TeSys® U LULC09 DeviceNet Communication Module

User's Manual

07/2019

Schneider Electric

Table of Contents

Safety Information

Important Information

NOTICE

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

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

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

⚠ DANGER

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

⚠ WARNING

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

⚠ CAUTION

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

CAUTION

CAUTION, used without the safety alert symbol, indicates a potentially hazardous situation which, if not avoided, **can result in** equipment damage.

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 the installation, and has received safety training to recognize and avoid the hazards involved.

About the Book

At a Glance

Document Scope

This manual describes the implementation, functionalities and operation of the TeSys U DeviceNet communication module (LULC09).

Field of application: mainly automation systems in industry and building areas.

Validity Note

This manual is valid for LULC09 V1.3 and later versions.

The Idle Message Management function is valid for LULC09 V1.6 and later versions.

Related Documents

Title of Documentation	Reference Number
LULC09 DeviceNet Module - Instruction Sheet	1639547
TeSys U Communication Variables - User's Manual	1744082
LU•B/LU•S• TeSys U Starters - Instruction Sheet	1629984
LUTM• TeSys U Controller - User's Manual	1743233
LUTM• TeSys U Controller - Instruction Sheet	1743236
LUCM/LUCMT Multifunction Control Units - User's Manual	1743237
LUCM/LUCMT/LUCBT/LUCDT Control Units - Instruction Sheet	AAV40504
LUCA/LUCB/LUCC/LUCD Control Units - Instruction Sheet	AAV40503
Electromagnetic Compatibility - Practical Installation Guidelines	DEG999

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

Part I
Hardware Implementation

This part describes the installation and technical characteristics of a TeSys U DeviceNet communication module (LULC09).

What Is in This Part?

This part contains the following chapters:

Chapter	Chapter Name	Page
1	Installation of TeSys U DeviceNet Module (LULC09)	11
2	Technical Characteristics	27

Chapter 1
Installation of TeSys U DeviceNet Module (LULC09)

This chapter introduces TeSys U DeviceNet communication module (named LULC09) and describes the different physical installation steps of the product. There are different possible configurations, that will be described in this chapter.

What Is in This Chapter?

This chapter contains the following topics:

Presentation of LULC09 DeviceNet Communication Module

Reception of the Product

When opening the LULC09 DeviceNet communication module box, you will find two items:
● An Instruction Sheet (IS), giving brief and illustrated information about the basic installation of a module.
● An LULC09 DeviceNet communication module equipped with connectors.

NOTE: Please check that you do have all items described above. You must have the IS, and the connectors must be present and correctly positioned.

Product Functions

The communication module allows you to control a motor starter remotely, via DeviceNet, from:

A TeSys U starter-controller	LUB•• / LU2B••
A TeSys U starter	LUS•• / LU2S••
A TeSys U controller	LUTM••

With the communication module, you can:
● read the motor starter states,
● control the motor starter (run forward or reverse),
● set the protection functions,
● read the data processed in the advanced and multifunction control units,
● read the input and output states.

⚠ DANGER

IMPROPER CONTROL VOLTAGE

Use 24VDC to power the LULC09 communication module. Do not connect voltage in excess of 24VDC.

Failure to follow these instructions will result in death or serious injury.

Data Available

The available protection and control data depend on the control unit with which the LULC09 DeviceNet communication module is used.

There are three types of control units:
● Standard (referenced as LUCA)
● Advanced (referenced as LUCB/C/D, LUCBT/DT)
● Multifunction (referenced as LUCM, LUCMT)

In order to select the right TeSys U configuration you need, the following table lets you check the data and controls you have access to:

Data - Controls	Configuration		
	Standard (LUCA)	Advanced (LUCB/C/D, LUCBT/DT)	Multifunction (LUCM/MT)
Start and stop controls	√	√	√
States (ready, running, faulty)	√	√	√
Warning		√	√
Automatic reset and remote reset via the bus		√	√
Indication of the motor load		√	√
Differentiation of faults		√	√
Remote parameter setting and viewing of all functions			√
"Statistics" function			√
"Monitoring" function			√

Description and Installation of the Module

Front View of the Module

Connectors and LEDs of the LULC09 DeviceNet communication module are described hereafter:
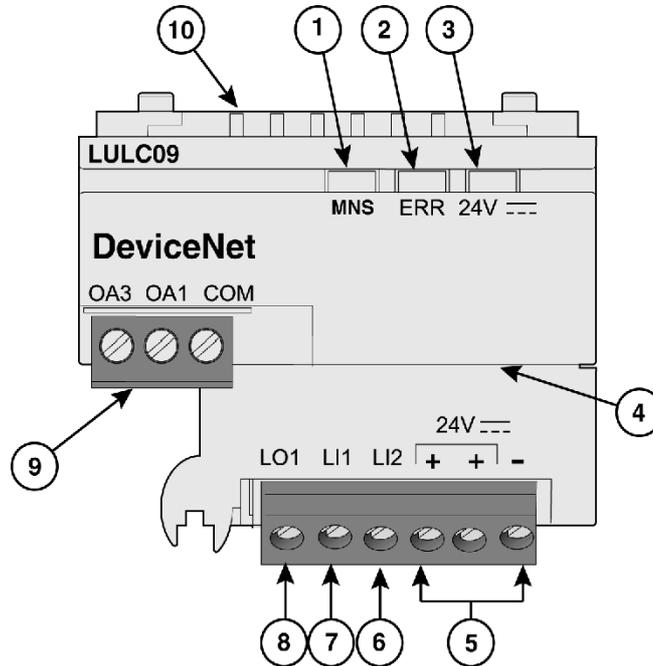
1 2-color MNS LED indicating DeviceNet module operational status.
2 Red ERR LED indicating DeviceNet module fault.
3 Green 24V⎓ LED indicating voltage presence at outputs OA1, OA3, LO1 and 24V Bus.
4 Open-style connector and 24V Bus (DeviceNet external power supply)
5 Connection of the 24V ⎓ power supply for outputs OA1, OA3, LO1 (The 2 terminals marked + are internally linked).
6 Logic input 2.
7 Logic input 1.
8 Logic output 1, assignable depending on configuration reg. 685 (LSB).
9 24V⎓ wiring coil connector for the power base:
   ○ OA1 assignment depends on configuration register 686 (LSB),
   ○ OA3 assignment depends on configuration register 686 (MSB).

10 Connector for communication with the advanced or multifunction control unit

MNS LED Description

The software-controlled **MNS** (Module/Network Status) is a two-color light-emitting diode (LED), alternating between two states: a Run state (green color) and an Error state (red color).

MNS LED colors can be flickering (every 50ms), or blinking (every 200ms), or flashing (1, 2 or 3 flashes), or steady, as described below.

