Defines behavior of the connection
4	Get	Produced_connection_id	UInt	10xxxxxx011	xxxxxx = Node address
5	Get	Consumed_connection_id	UInt	10xxxxxx100	xxxxxx = Node address
6	Get	Initial_comm_characteristics	USInt	21h	Explicit messaging via Group 2
7	Get	Produced_connection_size	UInt	7	-
8	Get	Consumed_connection_size	UInt	7	-
9	Get/Set	Expected_packet_rate	UInt	2500	2.5 s (TimeOut)
12	Get/Set	Watchdog_timeout_action	UInt	1 or 3	1: Auto-Delete (Factory setting) 3: Deferred Delete
13	Get	Produced connection path length	UInt	0	-
14	Get	Produced connection path	UInt	Null	empty
15	Get	Consumed connection path length	UInt	0	-
16	Get	Consumed connection path	UInt	Null	empty

Instance 2 Attributes: Polled I/O Message Instance

Attribute ID	Access	Name	Data Type	Value	Description
1	Get	State	USInt	-	0: Non-existent 1: Configuring 3: Established 4: TimeOut
2	Get	Instance_type	USInt	1	I/O Message
3	Get	TransportClass_trigger	USInt	82h	Class 2
4	Get	Produced_connection_id	UInt	01111xxxxxx	xxxxxx = Node address
5	Get	Consumed_connection_id	UInt	10xxxxxx101	xxxxxx = Node address
6	Get	Initial_comm_characteristics	USInt	01h	Group1/Group 2
7	Get	Produced_connection_size	UInt	4	-
8	Get	Consumed_connection_size	UInt	4	-
9	Get/Set	Expected_packet_rate	UInt	0	-
12	Get/Set	Watchdog_timeout_action	USInt	0, 1, or 2	0: Transition to TimeOut 1: Auto-delete 2: Auto-reset
13	Get	Produced connection path length	UInt	-	-
14	Get/Set	Produced connection path	UInt	-	-
15	Get	Consumed connection path length	UInt	-	-

Attribute ID	Access	Name	Data Type	Value	Description
16	Get/Set	Consumed connection path	UInt	-	-
17	Get/Set	Production_inhibit_time	UInt	0	Minimum time between new data production

Instance 4 Attributes: Change-of-State/Cyclic Message Instance

Attribute ID	Access	Name	Data Type	Value	Description
1	Get	State	USInt	-	0: Non-existent 1: Configuring 3: Established 4: TimeOut
2	Get	Instance_type	USInt	1	I/O Message
3	Get	TransportClass_trigger	USInt	xx	-
4	Get	Produced_connection_id	UInt	01101xxxxxx	xxxxxx = Node address
5	Get	Consumed_connection_id	UInt	10xxxxxx101	xxxxxx = Node address
6	Get	Initial_comm_characteristics	USInt	01h	Group1/Group 2
7	Get	Produced_connection_size	UInt	4	-
8	Get	Consumed_connection_size	UInt	4	-
9	Get/Set	Expected_packet_rate	UInt	0	-
12	Get/Set	Watchdog_timeout_action	USInt	0, 1 or 2	0: Transition to TimeOut 1: Auto-delete 2: Auto-reset
13	Get	Produced connection path length	UInt	-	-
14	Get/Set	Produced connection path	UInt	-	-
15	Get	Consumed connection path length	UInt	-	-
16	Get/Set	Consumed connection path	UInt	-	-
17	Get/Set	Production_inhibit_time	UInt	0	Not defined

Class Service

Service Code	Service Name	Description
08 hex	Create	Used to instantiate a Connection Object
0E hex	Get_Attribute_Single	Read 1 attribute

Instance Service

Service Code	Service Name	Description
0E hex	Get_Attribute_Single	Read 1 attribute
10 hex	Set_Attribute_Single	Write 1 attribute
05 hex	Reset	Reset Inactivity/Watchdog timer

Control Supervisor Object

Description

This object models all the management functions for devices within the "Hierarchy of Motor Control Devices".

Class Attributes

Attribute ID	Access	Name	Data Type	Value	Description
1	Get	Revision	UInt	02	-
2	Get	Max instance	UInt	1	-

Instance Attributes

Attribute ID	Access	Name	Data Type	Description
3	Get/Set	Run Fwd	Bool	704.0
4	Get	Run Rev	Bool	704.1
6	Get	State	USInt	0 = Vendor Specific 1 = Startup 2 = Not_Ready 3 = Ready 4 = Enabled 5 = Stopping 6 = Fault_Stop 7 = Faulted
7	Get	Running Fwd	Bool	455.7 AND 704.0
8	Get	Running Rev	Bool	455.7 AND 704.1
9	Get	Ready	Bool	455.0
10	Get	Faulted	Bool	455.2
11	Get	Warning	Bool	455.3
12	Get/Set	FaultRst	Bool	704.3 = 0 ->1 (rising edge)
13	Get	FaultCode	UInt	451
14	Get	WarnCode	UInt	460
15	Get	CtrlFromNet	Bool	NOT(455.14)
16	Get/Set	DNFaultMode	UInt	Action on network loss: 0 = Fault + Stop ' 682 = 2 1 = Ignore ' 682 = 0 2 = Frozen ' 682 = 1 3 = Inchange ' 682 = 3 4 = Force FW ' 682 = 4 5 = Force RV ' 682 = 5
17	Get/Set	ForceFault/Trip	Bool	704.12

Class Service

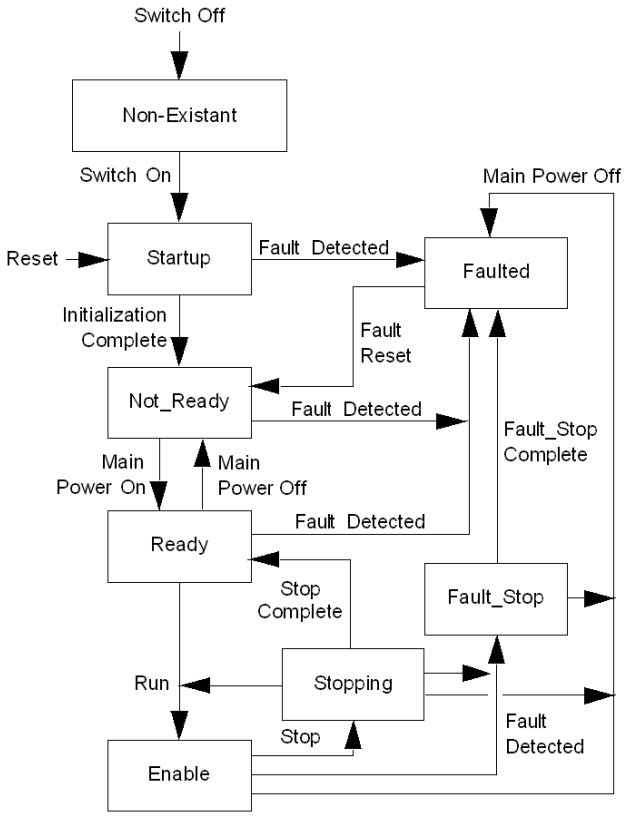
Service Code	Service Name	Description
0E hex	Get_Attribute_Single	Read 1 attribute

Instance Service

Service Code	Service Name	Description
0E hex	Get_Attribute_Single	Read 1 attribute
10 hex	Set_Attribute_Single	Write 1 attribute
05 hex	Reset	Reset Inactivity/Watchdog timer

Control Supervisor State Event

The following diagram shows the control supervisor state event matrix:



The following table describes the run/stop event matrix:

Event	State (N/A = No action)							
	Non-exist	Startup	Not_Ready	Ready	Enabled	Stopping	Fault-Stop	Faulted
Switch OFF	N/A	Transition to Non-exist	Transition to Non-exist	Transition to Non-exist	Transition to Non-exist	Transition to Non-exist	Transition to Non-exist	Transition to Non-exist
Switch ON	Transition to Startup	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Initialization Complete	N/A	Transition to Not_Ready	N/A	N/A	N/A	N/A	N/A	N/A
Main Power ON	N/A	N/A	Transition to Ready	N/A	N/A	N/A	N/A	N/A
Run	N/A	N/A	N/A	Transition to Enable	N/A	Transition to Enable	N/A	N/A
Stop	N/A	N/A	N/A	N/A	Transition to Stopping	N/A	N/A	N/A
Stop Complete	N/A	N/A	N/A	N/A	N/A	Transition to Ready	N/A	N/A
Reset	N/A	N/A	Transition to Startup	Transition to Startup	Transition to Startup	Transition to Startup	Transition to Startup	Transition to Startup
Main Power OFF	N/A	N/A	N/A	Transition to Not_Ready	Transition to Faulted	Transition to Faulted	Transition to Faulted	N/A
Fault Detected	N/A	Transition to Faulted	Transition to Faulted	Transition to Faulted	Transition to Fault_Stop	Transition to Fault_Stop	N/A	N/A
Fault_Stop Complete	N/A	N/A	N/A	N/A	N/A	N/A	Transition to Faulted	
Fault Reset	N/A	N/A	N/A	N/A	N/A	N/A	N/A	Transition to Not_Ready

Attribute 5 (NetCtrl) is used to request that Run Stop events be controlled from the network. You may inhibit these events, however, if you do not wish to allow Run Stop control from the network under certain circumstances, or if your application does not permit it. Only when attribute 15 (CtrlFromNet) is set to 1 by the device in response to a NetCtrl request, is Run Stop control actually enabled from the network.

If attribute 15 (CtrlFromNet) is 1, then the events Run and Stop are triggered by a combination of the Run1 and Run2 attributes, as shown in the following table. Note that Run1 and Run2 have different contexts for different device types.

The following table shows the Run1 and Run2 contexts for the devices within the motor control hierarchy:

	Drives and Servos
Run1	RunFwd
Run2	RunRev

If CtrlFromNet is 0, Run and Stop events must be controlled using local input(s) provided by the vendor.

Run1	**Run2**	**Trigger Event**	**Run Type**
0	0	Stop	N/A
0 -> 1	0	Run	Run1
0	0 -> 1	Run	Run2
0 -> 1	0 -> 1	No action	N/A
1	1	No action	N/A
1 -> 0	1	Run	Run2
1	1 -> 0	Run	Run1

NOTE: Local stop and run signals could override or be interlocked with the run/stop control through DeviceNet.

Overload Object

Description

This object models all the functions specific to an AC motor overload protection device.

Class Attributes

Attribute ID	Access	Name	Data Type	Value	Description
1	Get	Revision	UInt	01	-
2	Get	Max instance	UInt	1	-

Instance Attributes

Attribute ID	Access	Name	Data Type	Value	Description
1	Get	NumAttr	UInt		Number of Attributes Supported
3	Set/Get	TripFLCSet	UInt	[652]	% of FLC max
4	Set/Get	TripClass	USInt	[606]	Trip Class Setting (0 to 200)
5	Get	AvgCurrent	Int	65535x[501]+[500]/10	0.1 A
6	Get	%PhImbal	USInt	[471]	% Phase Imbalance
7	Get	%Thermal	USInt	[465]	% Thermal Capacity
8	Get	IL1 Current	Int	65535x[503]+[504]/10	0.1 A
9	Get	IL2 Current	Int	65535x[505]+[506]/10	0.1 A
10	Get	IL3 Current	Int	65535x[507]+[506]/10	0.1 A
11	Get	Ground Current	Int	65535x[509]+[508]/10	0.1 A
101	Get	IL1 Current	Int	Idem Att. 8	0.1 A
102	Get	IL2 Current	Int	Idem Att. 9	0.1 A
103	Get	IL3 Current	Int	Idem Att. 10	0.1 A
104	Get	Ground Current	Int	Idem Att. 11	0.1 A
105	Get	IL1 Current Ratio	UInt	[467]	% of FLC
106	Get	IL2 Current Ratio	UInt	[468]	% of FLC
107	Get	IL3 Current Ratio	UInt	[469]	% of FLC
108	Get	IAV Average Current Ratio	UInt	[466]	% of FLC
109	Get	Thermal Capacity Level	UInt	[465]	%TripLevel
110	Get	Ground Current	Int	[Idem Att. 11	0.1 A
111	Get	Current phase imbalance	UInt	[471]	% Imbalance
112	Get	Time to trip	UInt	[511]	Seconds
113	Get/Set	Time to Reset	UInt	[450]	Seconds
127	Get/Set	Single/Three Ph	Bool	If [601.14]=1, return 0 If [601.13]=1, return 1	0 = Single phase 1 = Three phases
128	Get/Set	FLC Setting	UInt	[652]	Seconds
129	Get/Set	Load Class	UInt	[606]	Seconds
132	Get/Set	Thermal Warn Level	UInt	[609]	%TripLevel
133	Get/Set	PL Inhibit Time	USInt	[613]	Seconds
134	Get/Set	PL Trip Delay	USInt	[614]	Seconds
136	Get/Set	GF Trip Delay	USInt	[610]	0.1...25.0 S
137	Get/Set	GF Trip Level	USInt	[611]	1.0...5.0 A
138	Get/Set	GF Warn Level	USInt	[612]	1.0...5.0 A
139	Get/Set	Stall Enabled Time	USInt	[623]	0...250 S
140	Get/Set	Stall Trip Level	UInt	[624]	100...600
142	Get/Set	Jam Trip Delay	USInt	[617]	0.1...25.0 S
143	Get/Set	Jam Trip Level	UInt	[618]	0...600 % FLC
144	Get/Set	Jam Warn Level	UInt	[619]	0...600 % FLC

Attribute ID	Access	Name	Data Type	Value	Description
146	Get/Set	UL Trip Delay	USInt	[620]	0.1...25.0 S
147	Get/Set	UL Trip Level	USInt	[621]	10...100 % FLC
148	Get/Set	UL Warn Level	USInt	[622]	10...100 % FLC
149	Get/Set	CI Inhibit Time	USInt	[613]	0...250 S
150	Get/Set	CI Trip Delay	USInt	[614]	0.1...25.0 S
151	Get/Set	CI Trip Level	USInt	[615]	10...100 % FLC
152	Get/Set	CI Warn Level	USInt	[616]	10...100 % FLC
178	Get	CT Ratio	USInt	$95 = \frac{[628] \times [630]}{[629]}$	

NOTE: In the table above:
- PL = Current Phase Loss
- GF = Ground Fault
- Stall = Long Start
- UL = Underload
- CI = Current Phase Imbalance

Class Service

Service Code	Service Name	Description
0E hex	Get_Attribute_Single	Read 1 attribute

Instance Service

Service Code	Service Name	Description
0E hex	Get_Attribute_Single	Read 1 attribute
10 hex	Set_Attribute_Single	Write 1 attribute

DeviceNet Interface Object

Description

This object enables you to select the data that will be exchanged on the network through I/O messaging. A single instance (instance 1) of the DeviceNet Interface Object is supported.

Instance Attributes

The following instance attributes are supported:

Attribute ID	Access	Name	Data Type	Value
1	Set/Get	Poll-produced assembly instance	Byte (0...7)	0: Instance 50: Basic Overload 1: Instance 51: Extended Overload 2: Instance 52: Basic Motor Starter 3: Instance 53: Extended Motor Starter 1 (EMS1) 4: Instance 54: Extended Motor Starter 2 (EMS2) (Factory setting) 5: Instance 110: LTM1 Monitoring registers 6: Instance 111: PKW response object 7: Instance 112: PKW response + EMS2 8:Instance 113: PKW response + LTM1 monitoring
2	Set/Get	Poll-consumed assembly instance	Byte (0...7)	0: Instance 2: Basic Overload 1: Instance 3: Basic Motor Starter 2: Instance 4: Extended Contactor 3: Instance 5: Extended Motor Starter (EMS) 4: Instance 5: Extended Motor Starter (EMS) (Factory setting)(1) 5: Instance 100: LTM1 control registers 6: Instance 101: PKW Request object 7: Instance 102: PKW Request + EMS 8: Instance 103: PKW Request + LTM1 control
3	Set/Get	COS-produced assembly instance	Byte (0...7)	0: Instance 50: Basic Overload 1: Instance 51: Extended Overload 2: Instance 52: Basic Motor Starter 3: Instance 53: Extended Motor Starter 1 (EMS1) 4: Instance 54: Extended Motor Starter 2 (EMS2) (Factory setting) 5: Instance 110: LTM1 Monitoring registers 6: Instance 111: PKW response object 7: Instance 112: PKW response + EMS2 8: Instance 113: PKW response + LTM1 monitoring
4	Set/Get	AutoBaud enable	Bool	0: AutoBaud disable (Factory setting) 1: AutoBaud enable(2)
5	Set/Get	LTMR monitoring Word 0	UInt	Register of word 0 (Factory setting: 455)(3)
6	Set/Get	LTMR monitoring Word 1	UInt	Register of word 1 (Factory setting: 456)(3)
7	Set/Get	LTMR monitoring Word 2	UInt	Register of word 2 (Factory setting: 457)(3)
8	Set/Get	LTMR monitoring Word 3	UInt	Register of word 3 (Factory setting: 459)(3)

(1) The Extended Motor Starter (EMS) is repeated twice (value 3 and 4) in the Poll-consumed assembly list of values. This is done to be consistent with values 3 and 4 of the Poll-produced assembly list of values.

(2) The AutoBaud enable value (attribute 4) is read at power-up only. When this bit is cleared (when disabling auto-baud), the current baudrate is written to the register Network Port Baud Rate Setting [695]. Network Port Baud Rate Setting has priority over this bit in the event of inconsistency (checked at power up). In this case, the AutoBaud enable value is set according to the register Network Port Baud Rate Setting at power up.

(3) The configuration of the LTMR monitoring assembly (attributes 5 to 8) is read when the device is allocated to a master, i.e. when the device is connected. Any change that occurs after allocation will not take effect before the connection release/reallocation phases. Permitted values for these 4 sattributes are 0 to 19999.

Instance Service

Service Code	Service Name	Description
0E hex	Get_Attribute_Single	Read 1 attribute
10 hex	Set_Attribute_Single	Write 1 attribute

Register Map - Organization of Communication Variables

Introduction

Communication variables are listed in tables. They belong to groups (identification, statistics, monitoring,...). They are associated with an LTMR Controller, which may or may not have an LTME expansion module attached.

Communication Variable Groups

Communication variables are grouped according to the following criteria:

Variable Groups	Register	DeviceNet Addresses
Identification variables	00 to 99	64 : 01 : 32 to 64 : 01 : 62
Statistics variables	100 to 449	65 : 01 : 01 to 67 : 01 : 82
Monitoring variables	450 to 539	68 : 01 : 01 to 68 : 01 : 54
Configuration variables	540 to 699	69 : 01 : 01 to 6B : 01 : 32
Command variables	700 to 799	6C : 01 : 01 to 6C : 01 : 0F
Custom Logic variables	1200 to 1399	71 : 01 : 01 to 71 : 01 : 0A

Table Structure

Communication variables are listed in 5-column tables:

Column 1 Register number (decimal)	Column 2 DeviceNet address (class : instance : attribute)	Column 3 Variable type: integer, word, word[n], DT_type *(see page 53)*	Column 4 Variable name and access via Read only or Read/Write requests	Column 5 Note: code for additional information

Note

The Note column gives a code for additional information.

Variables without a code are available for all hardware configurations, and without functional restrictions.

The code can be:

- numerical (1 to 9), for specific hardware combinations
- alphabetical (A to Z), for specific system behaviors.

If the Note is...	Then the Variable is...
1	available for the LTMR + LTMEV40 combination
2	always available but with a value equal to 0 if no LTMEV40 is connected
3-9	Not used

If the Note is...	Then...
A	the variable can be written only when the motor is OFF
B	the variable can be written only in configuration mode
C	the variable can be written only with no fault
D-Z	the variable is available for future exceptions

Unused Addresses

Unused addresses fall into three categories:

- **Not significant**, in Read only tables, means that you should ignore the value read, whether equal to 0 or not.
- **Reserved**, in Read/Write tables, means that you must write 0 in these variables.
- **Forbidden**, means that read or write requests are rejected, that these addresses are not accessible at all.

Data Formats

Overview

The data format of a communication variable can be integer, Word, or Word[n], as described below. For more information about a variable size and format, refer to Data Types *(see page 53)*.

Integer (Int, UInt, DInt, IDInt)

Integers fall into the following categories:

- **Int**: signed integer using one register (16 bits)
- **UInt**: unsigned integer using one register (16 bits)
- **DInt**: signed double integer using two registers (32 bits)
- **UDInt**: unsigned double integer using two registers (32 bits)

For all integer-type variables, the variable name is completed with its unit or format, if necessary.

Example:

Address 474, **UInt**, Frequency (x 0.01 Hz).

Word

Word: Set of 16 bits, where each bit or group of bits represents command, monitoring or configuration data.

Example:

Address 455, **Word**, System Status Register 1.

bit 0	System ready
bit 1	System ON
bit 2	System fault
bit 3	System warning
bit 4	System tripped
bit 5	Fault reset authorized
bit 6	*(Not significant)*
bit 7	Motor running
bits 8-13	Motor average current ratio
bit 14	In remote
bit 15	Motor starting (in progress)

Word[n]

Word[n]: Data encoded on contiguous registers.

Examples:

Addresses 64 to 69, **Word[6]**, Controller Commercial Reference (DT_CommercialReference *(see page 53)*).

Addresses 655 to 658, **Word[4]**, (DT_DateTime *(see page 54)*).

Data Types

Overview

Data types are specific variable formats which are used to complement the description of internal formats (for instance, in case of a structure or of an enumeration). The generic format of data types is DT_xxx.

List of Data Types

Here is the list of the most commonly used data types:

- DT_ACInputSetting
- DT_CommercialReference
- DT_DateTime
- DT_ExtBaudRate
- DT_ExtParity
- DT_FaultCode
- DT_FirmwareVersion
- DT_Language5
- DT_OutputFallbackStrategy
- DT_PhaseNumber
- DT_ResetMode
- DT_WarningCode

These data types are described in the following tables.