2-color MNS LED	Color display mode	Meaning	Action
Off	-	No power or autobaud detection in progress or no error	-
Flash Green	The LED repeatedly flashes on for 500ms, then off for 500ms	Device is in pre-operational state and no connections have been made.	Check the device configuration OR Wait for the end of auto-addressing
Green	Steady	Operational state. The device is online and connected.	-
Flash Red	The LED flashes on for 500ms, then off for 500ms	A minor fault has occured or one or more I/O connections has timed-out.	Restart communication

2-color MNS LED	Color display mode	Meaning	Action
Red	Critical fault or link failure	The device has detected an error that has made it incapable of communicating on the network (Duplicated MAC-ID or Bus-off).	Check the MAC-ID, then cycle power and restart communication
Flashing Green+Red	The LED repeatedly flashes red for 500ms, then green for 500ms	The device has detected a network access error and is in Communication Faulted state.	Restart communication

MNS LED Blink Patterns

Below is a representation of the **MNS** LEDs, showing the different colors and flashing rates:
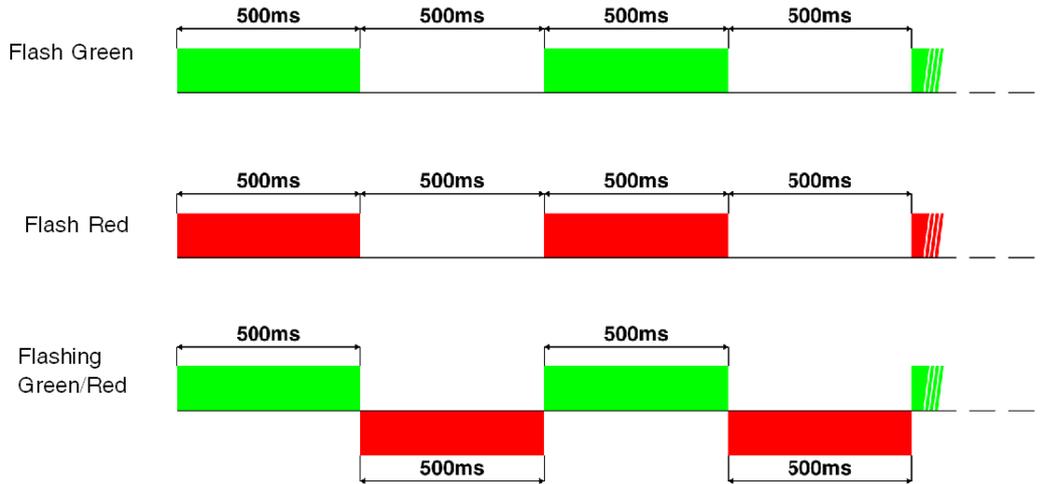
Error (ERR) LED

The signalling is active provided that the communication module is powered up by the Bus (DeviceNet external power supply).

The red error (**ERR**) LED has 3 different states:

Error LED	Meaning	Action
Off	Working condition. No error	-
On	Presence of an internal fault	See *Internal Faults, page 67*
Blinking	Loss of communication. A fallback strategy is in progress	Check the cabling of your DeviceNet network

24V⎓ LED

The green 24V⎓ LED has 2 different states:

Off	24V Bus or 24V⎓ power is missing.
On	LULC09 communication module is correctly powered.

Bottom View of the Module

Here is a bottom view of an LULC09 communication module:
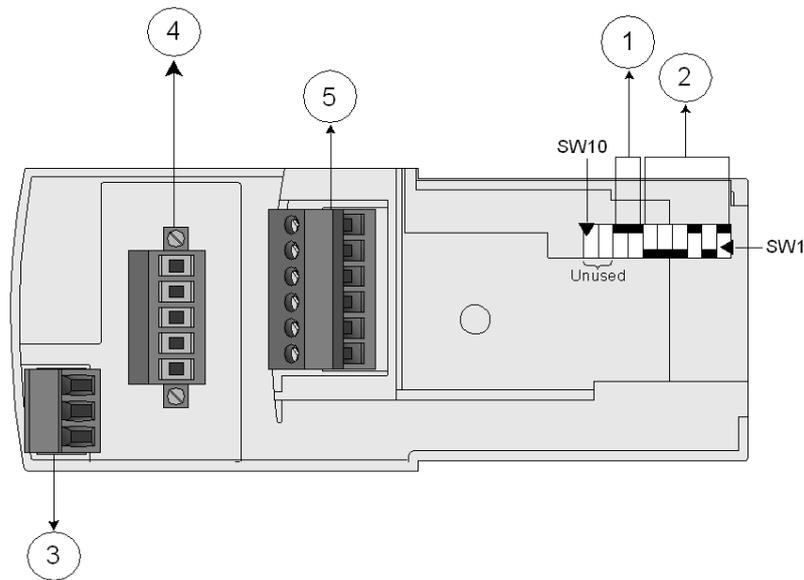
1 Baud rate switches (SW7 and SW8)
2 Address switches (SW1 to SW6)
3 Power base connector
4 DeviceNet bus connector
5 24V and I/O connector

Baud Rate

The system allows you to assign a baud rate (with the following speeds: 125, 250, 500, and AutoBaud), using the switches SW7 and SW8.

The table below shows how to set the baud rate switches to the required speed:

SW8	SW7	Baud Rate
0	0	125 kbps (default value)
0	1	250 kbps
1	0	500 kbps
1	1	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.

Address

The communication module's address on the DeviceNet bus is the MAC-ID. You can assign an address from 0 to 63, using the 6 right-most switches (SW1 to SW6).

Example:
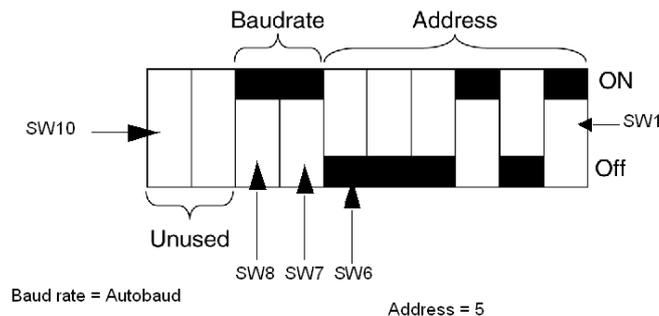
(SW = Switch)

The following table shows examples of address settings (the first 5 settings):

SW6	SW5	SW4	SW3	SW2	SW1	Address
0	0	0	0	0	0	0
0	0	0	0	0	1	1
0	0	0	0	1	0	2
0	0	0	0	1	1	3
0	0	0	1	0	0	4
0	0	0	1	0	1	5

NOTE: The default address is 63.

Assembly Order

The LULC09 DeviceNet communication module is installed in a power base or a controller base, beneath the control unit which locks it in position.

To install the module within the power base or the controller base:

Step	Action
1	Choose the prewired coil connection.
2	Insert the LULC09 DeviceNet communication module.
3	Insert the control unit that locks the module.

The illustration below details the steps. Installation of LULC09 DeviceNet communication module is (2). Numbers correspond both to components assembly order and to their positions.
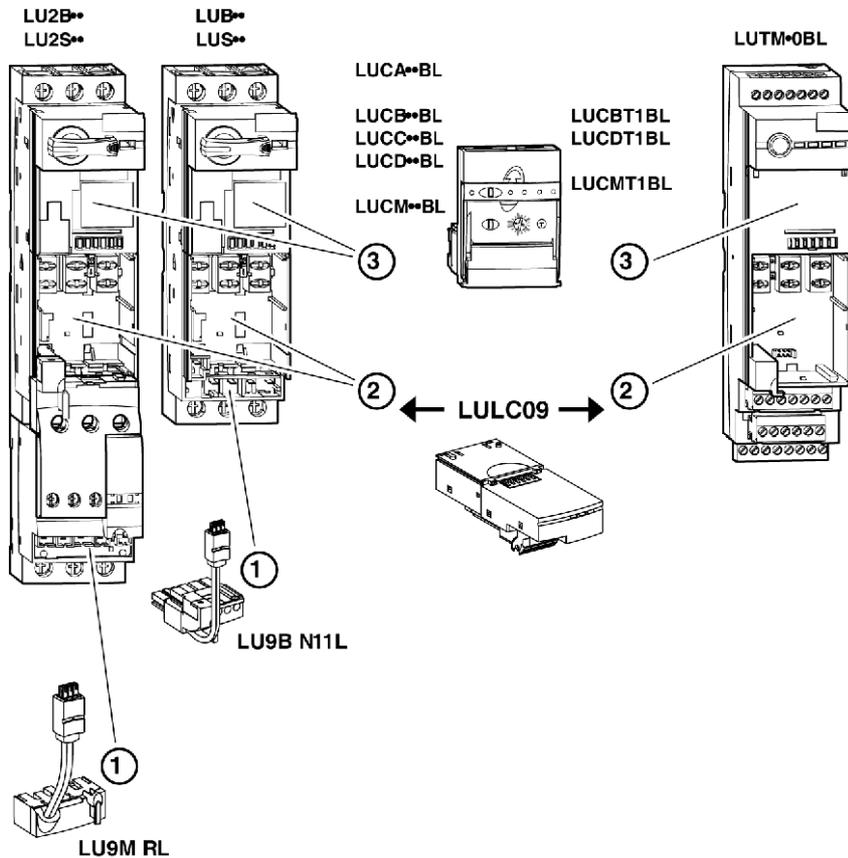
Electrical Connection

24V⎓ and Internal Power Supplies

Schematic of the 24V⎓ (24VDC) and internal power supplies:

Power base   Controller base

24 V Aux → LUCM   LUCMT ← (1)

LUCB
LUCC
LUCD

LUCBT
LUCDT

LUCA

A1/A2
Terminals   LUTM ← 24 V Aux

24 V Bus* → LULC09   LULC09 ← 24 V Bus
Bus DN* →   ← Bus DN
* same cable

24V ⎓   24V ⎓

— — — — Internal links
———— External links (to wire)
(1) A cap prevents external power supply connection.
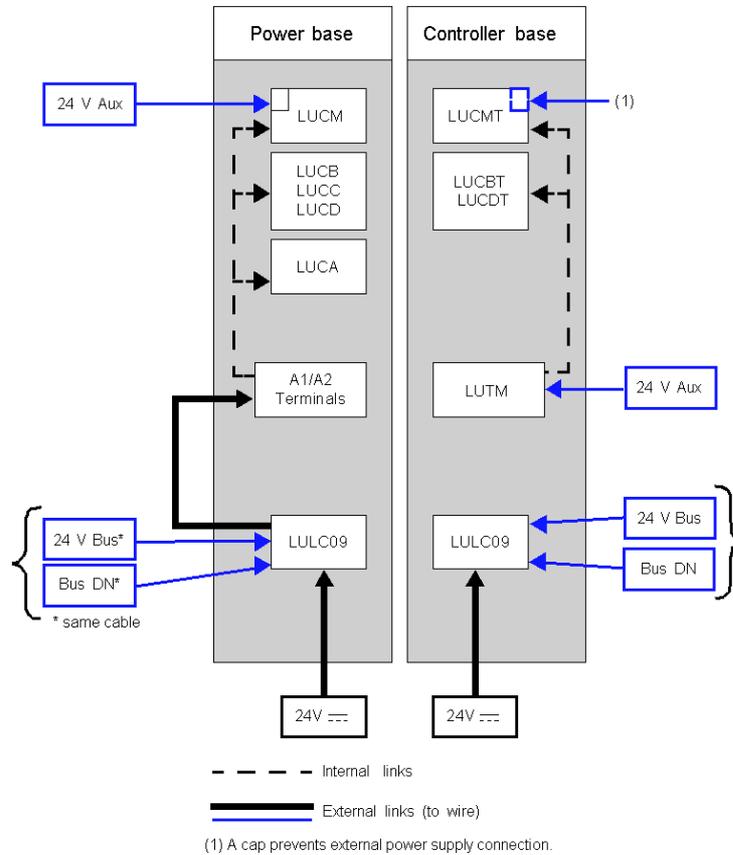
24V Bus = Communication module power supply (V+ and V-)

24V⎓ = Power supply for OA1, OA3 and LO1

24V Aux = Power supply for LUCM control unit or LUTM controller

TeSys U Behaviour at Power-up with an LUCM/LUCMT

Here is a description of the system behaviour at power-up of:
- a power base (LUB/LUS/LU2) with an LULC09 communication module and LUCM control unit,
- a controller base (LUTM) with an LULC09 communication module and LUCMT control unit.

LULC09 Internal Power Supply	LUCM 24V Aux	A1/A2 Local Control	Comment
√	√		When LULC09 comm. module and LUCM multifunction control unit are powered-up simultaneously (recommended), the system is ready.
√			LULC09 is waiting for LUCM identification. The motor starter is not seen by the DeviceNet island. The ERR LED is on (steady red).
	√ or √		LUCM is waiting for LULC09, which provokes an **M15** fault that must be acknowledged via the LUCM keypad or via the bus, once LULC09 has been powered-up.

LULC09 Internal Power Supply	LUCMT	Comment
√	√	When LULC09 comm. module and LUCMT multifunction control unit are powered-up simultaneously (recommended), the system is ready.
√		LULC09 is waiting for LUCMT identification. The motor starter is not seen by the DeviceNet island. The ERR LED is on (steady red).
	√	LUCMT is waiting for LULC09, which provokes an **M15** fault that must be acknowledged via the LUCMT keypad, via the bus, or via the LUTM push-button, once LULC09 has been powered-up.

Power Supply for the LULC09 and Outputs OA1, OA3 and LO1

To operate, the LULC09 DeviceNet communication module must be powered by a 24V⎓ internal power.