DT_ACInputSetting

DT_ACInputSetting format is an **enumeration** that improves AC input detection:

Value	Description
0	None (factory setting)
1	< 170 V 50 Hz
2	< 170 V 60 Hz
3	> 170 V 50 Hz
4	> 170 V 60 Hz

DT_CommercialReference

DT_CommercialReference format is **Word[6]** and indicates a Commercial Reference:

Register	MSB	LSB
Register N	character 1	character 2
Register N+1	character 3	character 4
Register N+2	character 5	character 6
Register N+3	character 7	character 8
Register N+4	character 9	character 10
Register N+5	character 11	character 12

Example:

Addresses 64 to 69, **Word[6]**, Controller Commercial Reference.

If Controller Commercial Reference = LTMR:

Register	MSB	LSB
64	L	T
65	M	(space)
66	R	
67		
68		
69		

DT_DateTime

DT_DateTime format is **Word[4]** and indicates Date and Time:

Register	Bits 12-15	Bits 8-11	Bits 4-7	Bits 0-3
Register N	S	S	0	0
Register N+1	H	H	m	m
Register N+2	M	M	D	D
Register N+3	Y	Y	Y	Y

Where:

- S = second
  The format is 2 BCD digits.
  The value range is [00...59] in BCD.
- 0 = unused
- H = hour
  The format is 2 BCD digits.
  The value range is [00...23] in BCD.
- m = minute
  The format is 2 BCD digits.
  The value range is [00...59] in BCD.
- M = month
  The format is 2 BCD digits.
  The value range is [01...12] in BCD.
- D = day
  The format is 2 BCD digits.
  The value range is (in BCD):
  [01-31] for months 01, 03, 05, 07, 08, 10, 12
  [01-30] for months 04, 06, 09, 11
  [01-29] for month 02 in a leap year
  [01-28] for month 02 in a non-leap year.
- Y = year
  The format is 4 BCD digits.
  The value range is [2006...2099] in BCD.

Data entry format and value range are:

Data Entry Format	DT#YYYY-MM-DD-HH:mm:ss	
Minimum value	DT#2006-01-01:00:00:00	January 1, 2006
Maximum value	DT#2099-12-31-23:59:59	December 31, 2099
Note: If you give values outside the limits, the system will return an error.		

Example:

Addresses 655 to 658, **Word[4]**, Date and Time setting.

If date is September 4, 2008 at 7 a.m., 50 minutes and 32 seconds:

Register	15 12	11 8	7 4	3 0
655	3	2	0	0
656	0	7	5	0
657	0	9	0	4
658	2	0	0	8

With data entry format: DT#2008-09-04-07:50:32.

DT_ExtBaudRate

DT_ExtbaudRate depends on the bus used:

DT_ModbusExtBaudRate format is an **enumeration** of possible baud rates with Modbus network:

Value	Description
1200	1200 Baud
2400	2400 Baud
4800	4800 Baud
9600	9600 Baud
19200	19,200 Baud
65535	Autodetection (factory setting)

DT_ProfibusExtBaudRate format is an **enumeration** of possible baud rates with PROFIBUS DP network:

Value	Description
65535	Autobaud (factory setting)

DT_DeviceNetExtBaudRate format is an **enumeration** of possible baud rates with DeviceNet network:

Value	Description
0	125 kBaud
1	250 kBaud
2	500 kBaud
3	Autobaud (factory setting)

DT_CANopenExtBaudRate format is an **enumeration** of possible baud rates with CANopen network:

Value	Description
0	10 kBaud
1	20 kBaud
2	50 kBaud
3	125 kBaud
4	250 kBaud (factory setting)
5	500 kBaud
6	800 kBaud
7	1000 kBaud
8	Autobaud
9	Factory setting

DT_ExtParity

DT_ExtParity depends on the bus used:

DT_ModbusExtParity format is an **enumeration** of possible parities with Modbus network:

Value	Description
0	None
1	Even
2	Odd

DT_FaultCode

DT_FaultCode format is an **enumeration** of fault codes:

Fault Code	Description
0	No error
3	Ground current
4	Thermal overload
5	Long start

Fault Code	Description
6	Jam
7	Current phase imbalance
8	Undercurrent
10	Test
11	HMI port error
12	HMI port communication loss
13	Network port internal error
16	External fault
18	ON-OFF diagnostic
19	Wiring diagnostic
20	Overcurrent
21	Current phase loss
22	Current phase reversal
23	Motor temp sensor
24	Voltage phase imbalance
25	Voltage phase loss
26	Voltage phase reversal
27	Undervoltage
28	Overvoltage
29	Underpower
30	Overpower
31	Under power factor
32	Over power factor
33	LTME configuration
34	Temperature sensor short-circuit
35	Temperature sensor open-circuit
36	CT reversal
37	Out of boundary CT ratio
46	Start check
47	Run checkback
48	Stop check
49	Stop checkback
51	Controller internal temperature error
55	Controller internal error (Stack overflow)
56	Controller internal error (RAM error)
57	Controller internal error (RAM checksum error)
58	Controller internal error (Hardware watchdog fault)
60	L2 current detected in single-phase mode
64	Non volatile memory error
65	Expansion module communication error
66	Stuck reset button
67	Logic function error
100-104	Network port internal error
109	Network port comm error
111	Faulty device replacement fault
555	Network port configuration error

DT_FirmwareVersion

DT_FirmwareVersion format is an **XY000 array** that describes a firmware revision:
- X = major revision
- Y = minor revision.

Example:

Address 76, **UInt**, Controller firmware version.

DT_Language5

DT_Language5 format is an **enumeration** used for language display:

Language Code	Description
1	English (factory setting)
2	Français
4	Español
8	Deutsch
16	Italiano

Example:

Address 650, **Word**, HMI language.

DT_OutputFallbackStrategy

DT_OutputFallbackStrategy format is an **enumeration** of motor output states when loosing communication.

Value	Description	Motor Modes
0	Hold LO1 LO2	For all modes
1	Run	For two step mode only
2	LO1, LO2 Off	For all modes
3	LO1, LO2 On	Only for overload, independent and custom operating modes
4	LO1 On	For all modes except two step
5	LO2 On	For all modes except two step

DT_PhaseNumber

DT_PhaseNumber format is an **enumeration**, with only 1 bit activated:

Value	Description
1	1 phase
2	3 phases

DT_ResetMode

DT_ResetMode format is an **enumeration** of possible modes for thermal fault reset:

Value	Description
1	Manual or HMI
2	Remote by network
4	Automatic

DT_WarningCode

DT_WarningCode format is an **enumeration** of warning codes:

Warning Code	Description
0	No warning
3	Ground current
4	Thermal overload
5	Long start
6	Jam

Warning Code	Description
7	Current phase imbalance
8	Undercurrent
10	HMI port
11	LTMR internal temperature
18	Diagnostic
19	Wiring
20	Overcurrent
21	Current phase loss
23	Motor temp sensor
24	Voltage phase imbalance
25	Voltage phase loss
27	Undervoltage
28	Overvoltage
29	Underpower
30	Overpower
31	Under power factor
32	Over power factor
33	LTME configuration
46	Start check
47	Run checkback
48	Stop check
49	Stop checkback
109	Network port comm loss
555	Network port configuration

Identification Variables

Identification Variables

Identification Variables

Identification variables are described in the following table:

Register	DeviceNet Address	Variable Type	Read-only Variables	Note *(see page 51)*
0-34	64 : 01 : 03 - 64 : 01 : 23		*(Not significant)*	
35-40	64 : 01 : 24 - 64 : 01 : 29	Word[6]	Expansion commercial reference *(see page 53)*	1
41-45	64 : 01 :2A - 64 : 01 : 2E	Word[5]	Expansion serial number	1
46	64 : 01 : 2F	UInt	Expansion ID code	1
47	64 : 01 : 30	UInt	Expansion firmware version *(see page 57)*	1
48	64 : 01 : 31	UInt	Expansion compatibility code	1
49-60	64 : 01 : 32 - 64 : 01 : 3D		*(Not significant)*	
61	64 : 01 : 3E	UInt	Network port ID code	
62	64 : 01 : 3F	UInt	Network port firmware version *(see page 57)*	
63	64 : 01 : 40	UInt	Network port compatibility code	
64-69	64 : 01 : 41 - 64 : 01 : 46	Word[6]	Controller commercial reference *(see page 53)*	
70-74	64 : 01 : 47 - 64 : 01 : 4B	Word[5]	Controller serial number	
75	64 : 01 :4 C	UInt	Controller ID code	
76	64 : 01 : 4D	UInt	Controller firmware version *(see page 57)*	
77	64 : 01 :4E	UInt	Controller compatibility code	
78	64 : 01 : 4F	UInt	Current scale ratio (0.1 %)	
79	64 : 01 : 50	UInt	Current sensor max	
80	64 : 01 : 51		*(Not significant)*	
81	64 : 01 : 52	UInt	Current range max (x 0.1 A)	
82-94	64 : 01 : 53 - 64 : 01 : 5D		*(Not significant)*	
95	64 : 01 : 60	UInt	Load CT ratio (x 0.1 A)	
96	64 : 01 : 61	UInt	Full load current max (maximum FLC range, *FLC = Full Load Current*) (x 0.1 A)	
97-99	64 : 01 : 62 - 64 : 01 : 64		*(Forbidden)*	

Statistics Variables

Statistics Overview

Statistics variables are grouped according to the following criteria. Trip statistics are described in a main table and in an extension table.

Statistics Variable Groups	Register	DeviceNet Addresses
Global statistics	100 to 121	65 : 1 : 1 to 65 : 1 : 16
LTM monitoring statistics	122 to 149	65 : 1 : 17 to 65 : 1 : 32
Last trip statistics and extension	150 to 179 300 to 309	66 : 1 : 1 to 66 : 1 : 1E 67 : 1 : 1 to 67 : 1 : 0A
Trip n-1 statistics and extension	180 to 209 330 to 339	66 : 1 : 1F to 66 : 1 : 3C 67 : 1 : 1F to 67 : 1 : 28
Trip n-2 statistics and extension	210 to 239 360 to 369	66 : 1 : 3D to 66 : 1 : 5A 67 : 1 : 3D to 67 : 1 : 46
Trip n-3 statistics and extension	240 to 269 390 to 399	66 : 1 : 5B to 66 : 1 : 78 67 : 1 : 5B to 67 : 1 : 64
Trip n-4 statistics and extension	270 to 299 420 to 429	66 : 1 : 79 to 66 : 1 : 96 67 : 1 : 79 to 67 : 1 : 82

Global Statistics

The global statistics are described in the following table:

Register	DeviceNet Address	Variable Type	Read-only Variables	Note *(see page 51)*
100-101	65 : 01 : 01 - 65 : 01 : 02		*(Not significant)*	
102	65 : 01 : 03	UInt	Ground current faults count	
103	65 : 01 : 04	UInt	Thermal overload faults count	
104	65 : 01 : 05	UInt	Long start faults count	
105	65 : 01 : 06	UInt	Jam faults count	
106	65 : 01 : 07	UInt	Current phase imbalance faults count	
107	65 : 01 : 08	UInt	Undercurrent faults count	
109	65 : 01 : 0A	UInt	HMI port faults count	
110	65 : 01 : 0B	UInt	Controller internal faults count	
111	65 : 01 : 0C	UInt	Internal port faults count	
112	65 : 01 : 0D		*(Not significant)*	
113	65 : 01 : 0E	UInt	Network port config faults count	
114	65 : 01 : 0F	UInt	Network port faults count	
115	65 : 01 : 10	UInt	Auto-resets count	
116	65 : 01 : 11	UInt	Thermal overload warnings count	
117-118	65 : 01 : 12 - 65 : 01 : 13	UDInt	Motor starts count	
119-120	65 : 01 : 14 - 65 : 01 : 15	UDInt	Operating time (s)	
121	65 : 01 : 16	Int	Controller internal temperature max (°C)	