LUB•• / LUS•• / LU2B•• / LU2S•• Power bases power-up
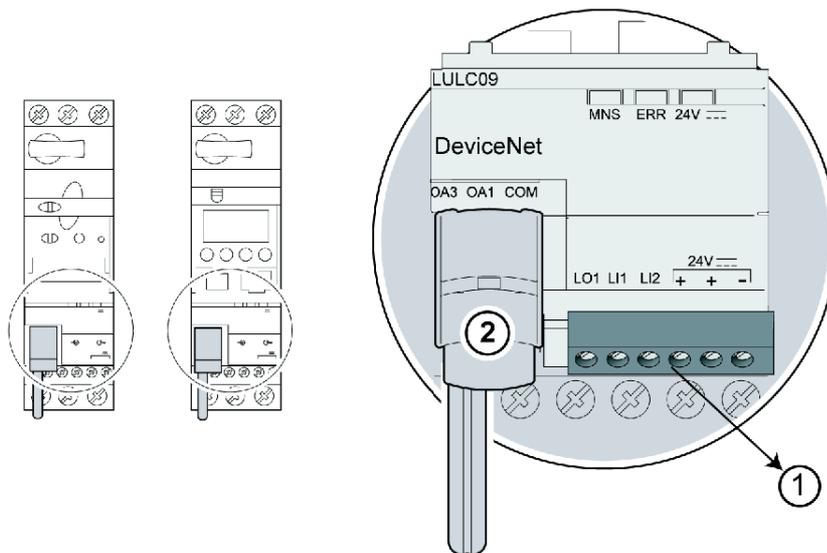
1 24V=: power supply terminal for outputs OA1, OA3 and LO1
2 Prewired coil connection for outputs OA1 and OA3 to terminals A1/A3/A2 on the starter

LUTM Controller bases power-up
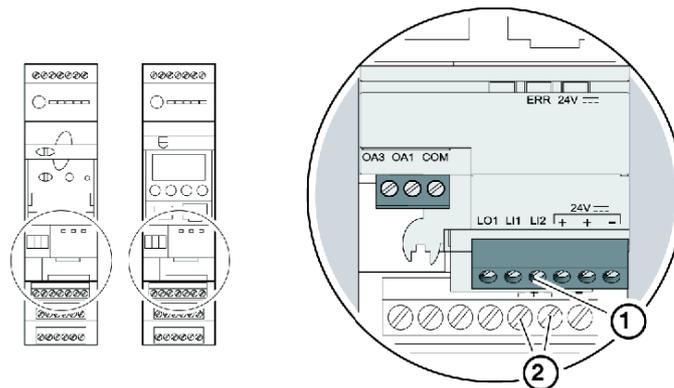
1 24V=: power supply terminal for outputs OA1, OA3 and LO1 (if required)
2 24V power supply terminals for LUTM

Power base: Coil Terminals Power-up

You have 2 options for connecting the power base coil terminals:
● power supply via the LULC09 DeviceNet communication module with a prewired link,
● direct power supply with a wire-to-wire link.

1744085 07/2019

Prewired link

References of the 2 prewired coil connections:

Description	with a Power Base	Reference
Prewired Coil Connection	LUB•• / LUS••	LU9B N11L
	LU2B•• / LU2S••	LU9M RL

Illustrations for both types of power bases:
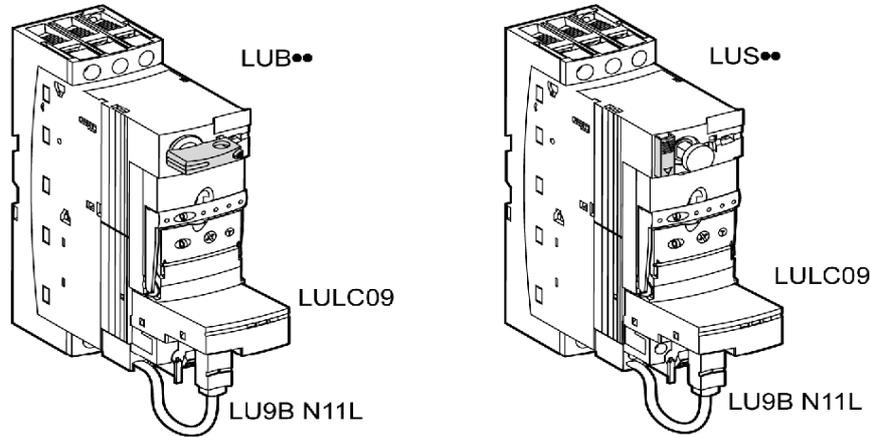
Wire-to-wire link (supplying power from outputs OA1 and OA3 to coil terminals A1, A2, and A3)

This type of link is compulsory in the case of a reversing starter-controller created from a separate **LU6M** reverser block.

The wire-to-wire link is also used to insert, for example, a local control or an external stop control.

LULC09 Connection Cross-Sections

The following table shows the conductor cross-sections that may be used on LULC09 terminals:

Connection	Conductor type	Cross-section (min. - max.)	
1 conductor	Solid conductor	0.14 ... 1 mm^2	26 ... 18 AWG
	Stranded conductor	0.14 ... 1 mm^2	26 ... 18 AWG
	Stranded conductor with cable end: - non-insulated - insulated	0.25 ... 1 mm^2 0.25 ... 0.5 mm^2	24 ... 18 AWG 24 ... 20 AWG
2 conductors (same cross-section)	2 solid conductors 2 stranded conductors	0.14 ... 0.5 mm^2 0.14 ... 0.75 mm^2	26 ... 20 AWG 26 ... 20 AWG
	2 stranded conductors with cable end: - non-insulated - insulated	0.25 ... 0.34 mm^2 0.75 mm^2	24 ... 22 AWG 20 AWG

Connectors	3 and 6 pins	
Pitch	3.81 mm	
Tightening torque	0.2 / 0.25 N.m.	28.3 / 35.4 lb-in.
Flat screwdriver	2.5 mm	0.10 in.

DeviceNet Connection Cross-Sections

The following table shows the conductor cross-sections that may be with the DeviceNet open-style connector:

Connection	Conductor type	Cross-section (min. - max.)	
1 conductor	Solid conductor	0.2 ... 2.5 mm^2	24 ... 12 AWG
	Stranded conductor	0.2 ... 2.5 mm^2	24 ... 12 AWG
	Stranded conductor with cable end: - non-insulated - insulated	0.25 ... 2.5 mm^2 0.25 ... 2.5 mm^2	24 ... 12 AWG 24 ... 12 AWG
2 conductors (same cross-section)	2 solid conductors	0.2 ... 1 mm^2	24 ... 18 AWG
	2 stranded conductors	0.2 ... 1.5 mm^2	24 ... 16 AWG
	2 stranded conductors with cable end: - non-insulated - insulated	0.2 ... 1 mm^2 0.5 ... 1.5 mm^2	24 ... 18 AWG 20 ... 16 AWG

Connectors	5 pins	
Pitch	5.08 mm	0.12 in.
Tightening torque	0.5 - 0.6 N.m.	4.42 - 5.31 lb-in.
Flat screwdriver	3.5 mm	0.14 in.

Connection to the DeviceNet Bus

General Characteristics of a DeviceNet Connection

The following table provides general characteristics of a connection to the DeviceNet bus:

Characteristics	Description
Type of communication protocol	ODVA: DeviceNet specification Vol 1, Release 2.0 and Vol 2, Release 2.0.
Type of hardware interface	CAN 2.0 A (2.0 B passive)
Baud rate	125 - 500 kbaud
Maximum connection distance	Depending on the baud rate (see *Maximum Network Lengths, page 24*).
Maximum number of slaves connected to 1 master	63
Connector type	Open-style terminal connector
Cable structure	2 pairs with separate shielding and a different gauge. Shielding is aluminium foil + tinned copper braid + drain.
EMC protection	*See the TSX DG KBL F Guide: Electromagnetic compatibility of industrial networks and fieldbuses.*

Electrical Interface

The DeviceNet bus uses two twisted pairs to transmit:
● the differential communication signals CAN_H and CAN_L
● the communication module power supply V+ and V-
● the connection of a shield cable
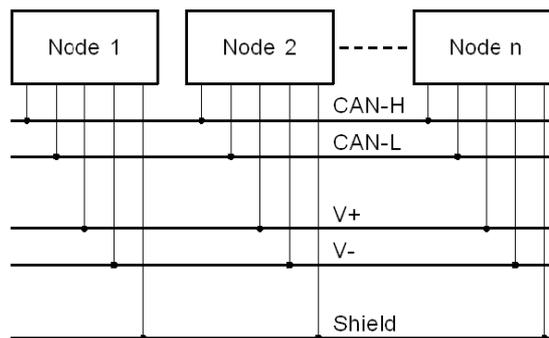
DeviceNet Open-Style Connector

The bus connector is a five-pin connector with pins spaced 5.08 mm (0.2 in.) apart, which can be used with standard DeviceNet thick and thin cables.

The following diagrams show the connection of a DeviceNet cable to a DeviceNet communication module:
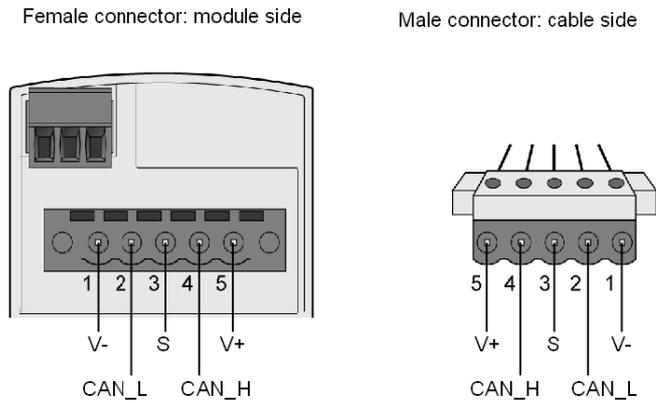
Standard DeviceNet thin cable must be used to connect TeSys U on the network.

Each TeSys U DeviceNet communication module allows interconnection of the following signals:

Pin number	Signal	Description	Wire colour
1	V-	Common 24V bus	Black
2	CAN_L	CAN Low bus conductor	Blue
3	S	Shield	(none)
4	CAN_H	CAN High bus conductor	White
5	V+	+ 24V bus	Red

DeviceNet Network Connection

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 network must be terminated at each end with 121 Ω resistors.

A sample DeviceNet network topology is shown in the following figure:
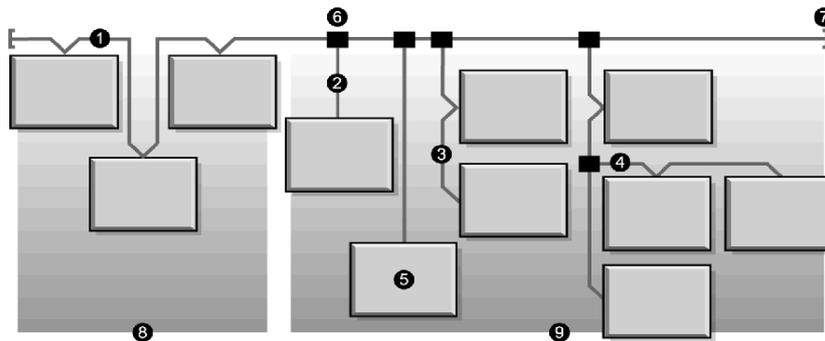
1	trunk line
2	drop line - 0 to 6 m (0 - 19.7 ft.)
3	daisy chain drop-off
4	branched drop-off
5	network node
6	trunk line tap junction
7	terminating resistor
8	zero drop
9	short drops

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.

Refer to ODVA litterature for information on how to design and install your complete DeviceNet installation.

Thin cable must be used to connect TeSys U on the DeviceNet network. If a complete daisy chaining solution is selected, the total length of the daisy chain must not exceed 100 meters without repeaters. You can connect a maximum of 63 TeSys U devices on the DeviceNet network if the total data size exchanged through I/O messages doesn't exceed the capacity of the DeviceNet scanner.

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 kbits/s	250 kbits/s	500 kbits/s
Thick Trunk	500 m	250 m	100 m
Thin Trunk	100 m	100 m	100 m
Flat Trunk	420 m	200 m	75 m
Maximum Drop Length	6 m	6 m	6 m
Cumulative Drop Length*	156 m	78 m	39 m
*The sum of the length of all drop lines.			

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 will be 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.

Example of Architecture with TeSys U
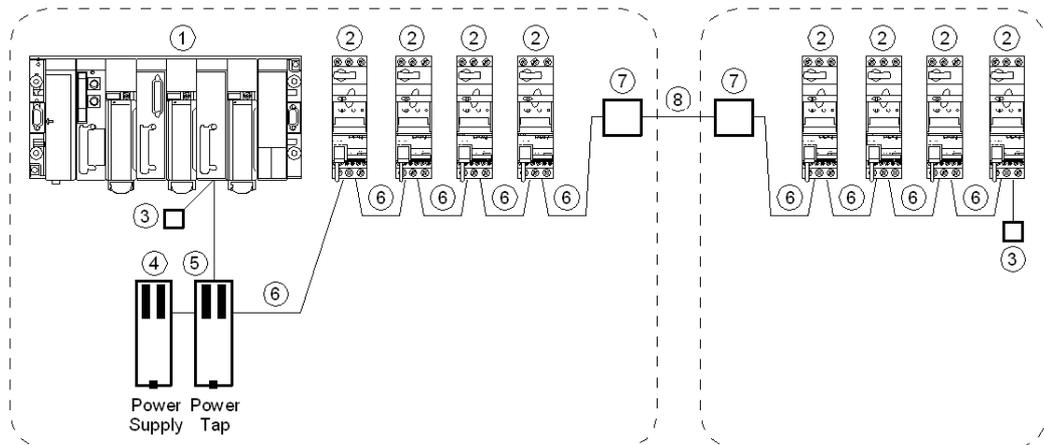
1   PLC with DeviceNet scanner
2   TeSys U with DeviceNet communication module LULC09
3   Line terminators (121 Ω resistance)
4   24 VDC power supply
5   DeviceNet power tap
6   DeviceNet thin cable
7   DeviceNet open-style connnector tap
8   DeviceNet thick cable

Architecture Constraints with TeSys U and One Power Supply

The minimum distance between two TeSys U DeviceNet devices is 0.3 m (11.8 in.).