LTM Monitoring Statistics

The LTM monitoring statistics are described in the following table:

Register	DeviceNet Address	Variable Type	Read-only Variables	Note *(see page 51)*
122	65 : 01 : 17	UInt	Faults count	
123	65 : 01 : 18	UInt	Warnings count	
124-125	65 : 01 : 14 - 65 : 01 : 1A	UDInt	Motor LO1 closings count	
126-127	65 : 01 : 1B - 65 : 01 : 1C	UDInt	Motor LO2 closings count	
128	65 : 01 : 1C	UInt	Diagnostic faults count	
129	65 : 01 : 1E		*(Reserved)*	
130	65 : 01 : 1F	UInt	Overcurrent faults count	
131	65 : 01 : 20	UInt	Current phase loss faults count	
132	65 : 01 : 21	UInt	Motor temperature sensor faults count	
133	65 : 01 : 22	UInt	Voltage phase imbalance faults count	1
134	65 : 01 : 23	UInt	Voltage phase loss faults count	1
135	65 : 01 : 24	UInt	Wiring faults count	1
136	65 : 01 : 25	UInt	Undervoltage faults count	1
137	65 : 01 : 26	UInt	Overvoltage faults count	1
138	65 : 01 : 27	UInt	Underpower faults count	1
139	65 : 01 : 28	UInt	Overpower faults count	1
140	65 : 01 : 29	UInt	Under power factor faults count	1
141	65 : 01 : 2A	UInt	Over power factor faults count	1
142	65 : 01 : 2B	UInt	Load sheddings count	1
143-144	65 : 01 : 2C - 65 : 01 : 2D	UDInt	Active power consumption (x 0.1kWh)	1
145-146	65 : 01 : 2E - 65 : 01 : 2F	UDInt	Reactive power consumption (x 0.1 kVARh)	1
147	65 : 01 : 30	UInt	Auto restart immediate count	
148	65 : 01 : 31	UInt	Auto restart delayed count	
149	65 : 01 : 32	UInt	Auto restart manual count	

Last Fault (n-0) Statistics

The last fault statistics are completed by variables at addresses 300 to 309.

Register	DeviceNet Address	Variable Type	Read-only Variables	Note *(see page 51)*
150	66 : 01 : 01	UInt	Fault code n-0	
151	66 : 01 : 02	UInt	Motor full load current ratio n-0 (% FLC max)	
152	66 : 01 : 03	UInt	Thermal capacity level n-0 (% trip level)	
153	66 : 01 : 04	UInt	Average current ratio n-0 (% FLC)	
154	66 : 01 : 05	UInt	L1 current ratio n-0 (% FLC)	
155	66 : 01 : 06	UInt	L2 current ratio n-0 (% FLC)	
156	66 : 01 : 07	UInt	L3 current ratio n-0 (% FLC)	
157	66 : 01 : 08	UInt	Ground current ratio n-0 (x 0.1 % FLC min)	
158	66 : 01 : 09	UInt	Full load current max n-0 (x 0.1 A)	
159	66 : 01 : 0A	UInt	Current phase imbalance n-0 (%)	
160	66 : 01 : 0B	UInt	Frequency n-0 (x 0.1 Hz)	2
161	66 : 01 : 0C	UInt	Motor temperature sensor n-0 (x 0.1 Ω)	
162-165	65 : 01 : 2D - 65 : 01 : 10	Word[4]	Date and time n-0 *(see page 54)*	

Register	DeviceNet Address	Variable Type	Read-only Variables	Note *(see page 51)*
166	66 : 01 : 11	UInt	Average voltage n-0 (V)	1
167	66 : 01 : 12	UInt	L3-L1 voltage n-0 (V)	1
168	66 : 01 : 13	UInt	L1-L2 voltage n-0 (V)	1
169	66 : 01 : 14	UInt	L2-L3 voltage n-0 (V)	1
170	66 : 01 : 15	UInt	Voltage phase imbalance n-0 (%)	1
171	66 : 01 : 16	UInt	Active power n-0 (x 0.1 kWh)	1
172	66 : 01 : 17	UInt	Power factor n-0 (x 0.01)	1
173-179	66 : 01 : 18 - 66 : 01 : 1E		*(Not significant)*	

N-1 Fault Statistics

The n-1 fault statistics are completed by variables at addresses 330 to 339.

Register	DeviceNet Address	Variable Type	Read-only Variables	Note *(see page 51)*
180	66 : 01 : 1F	UInt	Fault code n-1	
181	66 : 01 : 20	UInt	Motor full load current ratio n-1 (% FLC max)	
182	66 : 01 : 21	UInt	Thermal capacity level n-1 (% trip level)	
183	66 : 01 : 22	UInt	Average current ratio n-1 (% FLC)	
184	66 : 01 : 23	UInt	L1 current ratio n-1 (% FLC)	
185	66 : 01 : 24	UInt	L2 current ratio n-1 (% FLC)	
186	66 : 01 : 25	UInt	L3 current ratio n-1 (% FLC)	
187	66 : 01 : 26	UInt	Ground current ratio n-1 (x 0.1 % FLC min)	
188	66 : 01 : 27	UInt	Full load current max n-1 (x 0.1 A)	
189	66 : 01 : 28	UInt	Current phase imbalance n-1 (%)	
190	66 : 01 : 29	UInt	Frequency n-1 (x 0.1 Hz)	2
191	66 : 01 : 2A	UInt	Motor temperature sensor n-1 (x 0.1 Ω)	
192-195	66 : 01 : 2B - 66 : 01 : 2E	Word[4]	Date and time n-1 *(see page 54)*	
196	66 : 01 : 2F	UInt	Average voltage n-1 (V)	1
197	66 : 01 : 30	UInt	L3-L1 voltage n-1 (V)	1
198	66 : 01 : 31	UInt	L1-L2 voltage n-1 (V)	1
199	66 : 01 : 32	UInt	L2-L3 voltage n-1 (V)	1
200	66 : 01 : 33	UInt	Voltage phase imbalance n-1 (%)	1
201	66 : 01 : 34	UInt	Active power n-1 (x 0.1 kWh)	1
202	66 : 01 : 35	UInt	Power factor n-1 (x 0.01)	1
203-209	66 : 01 : 36 - 66 : 01 : 3C		*(Not significant)*	

N-2 Fault Statistics

The n-2 fault statistics are completed by variables at addresses 360 to 369.

Register	DeviceNet Address	Variable Type	Read-only Variables	Note *(see page 51)*
210	66 : 01 : 3D	UInt	Fault code n-2	
211	66 : 01 : 3E	UInt	Motor full load current ratio n-2 (% FLC max)	
212	66 : 01 : 3F	UInt	Thermal capacity level n-2 (% trip level)	
213	66 : 01 : 40	UInt	Average current ratio n-2 (% FLC)	
214	66 : 01 : 41	UInt	L1 current ratio n-2 (% FLC)	
215	66 : 01 : 42	UInt	L2 current ratio n-2 (% FLC)	

Register	DeviceNet Address	Variable Type	Read-only Variables	Note *(see page 51)*
216	66 : 01 : 43	UInt	L3 current ratio n-2 (% FLC)	
217	66 : 01 : 44	UInt	Ground current ratio n-2 (x 0.1% FLC min)	
218	66 : 01 : 45	UInt	Full load current max n-2 (x 0.1 A)	
219	66 : 01 : 46	UInt	Current phase imbalance n-2 (%)	
220	66 : 01 : 47	UInt	Frequency n-2 (x 0.1 Hz)	2
221	66 : 01 : 48	UInt	Motor temperature sensor n-2 (x 0.1 Ω)	
222-225	66 : 01 : 49 - 66 : 01 : 4C	Word[4]	Date and time n-2 *(see page 54)*	
226	66 : 01 : 4D	UInt	Average voltage n-2 (V)	1
227	66 : 01 : 4E	UInt	L3-L1 voltage n-2 (V)	1
228	66 : 01 : 4F	UInt	L1-L2 voltage n-2 (V)	1
229	66 : 01 : 50	UInt	L2-L3 voltage n-2 (V)	1
230	66 : 01 : 51	UInt	Voltage phase imbalance n-2 (%)	1
231	66 : 01 : 52	UInt	Active power n-2 (x 0.1 kWh)	1
232	66 : 01 : 53	UInt	Power factor n-2 (x 0.01)	1
233-239	66 : 01 : 54 - 66 : 01 : 5A		*(Not significant)*	

N-3 Fault Statistics

The n-3 fault statistics are completed by variables at addresses 390 to 399.

Register	DeviceNet Address	Variable Type	Read-only Variables	Note *(see page 51)*
240	66 : 01 : 5B	UInt	Fault code n-3	
241	66 : 01 : 5C3	UInt	Motor full load current ratio n-3 (% FLC max)	
242	66 : 01 : 5D	UInt	Thermal capacity level n-3 (% trip level)	
243	66 : 01 : 5E	UInt	Average current ratio n-3 (% FLC)	
244	66 : 01 : 5F	UInt	L1 current ratio n-3 (% FLC)	
245	66 : 01 : 60	UInt	L2 current ratio n-3 (% FLC)	
246	66 : 01 : 61	UInt	L3 current ratio n-3 (% FLC)	
247	66 : 01 : 62	UInt	Ground current ratio n-3 (x 0.1 % FLC min)	
248	66 : 01 : 63	UInt	Full load current max n-3 (0.1 A)	
249	66 : 01 : 64	UInt	Current phase imbalance n-3 (%)	
250	66 : 01 : 65	UInt	Frequency n-3 (x 0.1 Hz)	2
251	66 : 01 : 66	UInt	Motor temperature sensor n-3 (x 0.1 Ω)	
252-255	66 : 01 : 67 - 66 : 01 : 6A	Word[4]	Date and time n-3 *(see page 54)*	
256	66 : 01 : 6B	UInt	Average voltage n-3 (V)	1
257	66 : 01 : 6C	UInt	L3-L1 voltage n-3 (V)	1
258	66 : 01 : 6D	UInt	L1-L2 voltage n-3 (V)	1
259	66 : 01 : 6E	UInt	L2-L3 voltage n-3 (V)	1
260	66 : 01 : 6F	UInt	Voltage phase imbalance n-3 (%)	1
261	66 : 01 : 70	UInt	Active power n-3 (x 0.1 kWh)	1
262	66 : 01 : 71	UInt	Power factor n-3 (x 0.01)	1
263-269	66 : 01 : 72 - 66 : 01 : 78		*(Not significant)*	

N-4 Fault Statistics

The n-4 fault statistics are completed by variables at addresses 420 to 429.

Register	DeviceNet Address	Variable Type	Read-only Variables	Note *(see page 51)*
270	66 : 01 : 79	UInt	Fault code n-4	
271	66 : 01 : 7A	UInt	Motor full load current ratio n-4 (% FLC max)	
272	66 : 01 : 7B	UInt	Thermal capacity level n-4 (% trip level)	
273	66 : 01 : 7C	UInt	Average current ratio n-4 (% FLC)	
274	66 : 01 : 7D	UInt	L1 current ratio n-4 (% FLC)	
275	66 : 01 : 7E	UInt	L2 current ratio n-4 (% FLC))	
276	66 : 01 : 7F	UInt	L3 current ratio n-4 (% FLC)	
277	66 : 01 : 80	UInt	Ground current ratio n-4 (x 0.1 % FLC min)	
278	66 : 01 : 81	UInt	Full load current max n-4 (x 0.1 A)	
279	66 : 01 : 82	UInt	Current phase imbalance n-4 (%)	
280	66 : 01 : 83	UInt	Frequency n-4 (x 0.1 Hz)	2
281	66 : 01 : 84	UInt	Motor temperature sensor n-4 (x 0.1 Ω)	
282-285	66 : 01 : 85 - 66 : 01 : 88	Word[4]	Date and time n-4 *(see page 54)*	
286	66 : 01 : 89	UInt	Average voltage n-4 (V)	1
287	66 : 01 : 8A	UInt	L3-L1 voltage n-4 (V)	1
288	66 : 01 : 8B	UInt	L1-L2 voltage n-4 (V)	1
289	66 : 01 : 8C	UInt	L2-L3 voltage n-4 (V)	1
290	66 : 01 : 8D	UInt	Voltage phase imbalance n-4 (x 1%)	1
291	66 : 01 : 8E	UInt	Active power n-4 (x 0.1 kWh)	1
292	66 : 01 : 8F	UInt	Power factor n-4 (x 0.01)	1
293-299	66 : 01 : 90 - 66 : 01 : 96		*(Not significant)*	

Last Fault (n-0) Statistics Extension

The last fault main statistics are listed at addresses 150 to 179.

Register	DeviceNet Address	Variable Type	Read-only Variables	Note *(see page 51)*
300-301	67 : 01 : 01 - 67 : 01 : 02	UDInt	Average current n-0 (x 0.01 A)	
302-303	67 : 01 : 03 - 67 : 01 : 04	UDInt	L1 current n-0 (x 0.01 A)	
304-305	67 : 01 : 05 - 67 : 01 : 06	UDInt	L2 current n-0 (x 0.01 A)	
306-307	67 : 01 : 07 - 67 : 01 : 08	UDInt	L3 current n-0 (x 0.01 A)	
308-309	67 : 01 : 09 - 67 : 01 : 0A	UDInt	Ground current n-0 (mA)	
310	67 : 01 : 0B	UInt	Motor temperature sensor degree n-0 (°C)	

N-1 Fault Statistics Extension

The n-1 fault main statistics are listed at addresses 180 to 209.

Register	DeviceNet Address	Variable Type	Read-only Variables	Note *(see page 51)*
330-331	67 : 01 : 1F - 67 : 01 : 20	UDInt	Average current n-1 (x 0.01 A)	
332-333	67 : 01 : 21 - 67 : 01 : 22	UDInt	L1 current n-1 (x 0.01 A)	
334-335	67 : 01 : 23 - 67 : 01 : 24	UDInt	L2 current n-1 (x 0.01 A)	
336-337	67 : 01 : 25 - 67 : 01 : 26	UDInt	L3 current n-1 (x 0.01 A)	
338-339	67 : 01 : 27 - 67 : 01 : 28	UDInt	Ground current n-1 (mA)	
340	67 : 01 : 29	UInt	Motor temperature sensor degree n-1 (°C)	

N-2 Fault Statistics Extension

The n-2 fault main statistics are listed at addresses 210 to 239.

Register	DeviceNet Address	Variable Type	Read-only Variables	Note *(see page 51)*
360-361	67 : 01 : 3D - 67 : 01 : 3E	UDInt	Average current n-2 (x 0.01 A)	
362-363	67 : 01 : 3F - 67 : 01 : 40	UDInt	L1 current n-2 (x 0.01 A)	
364-365	67 : 01 : 41 - 67 : 01 : 42	UDInt	L2 current n-2 (x 0.01 A)	
366-367	67 : 01 : 43 - 67 : 01 : 44	UDInt	L3 current n-2 (x 0.01 A)	
368-369	67 : 01 : 45 - 67 : 01 : 46	UDInt	Ground current n-2 (mA)	
370	67 : 01 : 47	UInt	Motor temperature sensor degree n-2 (°C)	

N-3 Fault Statistics Extension

The n-3 fault main statistics are listed at addresses 240 to 269.

Register	DeviceNet Address	Variable Type	Read-only Variables	Note *(see page 51)*
390-391	67 : 01 : 5B - 67 : 01 : 5C	UDInt	Average current n-3 (x 0.01 A)	
392-393	67 : 01 : 5D - 67 : 01 : 5E	UDInt	L1 current n-3 (x 0.01 A)	
394-395	67 : 01 : 5F - 67 : 01 : 60	UDInt	L2 current n-3 (x 0.01 A)	
396-397	67 : 01 : 61 - 67 : 01 : 62	UDInt	L3 current n-3 (x 0.01 A)	
398-399	67 : 01 : 63 - 67 : 01 : 64	UDInt	Ground current n-3 (mA)	
400	67 : 01 : 65	UInt	Motor temperature sensor degree n-3 (°C)	

N-4 Fault Statistics Extension

The n-4 fault main statistics are listed at addresses 270 to 299.

Register	DeviceNet Address	Variable Type	Read-only Variables	Note *(see page 51)*
420-421	67 : 01 : 79 - 67 : 01 : 7A	UDInt	Average current n-4 (x 0.01 A)	
422-423	67 : 01 : 7B - 67 : 01 : 7C	UDInt	L1 current n-4 (x 0.01 A)	
424-425	67 : 01 : 7D - 67 : 01 : 7E	UDInt	L2 current n-4 (x 0.01 A)	
426-427	67 : 01 : 7F - 67 : 01 : 80	UDInt	L3 current n-4 (x 0.01 A)	
428-429	67 : 01 : 81 - 67 : 01 : 82	UDInt	Ground current n-4 (mA)	
430	67 : 01 : 83	UInt	Motor temperature sensor degree n-4 (°C)	

Monitoring Variables

Monitoring Overview

Monitoring variables are grouped according to the following criteria:

Monitoring Variable Groups	Registers	DeviceNet Addresses
Monitoring of faults	450 to 454	68 : 01 : 01 to 68 : 01 : 05
Monitoring of status	455 to 459	68 : 01 : 06 to 68 : 01 : 0A
Monitoring of warnings	460 to 464	68 : 01 : 0B to 68 : 01 : 0F
Monitoring of measurements	465 to 539	68 : 01 : 10 to 68 : 01 : 5A

Monitoring of Faults

Variables for monitoring of faults are described in the following table:

Register	DeviceNet Address	Variable Type	Read-only Variables	Note *(see page 51)*
450	68 : 01 : 01	UInt	Minimum wait time (s)	
451	68 : 01 : 02	UInt	Fault code (code of the last fault, or of the fault that takes priority) *(see page 55)*	
452	68 : 01 : 03	Word	Fault register 1	
			bits 0-1 (Reserved)	
			bit 2 Ground current fault	
			bit 3 Thermal overload fault	
			bit 4 Long start fault	
			bit 5 Jam fault	
			bit 6 Current phase imbalance fault	
			bit 7 Undercurrent fault	
			bit 8 (Reserved)	
			bit 9 Test fault	
			bit 10 HMI port fault	
			bit 11 Controller internal fault	
			bit 12 Internal port fault	
			bit 13 (Not significant)	
			bit 14 Network port config fault	
			bit 15 Network port fault	
453	68 : 01 : 04	Word	Fault register 2	
			bit 0 External system fault	
			bit 1 Diagnostic fault	
			bit 2 Wiring fault	
			bit 3 Overcurrent fault	
			bit 4 Current phase loss fault	
			bit 5 Current phase reversal fault	
			bit 6 Motor temperature sensor fault	1
			bit 7 Voltage phase imbalance fault	1
			bit 8 Voltage phase loss fault	1
			bit 9 Voltage phase reversal fault	1
			bit 10 Undervoltage fault	1
			bit 11 Overvoltage fault	1
			bit 12 Underpower fault	1
			bit 13 Overpower fault	1
			bit 14 Under power factor fault	1
			bit 15 Over power factor fault	1

Register	DeviceNet Address	Variable Type	Read-only Variables	Note (see page 51)
454	68 : 01 : 05	Word	Fault register 3	
			bit 0 LTME configuration fault	
			bits 1-15 (Reserved)	

Monitoring of Status

Variables for monitoring of status are described in the following table:

Register	DeviceNet Address	Variable Type	Read-only Variables	Note (see page 51)
455	68 : 01 : 06	Word	System status register 1	
			bit 0 System ready	
			bit 1 System on	
			bit 2 System fault	
			bit 3 System warning	
			bit 4 System tripped	
			bit 5 Fault reset authorized	
			bit 6 Controller power	
			bit 7 Motor running (with detection of a current, if greater than 10% FLC)	
			bits 8-13 Motor average current ratio 32 = 100% FLC - 63 = 200% FLC	
			bit 14 In remote	
			bit 15 Motor starting (start in progress) 0 = descending current is less than 150% FLC 1 = ascending current is greater than 10% FLC	
456	68 : 01 : 07	Word	System status register 2	
			bit 0 Auto-reset active	
			bit 1 (Not significant)	
			bit 2 Fault power cycle requested	
			bit 3 Motor restart time undefined	
			bit 4 Rapid cycle lockout	
			bit 5 Load shedding	1
			bit 6 Motor speed 0 = FLC1 setting is used 1 = FLC2 setting is used	
			bit 7 HMI port comm loss	
			bit 8 Network port comm loss	
			bit 9 Motor transition lockout	
			bits 10-15 (Not significant)	

Register	DeviceNet Address	Variable Type	Read-only Variables	Note (see page 51)
457	68 : 01 : 08	Word	Logic inputs status	
			bit 0 Logic input 1	
			bit 1 Logic input 2	
			bit 2 Logic input 3	
			bit 3 Logic input 4	
			bit 4 Logic input 5	
			bit 5 Logic input 6	
			bit 6 Logic input 7	
			bit 7 Logic input 8	1
			bit 8 Logic input 9	1
			bit 9 Logic input 10	1
			bit 10 Logic input 11	1
			bit 11 Logic input 12	1
			bit 12 Logic input 13	1
			bit 13 Logic input 14	1
			bit 14 Logic input 15	1
			bit 15 Logic input 16	1
458	68 : 01 : 09	Word	Logic outputs status	
			bit 0 Logic output 1	
			bit 1 Logic output 2	
			bit 2 Logic output 3	
			bit 3 Logic output 4	
			bit 4 Logic output 5	1
			bit 5 Logic output 6	1
			bit 6 Logic output 7	1
			bit 7 Logic output 8	1
			bits 8-15 (Reserved)	
459	68 : 01 : 0A	Word	I/O status	
			bit 0 Input 1	
			bit 1 Input 2	
			bit 2 Input 3	
			bit 3 Input 4	
			bit 4 Input 5	
			bit 5 Input 6	
			bit 6 Input 7	
			bit 7 Input 8	
			bit 8 Input 9	
			bit 9 Input 10	
			bit 10 Input 11	
			bit 11 Input 12	
			bit 12 Output 1 (13-14)	
			bit 13 Output 2 (23-24)	
			bit 14 Output 3 (33-34)	
			bit 15 Output 4 (95-96, 97-98)	