In the case of a complete thin cable daisy chain architecture using one single power supply, the 2 curves below indicate the manximum number of TeSyS U devices that can be connected, depending on the bus length and temperature.
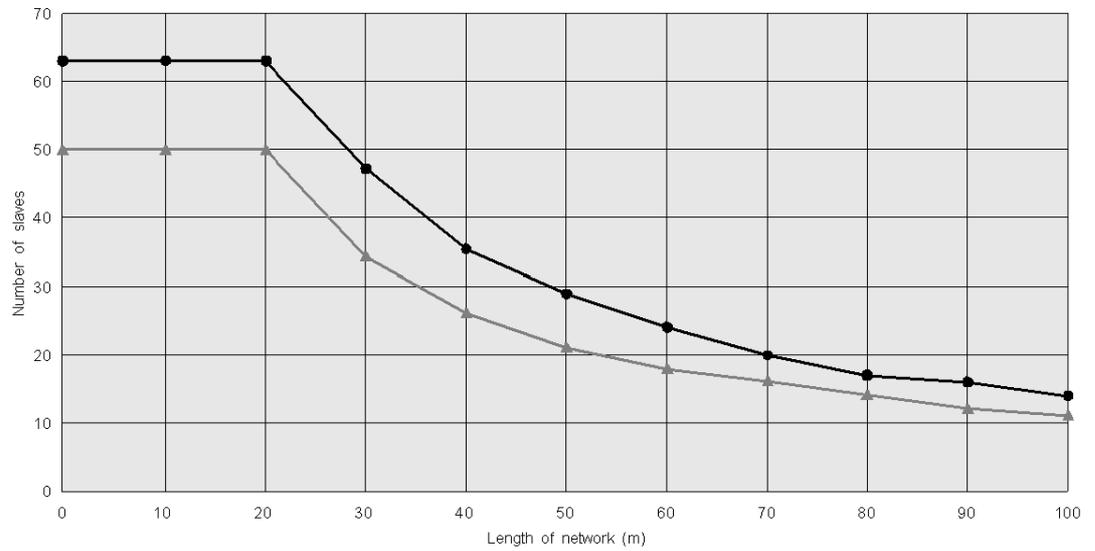
The number of TeSys U devices connected on the DeviceNet network can be increased if necessary by using a second DeviceNet power supply.

Chapter 2
Technical Characteristics

Service Conditions and Technical Characteristics

LULC09 DeviceNet communication module characteristics include:
- Service conditions
- 24V ⎓ and 24V internal power supply circuit characteristics
- Logic outputs (OA1, OA3 and LO1) and logic inputs (LI1 et LI2) characteristics.

Communication characteristics (module port) are also described.

Service Conditions

LULC09 DeviceNet communication module service conditions are:

Certification	UL, CSA	
Conformity to standards	IEC 62026-1 Overvoltage category III Degree of pollution: 3	
European Community Directives	CE marking. In conformity with the essential requirements of low voltage (LV) equipment and electromagnetic compatibility (EMC) directives.	
Ambiant air temperature around the device	Storage	- 40 ... + 85 °C (- 40 ... + 185 °F)
	Operation	- 25 ... + 85 °C (- 13 ... + 131 °F)

Product Dimensions

Dimensions of an LULC09 DeviceNet communication module are:

	LULC09
H x L x D	42 x 45 x 108 mm (1.6 x 1.8 x 4.25 in.)
Weight	94.4g (0.21lb)

For information about dimensions of overall TeSys U products, see *"Motor starters - open version TeSys U" Catalogue.*

24V ⎓ Power Supply Circuit

Technical characteristics of 24V ⎓ power supply circuit, for LULC09 DeviceNet communication module, include:

Supply voltage	$U_{nominal}$	V	24V ⎓
	Operating range	V	20 ... 28
Maximum current drawn		A	1.5 at +55 °C (131 °F)
Resistance to micro cuts		ms	3
Protection	against overvoltage		Yes
	against reverse polarity		Yes

24V Bus

Technical characteristics of 24V bus, for LULC09 DeviceNet communication module, include:

Supply voltage	$U_{nominal}$	V	24V ⎓
	Operating range	V	20 ... 28
Maximum current drawn		A	0.06
Resistance to micro cuts		ms	3
Protection	against overvoltage		Yes
	against reverse polarity		Yes

OA1, OA3 and LO1 Logic Outputs

Output characteristics of an LULC09 DeviceNet communication module include:

Nominal output values	Voltage	V	24V ⎓
	Current	mA	500
Output limit values	Voltage	V	20 ... 28
	Current	mA	700
Coincidence factor of the 3 outputs		%	100
Output response time (register 704) (Time duration between the request start bit and the change in the output state)		ms	<10 (OA1, OA3, LO1)
Protection	Against short-circuits and overloads		Electronic circuit-breaker with automatic reset
Number of operating cycles	In millions		15
Maximum rate	In operating cycles per hour		3600

LI1 and LI2 Logic Inputs

Intput characteristics of an LULC09 DeviceNet communication module include:

Nominal input values		Voltage	V	24V ⎓ (positive logic)
		Maximum voltage	V	28V ⎓
		Current	mA	7
Input limit values	State 1	Voltage	V	16
		Current	mA	6
	State 0	Voltage	V	5
		Current	mA	2
Response time	To state 1		ms	10 +/- 30%
	To state 0		ms	10 +/- 30%
Input type				Resistive
Protection	gl fuse		A	1

Communication (DeviceNet Port)

Technical characteristics of DeviceNet port for the LULC09 communication module include:

Physical interface	1	CAN
Connector		Open-style connector
Pinout		According to the DeviceNet specification (see *DeviceNet Open-Style Connector, page 22*).
Protocol		CAN 2.0 and CAN 2.B (passive mode)
Address	Range	1 to 63 (default = 63)
Transmission speed	Kbit/s	125 (default), 250, 500, or Autobaud

Part II
Software Implementation

Hardware implementation of an LULC09 DeviceNet communication module is being followed by its software implementation. It is focused on configuration (the different operating modes) and functions setting (e. g. protection, current measurement, ...).