Monitoring of Warnings

Variables for monitoring of warnings are described in the following table:

Register	DeviceNet Address	Variable Type	Read-only Variables	Note (see page 51)
460	68 : 01 : 0B	UInt	Warning code *(see page 57)*	
461	68 : 01 : 0C	Word	Warning register 1	
			bits 0-1 (Not significant)	
			bit 2 Ground current warning	
			bit 3 Thermal overload warning	
			bit 4 (Not significant)	
			bit 5 Jam warning	
			bit 6 Current phase imbalance warning	
			bit 7 Undercurrent warning	
			bits 8-9 (Not significant)	
			bit 10 HMI port warning	
			bit 11 Controller internal temperature warning	
			bits 12-14 (Not significant)	
			bit 15 Network port warning	
462	68 : 01 : 0D	Word	Warning register 2	
			bit 0 (Not significant)	
			bit 1 Diagnostic warning	
			bit 2 (Reserved)	
			bit 3 Overcurrent warning	
			bit 4 Current phase loss warning	
			bit 5 Current phase reversal warning	
			bit 6 Motor temperature sensor warning	
			bit 7 Voltage phase imbalance warning	1
			bit 8 Voltage phase loss warning	1
			bit 9 (Not significant)	
			bit 10 Undervoltage warning	1
			bit 11 Overvoltage warning	1
			bit 12 Underpower warning	1
			bit 13 Overpower warning	1
			bit 14 Under power factor warning	1
			bit 15 Over power factor warning	1
463	68 : 01 : 0E	Word	Warning register 3	
			bit 0 LTME configuration warning	
			bits 1-15 (Reserved)	
464	68 : 01 : 0F	UInt	Motor temperature sensor degree (°C)	

Monitoring of Measurements

Variables for monitoring of measurements are described in the following table:

Register	DeviceNet Address	Variable Type	Read-only Variables	Note (see page 51)
465	68 : 01 : 10	UInt	Thermal capacity level (% trip level)	
466	68 : 01 : 11	UInt	Average current ratio (% FLC)	
467	68 : 01 : 12	UInt	L1 current ratio (% FLC)	
468	68 : 01 : 13	UInt	L2 current ratio (% FLC)	
469	68 : 01 : 14	UInt	L3 current ratio (% FLC)	
470	68 : 01 : 15	UInt	Ground current ratio (x 0.1 % FLC min)	
471	68 : 01 : 16	UInt	Current phase imbalance (%)	

Register	DeviceNet Address	Variable Type	Read-only Variables	Note *(see page 51)*
472	68 : 01 : 17	Int	Controller internal temperature (°C)	
473	68 : 01 : 18	UInt	Controller config checksum	
474	68 : 01 : 19	UInt	Frequency (x 0.01 Hz)	2
475	68 : 01 : 1A	UInt	Motor temperature sensor (x 0.1 Ω)	
476	68 : 01 : 1B	UInt	Average voltage (V)	1
477	68 : 01 : 1C	UInt	L3-L1 voltage (V)	1
478	68 : 01 : 1D	UInt	L1-L2 voltage (V)	1
479	68 : 01 : 1E	UInt	L2-L3 voltage (V)	1
480	68 : 01 : 1F	UInt	Voltage phase imbalance (%)	1
481	68 : 01 : 20	UInt	Power factor (x 0.01)	1
482	68 : 01 : 21	UInt	Active power (x 0.1 kW)	1
483	68 : 01 : 22	UInt	Reactive power (x 0.1 kVAR)	1
484	68 : 01 : 23	Word	Auto restart status register	
			bit 0 Voltage dip occurred	
			bit 1 Voltage dip detection	
			bit 2 Auto restart immediate condition	
			bit 3 Auto restart delayed condition	
			bit 4 Auto restart manual condition	
			bits 5-15 (Not significant)	
485	68 : 01 : 24	Word	Controller last power OFF duration	
486-489	68 : 01 : 25 - 68 : 01 : 28		*(Not significant)*	
490	68 : 01 : 29	Word	Network port monitoring	
			bit 0 Network port communicating	
			bit 1 Network port connected	
			bit 2 Network port self-testing	
			bit 3 Network port self-detecting	
			bit 4 Network port bad config	
			bits 5-15 (Not significant)	
491	68 : 01 : 2A	UInt	Network port baud rate *(see page 55)*	
492	68 : 01 : 2B		*(Not significant)*	
493	68 : 01 : 2C	UInt	Network port parity *(see page 55)*	
494-499	68 : 01 : 2D - 68 : 01 : 32		*(Not significant)*	
500-501	68 : 01 : 33 - 68 : 01 : 34	UDInt	Average current (x 0.01 A)	
502-503	68 : 01 : 35 - 68 : 01 : 36	UDInt	L1 current (x 0.01 A)	
504-505	68 : 01 : 37 - 68 : 01 : 38	UDInt	L2 current (x 0.01 A)	
506-507	68 : 01 : 39 - 68 : 01 : 3A	UDInt	L3 current (x 0.01 A)	
508-509	68 : 01 : 3B - 68 : 01 : 3C	UDInt	Ground current (mA)	
510	68 : 01 : 3D	UInt	Controller port ID	
511	68 : 01 : 3E	UInt	Time to trip (x 1 s)	
512	68 : 01 : 3F	UInt	Motor last start current ratio (% FLC)	
513	68 : 01 : 40	UInt	Motor last start duration (s)	
514	68 : 01 : 41	UInt	Motor starts per hour count	

Register	DeviceNet Address	Variable Type	Read-only Variables	Note (see page 51)
515	68 : 01 : 42	Word	Phase imbalances register	
			bit 0 L1 current highest imbalance	
			bit 1 L2 current highest imbalance	
			bit 2 L3 current highest imbalance	
			bit 3 L1-L2 voltage highest imbalance	1
			bit 4 L2-L3 voltage highest imbalance	1
			bit 5 L3-L1 voltage highest imbalance	1
			bits 6-15 (Not significant)	
516-523	68 : 01 : 43 - 68 : 01 : 5A		*(Reserved)*	
524-539	68 : 01 : 4B - 68 : 01 : 5A		*(Forbidden)*	

Configuration Variables

Configuration Overview

Configuration variables are grouped according to the following criteria

Configuration Variable Groups	Registers	DeviceNet Addresses
Configuration	540 to 649	69 : 01 : 01 to 6A : 01 : 32
Setting	650 to 699	6B : 01 : 01 to 6B : 01 : 32

Configuration Variables

The configuration variables are described in the following tables:

Register	DeviceNet Address	Variable Type	Read/Write Variables	Note *(see page 51)*
540	69 : 01 : 01	UInt	Motor operating mode 2 = 2-wire overload 3 = 3-wire overload 4 = 2-wire independent 5 = 3-wire independent 6 = 2-wire reverser 7 = 3-wire reverser 8 = 2-wire 2-step 9 = 3-wire 2-step 10 = 2-wire 2-speed 11 = 3-wire 2-speed 256-511 = Custom logic program (0-255)	B
541	69 : 01 : 02	UInt	Motor transition timeout (s) *(see page 53)*	
542-544	69 : 01 : 03 - 6A : 01 : 05		*(Reserved)*	
545	69 : 01 : 06	Word	Controller AC inputs setting register	
			bits 0-3 Controller AC logic inputs configuration *(see page 53)*	
			bits 4-15 *(Reserved)*	
546	69 : 01 : 07	UInt	Thermal overload setting	B
			bits 0-2 Motor temperature sensor type: 0 = None 1 = PTC binary 2 = PT100 3 = PTC analog 4 = NTC analog	
			bits 3-4 Thermal overload mode: 0 = Definite 2 = Inverse thermal	
			bits 5-15 *(Reserved)*	
547	69 : 01 : 08	UInt	Thermal overload fault definite timeout (s)	
548	6A : 01 : 09		*(Reserved)*	
549	69 : 01 : 0A	UInt	Motor temperature sensor fault threshold (x 0.1 Ω)	
550	69 : 01 : 0B	UInt	Motor temperature sensor warning threshold (x 0.1 Ω)	
551	69 : 01 : 0C	UInt	Motor temperature sensor fault threshold degree (°C)	
552	6A : 01 : 0D	UInt	Motor temperature sensor warning threshold degree (°C)	
553	69 : 01 : 0E	UInt	Rapid cycle lockout timeout (s)	
554	69 : 01 : 0F		*(Reserved)*	
555	69 : 01 : 10	UInt	Current phase loss timeout (x 0.1 s)	
556	69 : 01 : 11	UInt	Overcurrent fault timeout (s)	
557	69 : 01 : 12	UInt	Overcurrent fault threshold (% FLC)	
558	69 : 01 : 13	UInt	Overcurrent warning threshold (% FLC)	

Register	DeviceNet Address	Variable Type	Read/Write Variables	Note (see page 51)
559	69 : 01 : 14	Word	Ground current fault configuration	B
			bit 0 Ground current mode	
			bits 1-15 *(Reserved)*	
560	69 : 01 : 15	UInt	Ground CT primary	
561	69 : 01 : 16	UInt	Ground CT secondary	
562	69 : 01 : 17	UInt	External ground current fault timeout (x 0.01 s)	
563	69 : 01 : 18	UInt	External ground current fault threshold (x 0.01 A)	
564	69 : 01 : 19	UInt	External ground current warning threshold (x 0.01 A)	
565	69 : 01 : 1A	UInt	Motor nominal voltage (V)	1
566	69 : 01 : 1B	UInt	Voltage phase imbalance fault timeout starting (x 0.1 s)	1
567	69 : 01 : 1C	UInt	Voltage phase imbalance fault timeout running (x 0.1 s)	1
568	69 : 01 : 1D	UInt	Voltage phase imbalance fault threshold (% imb)	1
569	69 : 01 : 1E	UInt	Voltage phase imbalance warning threshold (% imb)	1
570	69 : 01 : 1F	UInt	Overvoltage fault timeout (x 0.1 s)	1
571	69 : 01 : 20	UInt	Overvoltage fault threshold (% Vnom)	1
572	69 : 01 : 21	UInt	Overvoltage warning threshold (% Vnom)	1
573	69 : 01 : 22	UInt	Undervoltage fault timeout (x 0.1 s)	1
574	69 : 01 : 23	UInt	Undervoltage fault threshold (% Vnom)	1
575	69 : 01 : 24	UInt	Undervoltage warning threshold (% Vnom)	1
576	69 : 01 : 25	UInt	Voltage phase loss fault timeout (x 0.1 s)	1
577	69 : 01 : 26	Word	Voltage dip setting	1
			bit 0 Load shedding enable	
			bit 1 Auto-restart enable	
			bits 2-15 *(Reserved)*	
578	69 : 01 : 27	UInt	Load shedding timeout (s)	1
579	69 : 01 : 28	UInt	Voltage dip threshold (% Vnom)	1
580	69 : 01 : 29	UInt	Voltage dip restart timeout (s)	1
581	69 : 01 : 2A	UInt	Voltage dip restart threshold (% Vnom)	1
582	69 : 01 : 2B	UInt	Auto restart immediate timeout (x 0.1 s)	
583	69 : 01 : 2C	UInt	Motor nominal power (x 0.1 kW)	1
584	69 : 01 : 2D	UInt	Overpower fault timeout (s)	1
585	69 : 01 : 2E	UInt	Overpower fault threshold (% Pnom)	1
586	69 : 01 : 2F	UInt	Overpower warning threshold (% Pnom)	1
587	69 : 01 : 30	UInt	Underpower fault timeout (s)	1
588	69 : 01 : 31	UInt	Underpower fault threshold (% Pnom)	1
589	69 : 01 : 32	UInt	Underpower warning threshold (% Pnom)	1
590	69 : 01 : 33	UInt	Under power factor fault timeout (x 0.1 s)	1
591	69 : 01 : 34	UInt	Under power factor fault threshold (x 0.01 PF)	1
592	69 : 01 : 35	UInt	Under power factor warning threshold (x 0.01 PF)	1
593	69 : 01 : 36	UInt	Over power factor fault timeout (x 0.1 s)	1
594	69 : 01 : 37	UInt	Over power factor fault threshold (x 0.01 PF)	1
595	69 : 01 : 38	UInt	Over power factor warning threshold (x 0.01 PF)	1
596	69 : 01 : 39	UInt	Auto restart delayed timeout (s)	
597-599	69 : 01 : 3A - 69 : 01 : 3C		*(Reserved)*	
600	6A : 01 : 01		*(Not significant)*	