What Is in This Part?

This part contains the following chapters:

Chapter 3
DeviceNet Communication Module Management

Introduction to DeviceNet Bus

A TeSys U motor starter equipped with the LULC09 communication module is managed via the DeviceNet bus. This chapter describe basic DeviceNet features, as well as the DeviceNet module's object dictionary.

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 communications link between simple industrial devices (such as actuators and sensors) and controlling devices. The network transports control information, as well as information about the properties of the device being controlled. It enables operation in either master/slave or peer-to-peer mode.

The DeviceNet powered four-wire network operates in a trunkline-dropline configuration and supports up to 64 nodes.

Connections and Data Exchange

Connections

Connections are established through the motor starter's communication terminals. See *DeviceNet Open-Style Connector, page 22* for more information.

Messaging

The following connection types are established with DeviceNet's connection-based model:

- *explicit messaging connections*—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.
- *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.

Idle Message Management

When the DeviceNet module receives an Idle Message sent by the DeviceNet network master, it generates a communication loss and the DeviceNet module is in fallback mode *(see page 53)*.

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

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 TeSys U Motor Starter profile, see *Object Dictionary, page 75*.

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 75)*, 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.

EDS Download Procedure

The different TeSys U starter-controller variants are described in EDS (Electronic Data Sheet) files.

If the TeSys U starter controllers do not show up in your CANopen configuration tool, the corresponding EDS files must be imported.

The following table describes the steps to follow to download the EDS and icon files associated to Tesys U from the www.schneider-electric.com website:

Step	Action
1	Open the Schneider Electric website: www.schneider-electric.com.
2	Click **Products and Services** and then click **Automation and Control**.
3	In the **Downloads** section of the left menu bar, click **Current offers**.
4	- In the **Choose a function** drop-down list, select **Motor Control**. - In the **Choose a range** drop-down list, select **TeSys U**. - In the **Choose a type of document** drop-down list, select **Software/Firmware**. Click **>Find**.
5	Select **Communication Module Tesys U Canopen** and download LULC09_EDS_DIB_files_V100.exe file.
6	Double-click LULC09_EDS_DIB_files_V100.exe on your hard disk. Click **Accept** in the 'Licence for software downloaded from Schneider-Electric web sites' window which opens, then browse for a destination folder and click **Install**.
7	Select the EDS file(s) corresponding to your TeSys U configuration(s):

For information on how to register these EDS files in the RSNetWorx's EDS library system, see *Register the Motor Starter EDS, page 37*.

The table below gives the associations between the 7 TeSys U variants and the associated EDS files names.

Variants names	EDS file name
TeSys U C Ad	TE_TESYSU_C_AD••••E.eds
TeSys U C Mu L	TE_TESYSU_C_MU_L••••E.eds
TeSys U C Mu R	TE_TESYSU_C_MU_R••••E.eds
TeSys U Sc Ad	TE_TESYSU_SC_AD••••E.eds
TeSys U Sc Mu L	TE_TESYSU_SC_MU_L••••E.eds
TeSys U Sc Mu R	TE_TESYSU_SC_MU_R••••E.eds
TeSys U Sc St	TE_TESYSU_SC_ST••••E.eds

● **Sc** and **C** letters stand for **S**tarter-**C**ontroller and **C**ontroller, respectively.
● **St**, **Ad** and **Mu** letters stand for **St**andard, **Ad**vanced and **Mu**ltifunction control unit, respectively.
● **R** and **L** letters stand for **R**emote and **L**ocal configuration.

Local configuration is supported by Multifunction Control Unit >= V3.x.

For more information on TeSys U variants, see *Selection Criteria of a TeSys U Variant, page 48*.

Configuring the Motor Starter with Configuration Software

Introduction

Use these sample instructions to configure an Allen Bradley SLC-500 PLC (1747-SDN) with a DeviceNet controller at the head of a TeSys U Motor Starter equipped with a multifunction control unit. The configuration software is Rockwell's 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 37)*
2	Register the controller's EDS files *(see page 37)*
3	Connect devices to your network *(see page 37)*
4	Upload the controller configuration *(see page 40)*
5	Add the controller to the Scanlist *(see page 40)*

Before You Begin

Before you begin, make sure:
- the TeSys U Motor Starter is fully assembled, installed, and powered according to your particular system, application, and network requirements.
- you have properly set the node address and baud rate of the controller. See *Bottom View of the Module, page 15* for more information.
- you have the basic EDS files 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 motor starter using RSNetWorx, you must have a working familiarity with both the DeviceNet fieldbus protocol and RSNetWorx for DeviceNet, version 3.21.00. (The described procedures cannot practically anticipate every RSNetWorx 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 motor starter, and RSNetWorx:
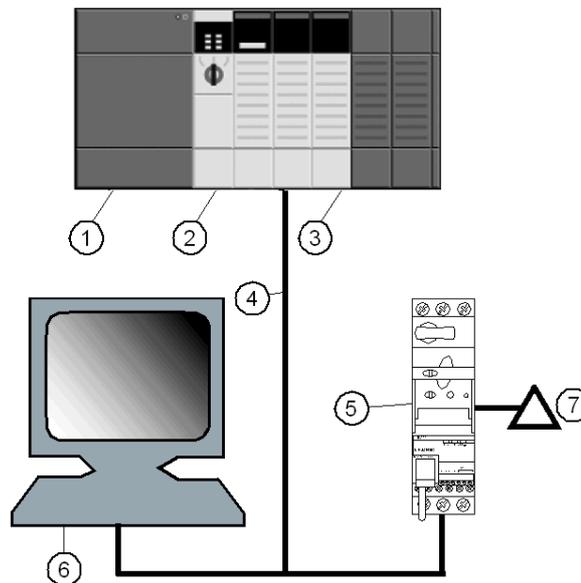
1 Allen-Bradley SLC-500 PLC
2 PLC processor module
3 1747-SDN DeviceNet scanner module
4 DeviceNet network cable
5 TeSys U Motor Starter
6 PC running RSNetWorx (properly connected to your network)
7 DeviceNet power supply

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
⚠ **WARNING**		
UNINTENDED EQUIPMENT OPERATION		
The application of this product requires expertise in the design and programming of control systems. Only persons with such expertise should be allowed to program, install, alter, and apply this product.		
Follow all local and national safety codes and standards.		
Failure to follow these instructions can result in death, serious injury, or equipment damage.		
⚠ **CAUTION**		
INSTALLATION UNDER POWER		
Disconnect all power to the PLC before making the network connection.		
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 36)* above shows the scanner in slot 2 of the PLC.
2	Check that the desired DeviceNet network node address and baud rate have been correctly set. See *Description and Installation of the Module, page 13*.	This example uses an address of *15*.
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 TeSys U Motor Starter with the DeviceNet cable.	
5	Connect the RSNetWorx PC to the network using the DeviceNet cable.	