Register	DeviceNet Address	Variable Type	Read/Write Variables	Note *(see page 51)*
601	6A : 01 : 02	Word	General configuration register 1	
			bit 0 Controller system config required: 0 = exit the configuration menu 1 = go to the configuration menu	A
			bits 1-7 *(Reserved)*	
			Control mode configuration, bits 8-10 (one bit is set to 1):	
			bit 8 Config via HMI keypad enable	
			bit 9 Config via HMI engineering tool enable	
			bit 10 Config via network port enable	
			bit 11 Motor star-delta	B
			bit 12 Motor phases sequence: 0 = A B C 1 = A C B	
			bits 13-14 Motor phases *(see page 57)*	B
			bit 15 Motor auxiliary fan cooled (factory setting = 0)	
602	6A : 01 : 03	Word	General configuration register 2	
			bits 0-2 Fault reset mode *(see page 57)*	C
			bit 3 HMI port parity setting: 0 = none 1 = even (factory setting)	
			bits 4-8 *(Reserved)*	
			bit 9 HMI port endian setting	
			bit 10 Network port endian setting	
			bit 11 HMI motor status LED color	
			bits 12-15 *(Reserved)*	
603	6A : 01 : 04	UInt	HMI port address setting	
604	6A : 01 : 05	UInt	HMI port baud rate setting (Baud)	
605	6A : 01 : 06		*(Reserved)*	
606	6A : 01 : 07	UInt	Motor trip class (s)	
607	6A : 01 : 08		*(Reserved)*	
608	6A : 01 : 09	UInt	Thermal overload fault reset threshold (% trip level)	
609	6A : 01 : 0A	UInt	Thermal overload warning threshold (% trip level)	
610	6A : 01 : 0B	UInt	Internal ground current fault timeout (x 0.1 s)	
611	6A : 01 : 0C	UInt	Internal ground current fault threshold (% FLCmin)	
612	6A : 01 : 0D	UInt	Internal ground current warning threshold (% FLCmin)	
613	6A : 01 : 0E	UInt	Current phase imbalance fault timeout starting (x 0.1 s)	
614	6A : 01 : 0F	UInt	Current phase imbalance fault timeout running (x 0.1 s)	
615	6A : 01 : 10	UInt	Current phase imbalance fault threshold (% imb)	
616	6A : 01 : 11	UInt	Current phase imbalance warning threshold (% imb)	
617	6A : 01 : 12	UInt	Jam fault timeout (s)	
618	6A : 01 : 13	UInt	Jam fault threshold (% FLC)	
619	6A : 01 : 14	UInt	Jam warning threshold (% FLC)	
620	6A : 01 : 15	UInt	Undercurrent fault timeout (s)	
621	6A : 01 : 16	UInt	Undercurrent fault threshold (% FLC)	
622	6A : 01 : 17	UInt	Undercurrent warning threshold (% FLC)	
623	6A : 01 : 18	UInt	Long start fault timeout (s)	
624	6A : 01 : 19	UInt	Long start fault threshold (% FLC)	
625	6A : 01 : 1A		*(Reserved)*	

Register	DeviceNet Address	Variable Type	Read/Write Variables	Note (see page 51)
626	6A : 01 : 1B	UInt	HMI display contrast setting	
			bits 0-7 HMI display contrast setting	
			HMI display brightness setting	
627	6A : 01 : 1C	UInt	Contactor rating (0.1 A)	
628	6A : 01 : 1D	UInt	Load CT primary	B
629	6A : 01 : 1E	UInt	Load CT secondary	B
630	6A : 01 : 1F	UInt	Load CT multiple passes (passes)	B
631	6A : 01 : 20	Word	Fault enable register 1	
			bits 0-1 *(Reserved)*	
			bit 2 Ground current fault enable	
			bit 3 Thermal overload fault enable	
			bit 4 Long start fault enable	
			bit 5 Jam fault enable	
			bit 6 Current phase imbalance fault enable	
			bit 7 Undercurrent fault enable	
			bit 8 *(Reserved)*	
			bit 9 Self test enable 0 = disable 1 = enable (factory setting)	
			bit 10 HMI port fault enable	
			bits 11-14 *(Reserved)*	
			bit 15 Network port fault enable	
632	6A : 01 : 21	Word	Warning enable register 1	
			bit 0 *(Not significant)*	
			bit 1 *(Reserved)*	
			bit 2 Ground current warning enable	
			bit 3 Thermal overload warning enable	
			bit 4 *(Reserved)*	
			bit 5 Jam warning enable	
			bit 6 Current phase imbalance warning enable	
			bit 7 Undercurrent warning enable	
			bits 8-9 *(Reserved)*	
			bit 10 HMI port warning enable	
			bit 11 Controller internal temperature warning enable	
			bits 12-14 *(Reserved)*	
			bit 15 Network port warning enable	

Register	DeviceNet Address	Variable Type	Read/Write Variables	Note (see page 51)
633	6A : 01 : 22	Word	Fault enable register 2	
			bit 0 *(Reserved)*	
			bit 1 Diagnostic fault enable	
			bit 2 Wiring fault enable	
			bit 3 Overcurrent fault enable	
			bit 4 Current phase loss fault enable	
			bit 5 Current phase reversal fault enable	
			bit 6 Motor temperature sensor fault enable	
			bit 7 Voltage phase imbalance fault enable	1
			bit 8 Voltage phase loss fault enable	1
			bit 9 Voltage phase reversal fault enable	1
			bit 10 Undervoltage fault enable	1
			bit 11 Overvoltage fault enable	1
			bit 12 Underpower fault enable	1
			bit 13 Overpower fault enable	1
			bit 14 Under power factor fault enable	1
			bit 15 Over power factor fault enable	1
634	6A : 01 : 23	Word	Warning enable register 2	
			bit 0 *(Reserved)*	
			bit 1 Diagnostic warning enable	
			bit 2 *(Reserved)*	
			bit 3 Overcurrent warning enable	
			bit 4 Current phase loss warning enable	
			bit 5 *(Reserved)*	
			bit 6 Motor temperature sensor warning enable	
			bit 7 Voltage phase imbalance warning enable	1
			bit 8 Voltage phase loss warning enable	1
			bit 9 *(Reserved)*	1
			bit 10 Undervoltage warning enable	1
			bit 11 Overvoltage warning enable	1
			bit 12 Underpower warning enable	1
			bit 13 Overpower warning enable	1
			bit 14 Under power factor warning enable	1
			bit 15 Over power factor warning enable	1
635-6	6A : 01 : 24 - 6A : 01 : 25		*(Reserved)*	
637	6A : 01 : 26	UInt	Auto-reset attempts group 1 setting	
638	6A : 01 : 27	UInt	Auto-reset group 1 timeout	
639	6A : 01 : 28	UInt	Auto-reset attempts group 2 setting	
640	6A : 01 : 29	UInt	Auto-reset group 2 timeout	
641	6A : 01 : 2A	UInt	Auto-reset attempts group 3 setting	
642	6A : 01 : 2B	UInt	Auto-reset group 3 timeout	
643	6A : 01 : 2C	UInt	Motor step 1 to 2 timeout	
644	6A : 01 : 2D	UInt	Motor step 1 to 2 threshold	
645	6A : 01 : 2E	UInt	HMI port fallback setting *(see page 57)*	
646-649	6A : 01 : 2F - 6A : 01 : 32		*(Reserved)*	

Setting Variables

The setting variables are described in the following table:

Register	DeviceNet Address	Variable Type	Read/Write Variables	Note *(see page 51)*
650	6B : 01 : 01	Word	HMI language setting register:	
			bit 0-4 HMI language setting *(see page 57)*	
			bits 5-15 *(Not significant)*	
651	6B : 01 : 02	Word	HMI display items register 1	
			bit 0 HMI display average current enable	
			bit 1 HMI display thermal capacity level enable	
			bit 2 HMI display L1 current enable	
			bit 3 HMI display L2 current enable	
			bit 4 HMI display L3 current enable	
			bit 5 HMI display ground current enable	
			bit 6 HMI display motor status enable	
			bit 7 HMI display current phase imbalance enable	
			bit 8 HMI display operating time enable	
			bit 9 HMI display I/O status enable	
			bit 10 HMI display reactive power enable	
			bit 11 HMI display frequency enable	
			bit 12 HMI display starts per hour enable	
			bit 13 HMI display control mode enable	
			bit 14 HMI display start statistics enable	
			bit 15 HMI motor temperature sensor enable	
652	6B : 01 : 03	UInt	Motor full load current ratio, FLC1 (% FLCmax)	
653	6B : 01 : 04	UInt	Motor high speed full load current ratio, FLC2 (% FLCmax)	
654	6B : 01 : 05	Word	HMI display items register 2	
			bit 0 HMI display L1-L2 voltage enable	1
			bit 1 HMI display L2-L3 voltage enable	1
			bit 2 HMI display L3-L1 voltage enable	1
			bit 3 HMI display average voltage enable	1
			bit 4 HMI display active power enable	1
			bit 5 HMI display power consumption enable	1
			bit 6 HMI display power factor enable	1
			bit 7 HMI display average current ratio enable	
			bit 8 HMI display L1 current ratio enable	1
			bit 9 HMI display L2 current ratio enable	1
			bit 10 HMI display L3 current ratio enable	1
			bit 11 HMI display thermal capacity remaining enable	
			bit 12 HMI display time to trip enable	
			bit 13 HMI display voltage phase imbalance enable	1
			bit 14 HMI display date enable	
			bit 15 HMI display time enable	
655-658	6B : 01 : 06 - 6B : 01 : 09	Word[4]	Date and time setting *(see page 54)*	
659	6B : 01 : 0A	Word[4]	HMI display items register 3	
			bit 0 HMI display temperature sensor degree CF	
			bits 1-15 *(Reserved)*	
660-681	6B : 01 : 0B - 6B : 01 : 20		*(Reserved)*	
682	6B : 01 : 21	UInt	Network port fallback setting *(see page 57)*	

Register	DeviceNet Address	Variable Type	Read/Write Variables	Note *(see page 51)*
683	6B : 01 : 22	Word	Control setting register	
			bits 0-1 *(Reserved)*	
			bits 2 Control remote local default mode (with LTMCU) 0 = remote 1 = local	
			bit 3 *(Reserved)*	
			bit 4 Control remote local buttons enable (with LTMCU) 0 = disable 1 = enable	
			bits 5-6 Control remote channel setting (with LTMCU) 0 = network 1 = terminal strip 2 = HMI	
			bit 7 *(Reserved)*	
			bit 8 Control local channel setting 0 = terminal strip 1 = HMI	
			bit 9 Control direct transition 0 = stop required during transition 1 = stop not required during transition	
			bit 10 Control transfer mode 0 = bump 1 = bumpless	
			bit 11 Stop terminal strip disable 0 = enable 1 = disable	
			bit 12 Stop HMI disable 0 = enable 1 = disable	
			bits 13-15 *(Reserved)*	
684-694	6B : 01 : 23 - 6B : 01 : 2D		*(Reserved)*	
695	6B : 01 : 2E	UInt	Network port baud rate setting (Baud) *(see page 55)*	
696	6B : 01 : 2F	UInt	Network port address setting	
697-699	6B : 01 : 30 - 6B : 01 : 32		*(Not significant)*	

Command Variables

Command Variables

Command Variables

Command variables are described in the following table:

Register	DeviceNet Address	Variable Type	Read/Write Variables	Note *(see page 51)*
700	6C : 01 : 01	Word	Register available to remotely write commands that can be processed in a specific custom logic	
701-703	6C : 01 : 02 - 6C : 01 : 04		*(Reserved)*	
704	6C : 01 : 05	Word	Control register 1	
			bit 0 Motor run forward command (1)	
			bit 1 Motor run reverse command (1)	
			bit 2 (Reserved)	
			bit 3 Fault reset command	
			bit 4 (Reserved)	
			bit 5 Self test command	
			bit 6 Motor low speed command	
			bits 7-15 (Reserved)	
705	6C : 01 : 06	Word	Control register 2	
			bit 0 Clear all command Clear all parameters, except: ● Motor LO1 closings count ● Motor LO2 closings count ● Controller internal temperature max ● Thermal capacity level	
			bit 1 Clear statistics command	
			bit 2 Clear thermal capacity level command	
			bit 3 Clear controller settings command	
			bit 4 Clear network port settings command	
			bits 5-15 (Reserved)	
706-709	6C : 01 : 07 - 6C : 01 : 0A		*(Reserved)*	
710-799	6C : 01 : 08 - 6C : 01 : 64		*(Forbidden)*	
(1) Even in Overload mode, bits 0 and 1 of register 704 can be used to remotely control LO1 and LO2.				

Custom Logic Variables

Custom Logic Variables

Custom Logic Variables

Custom logic variables are described in the following tables:

Register	DeviceNet Address	Variable Type	Read-only Variables	Note *(see page 51)*
1200	71 : 01 : 01	Word	Custom logic status register	
			bit 0 Custom logic run	
			bit 1 Custom logic stop	
			bit 2 Custom logic reset	
			bit 3 Custom logic second step	
			bit 4 Custom logic transition	
			bit 5 Custom logic phase reverse	
			bit 6 Custom logic network control	
			bit 7 Custom logic FLC selection	
			bit 8 (Reserved)	
			bit 9 Custom logic auxiliary 1 LED	
			bit 10 Custom logic auxiliary 2 LED	
			bit 11 Custom logic stop LED	
			bit 12 Custom logic LO1	
			bit 13 Custom logic LO2	
			bit 14 Custom logic LO3	
			bit 15 Custom logic LO4	
1201	71 : 01 : 02	Word	Custom logic version	
1202	71 : 01 : 03	Word	Custom logic memory space	
1203	71 : 01 : 04	Word	Custom logic memory used	
1204	71 : 01 : 05	Word	Custom logic temporary space	
1205	71 : 01 : 06	Word	Custom logic non volatile space	
1206-1249	71 : 01 : 0C - 71 : 01 : 32		*(Reserved)*	

Register	DeviceNet Address	Variable Type	Read/Write Variables	Note *(see page 51)*
1250	71 : 01 : 33	Word	Custom logic setting register 1	
			bit 0 (Reserved)	
			bit 1 Logic input 3 external ready enable	
			bits 2-15 (Reserved)	
1251-1269	71 : 01 : 34 - 71 : 01 : 46		*(Reserved)*	
1270	71 : 01 : 47	Word	Custom logic command register 1	
			bit 0 Custom logic external fault command	
			bits 1-15 (Reserved)	
1271-1279	71 : 01 : 48 - 71 : 01 : 50		*(Reserved)*	

Register	DeviceNet Address	Variable Type	Read-only Variables	Note *(see page 51)*
1280	71 : 01 : 51	Word	Custom logic monitoring register 1	
			bit 0 (Reserved)	
			bit 1 Custom logic system ready	
			bits 2-15 (Reserved)	
1281-1300	71 : 01 : 52 - 71 : 01 : 65		*(Reserved)*	

Register	DeviceNet Address	Variable Type	Read/Write Variables	Note *(see page 51)*
1301-1399	71 : 01 : 66 - 71 : 01 : C8	Word[99]	General purpose registers for logic functions	

Glossary

A

active power

Also known as *real power*, active power is the rate of producing, transferring or using electrical energy. It is measured in watts (W) and often expressed in kilowatts (kW) or megawatts (MW).

analog

Describes inputs (e.g. temperature) or outputs (e.g. motor speed) that can be set to a range of values. Contrast with discrete.

apparent power

The product of current and voltage, apparent power consists of both active power and reactive power. It is measured in volt-amperes and often expressed in kilovolt-amperes (kVA) or megavolt-amperes (MVA).

C

CANopen

An open industry standard protocol used on the internal communication bus. The protocol allows the connection of any standard CANopen device to the island bus.

CT

current transformer.

D

definite time

A variety of TCC or TVC where the initial magnitude of the trip time delay remains a constant, and does not vary in response to changes in the value of the measured quantity (e.g. current). Contrast with inverse thermal.

device

In the broadest terms, any electronic unit that can be added to a network. More specifically, a programmable electronic unit (e.g. PLC, numeric controller or robot) or I/O card.

DeviceNet™

DeviceNet™ is a low-level, connection-based network protocol that is based on CAN, a serial bus system without a defined application layer. DeviceNet, therefore, defines a layer for the industrial application of CAN.

DIN

Deutsches Institut für Normung. The European organization that organizes the creation and maintenance of dimensional and engineering standards.

DIN rail

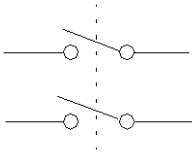
A steel mounting rail, made pursuant to DIN standards (typically 35 mm wide), that allows for easier "snap-on" mounting of IEC electrical devices, including the LTM R controller and the expansion module. Contrast with screw mounting of devices to a control panel by drilling and tapping holes.

discrete

Describes inputs (e.g. switches) or outputs (e.g. coils) that can be only *On* or *Off*. Contrast with analog.

DPST

double-pole/single-throw. A switch that connects or disconnects 2 circuit conductors in a single branch circuit. A DPST switch has 4 terminals, and is the equivalent of 2 single-pole/single-throw switches controlled by a single mechanism, as depicted below:



E

endian setting (big endian)

'big endian' means that the high-order byte/word of the number is stored in memory at the lowest address, and the low-order byte/word at the highest address (the big end comes first).

endian setting (little endian)

'little endian' means that the low-order byte/word of the number is stored in memory at the lowest address, and the high-order byte/word at the highest address (the little end comes first).

EtherNet/IP

(Ethernet Industrial Protocol) is an industrial application protocol built on TCP/IP and CIP protocols. It is mainly used on automated networks, it defines network devices as network objects as to allow the communication between industrial control system and their components; (programmable automation controller, programmable logic controller, I/O systems).

F

FLC

full load current. Also known as *rated current*. The current the motor will draw at the rated voltage and rated load. The LTM R controller has 2 FLC settings: FLC1 (Motor Full Load Current Ratio) and FLC2 (Motor High Speed Full Load Current Ratio), each set as a percentage of FLC max.

FLC1

Motor Full Load Current Ratio. FLC parameter setting for low or single speed motors.

FLC2

Motor High Speed Full Load Current Ratio. FLC parameter setting for high-speed motors.

FLCmax

Full Load Current Max. Peak current parameter.

FLCmin

Minimum Full Load Current. The smallest amount of motor current the LTM R controller will support. This value is determined by the LTM R controller model.

H

hysteresis

A value—added to lower limit threshold settings or subtracted from upper limit threshold settings—that retards the response of the LTM R controller before it stops measuring the duration of faults and warnings.

I

inverse thermal

A variety of TCC where the initial magnitude of the trip time delay is generated by a thermal model of the motor and varies in response to changes in the value of the measured quantity (e.g. current). Contrast with definite time.

M

Modbus®

Modbus® is the name of the master-slave/client-server serial communications protocol developed by Modicon (now Schneider Automation, Inc.) in 1979, which has since become a standard network protocol for industrial automation.

N

nominal power

Motor Nominal Power. Parameter for the power a motor will produce at rated voltage and rated current.

nominal voltage

Motor Nominal Voltage. Parameter for rated voltage.

NTC

negative temperature coefficient. Characteristic of a thermistor—a thermally sensitive resistor—whose resistance increases as its temperature falls, and whose resistance decreases as its temperature rises.

NTC analog

Type of RTD.

P

PLC

programmable logic controller.

power factor

Also called *cosine phi* (or ϕ), power factor represents the absolute value of the ratio of active power to apparent power in AC power systems.

PROFIBUS DP

An open bus system that uses an electrical network based on a shielded 2-wire line or an optical network based on a fiber-optic cable.

PT100

Type of RTD.

PTC

positive temperature coefficient. Characteristic of a thermistor—a thermally sensitive resistor—whose resistance increases as its temperature rises, and whose resistance decreases as its temperature falls.

PTC analog

Type of RTD.

PTC binary

Type of RTD.

R

reset time

Time between a sudden change in the monitored quantity (e.g. current) and the switching of the output relay.

rms

root mean square. A method of calculating average AC current and average AC voltage. Because AC current and AC voltage are bi-directional, the arithmetic average of AC current or voltage always equals 0.

RTD

resistance temperature detector. A thermistor (thermal resistor sensor) used to measure the temperature of the motor. Required by the LTM R controller's Motor Temp Sensor motor protection function.

T

TCC

trip curve characteristic. The type of delay used to trip the flow of current in response to a fault condition. As implemented in the LTM R controller, all motor protection function trip time delays are definite time, except for the Thermal Overload function, which also offers inverse thermal trip time delays.

TVC

trip voltage characteristic. The type of delay used to trip the flow of voltage in response to a fault condition. As implemented by the LTM R controller and the expansion module, all TVCs are definite time.

Index

D

E

F

G

H

I

J

L

M

N

O

P

R

S

T

U

V

W

DOCA0133EN-01

Schneider Electric Industries SAS
35, rue Joseph Monier
CS30323
F - 92506 Rueil Malmaison Cedex

www.schneider-electric.com

07/2018