Register the Motor Starter EDS

To register the motor starter EDS in RSNetWorx's EDS library, follow the procedure in the following table:

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 file(s) 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.
6	Click Next.	The Change Graphic Image screen appears. The TeSys U 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 device has been registered and click Next.	The completion screen appears.
9	Click Finish.	The EDS Wizard closes.

Connect Devices to Your Network

This example requires you to add two devices to your project view:
- a starter-controller up to 15 KW with a multifunction control unit configured in remote mode with the address 15.
- 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 EDS named TeSys U Sc Mu 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 motor starter graphic.	The motor starter's properties window appears.
3	Change the MAC ID in the Address text field to *15*.	*15* is the MAC ID used throughout this example.
4	Click OK.	Note that the MAC ID of the motor starter is now *15* 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 Online from the Network 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.	When the Browsing network screen finishes, the physically connected devices will appear in the project view.
3	Click OK to transfer the required device information.	

The RSNetWorx Project View

The RSNetWorx project view should resemble the following figure after you've added the motor starter and the master scanner to your network configuration (using either the online or offline connection procedure):
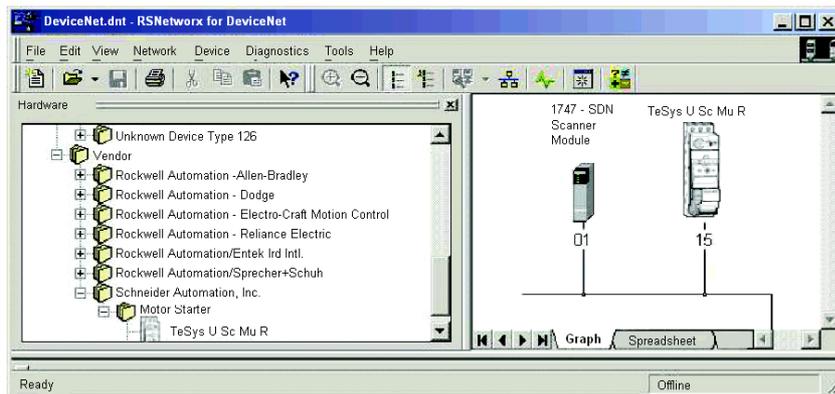
Read and Write TeSys U Motor Starter Parameters

To read and write to the controller's parameters:

Step	Action	Comment
1	From the project view, double -click on the motor starter icon.	The motor starter configuration screen appears
2	Select the Parameter tab.	The parameter lists appears.
3	Select Group View	The parameter groups appear.
4	Select the configuration group to access motor starter configuration parameters. Select Setting group to access motor starter setting parameters.	See *TeSys U Communication Variables User's Manual* for a complete list of configuration and setting parameters.
5	Select the parameter you want to access and read from or write to it.	Write access to parameters is not available with the variants TeSys U Sc Mu L and TeSys U C Mu L, which preserve local configurations.

The TeSys U Motor Starter Parameter Screen

The TeSys U Motor Starter parameter screen should resemble the following figure:
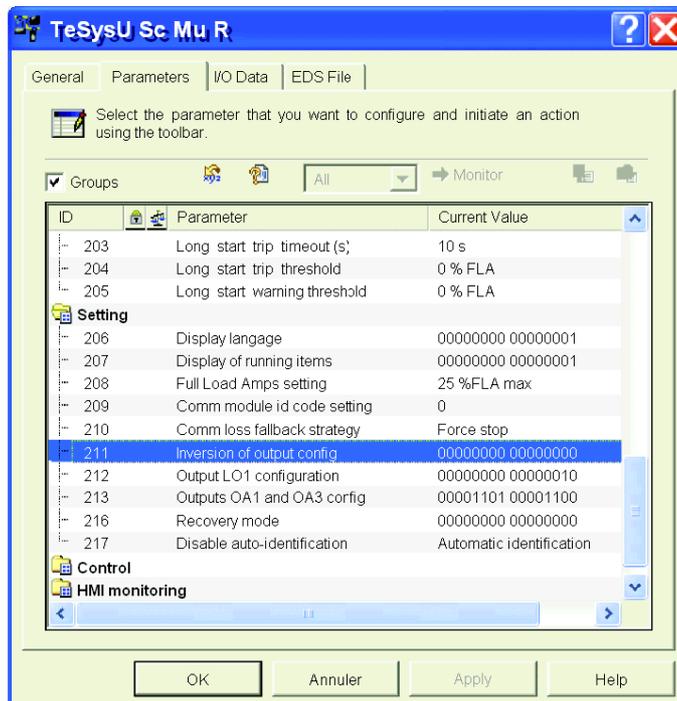
When you select a parameter (selected parameters are highlighted in blue), you can press the keys Ctrl + H to access a Help screen that provides the register number:
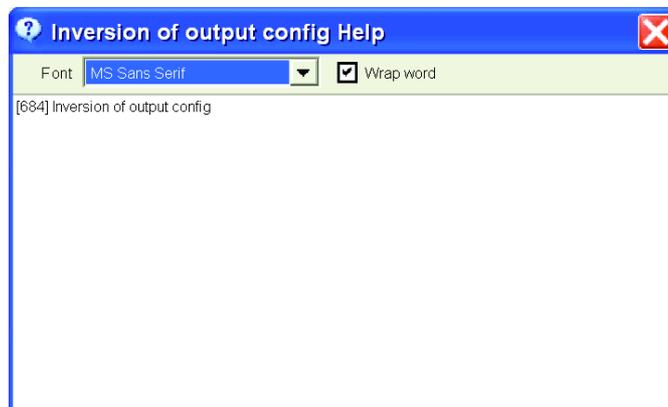
Select Data Exchanged via I/O Messaging

To select data exchanged through I/O messaging

Step	Action	Comment
1	In the TeSys U Sc Mu R parameter screen, select DeviceNet Interface Group	The parameter list appears.
2	For the PollProdPath parameter, select the input assembly object you want the motor starter to produce.	PollProdPath consists of data produced by the motor starter 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 motor starter to produce.	COSProdPath consists of data produced by the controller on Change-of-State (COS).
5	If you selected input assembly object 110 or 113 in steps 2 or 4, adjust the Monitoring Word 0 to 3 to the register you want the controller to produce.	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 Motor Starter to the Scanlist

To be recognized on the network, the motor starter 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.
4	At the Scanlist tab, highlight the motor starter (at MAC ID *15*) in the Available Devices list, and click the right arrow.	The motor starter now appears in the Scanlist.
5	With the motor starter selected, click the Edit I/O Parameters button.	The Edit I/O Parameters window appears.
6	Check Polled and enter *1* in the Input Size text field and *1* in the Output Size text field.	These are the default data sizes. (Determination of the device's input and output data lengths 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 motor starter's Edit I/O Parameters screen should resemble the following figure after you have customized it as described above:
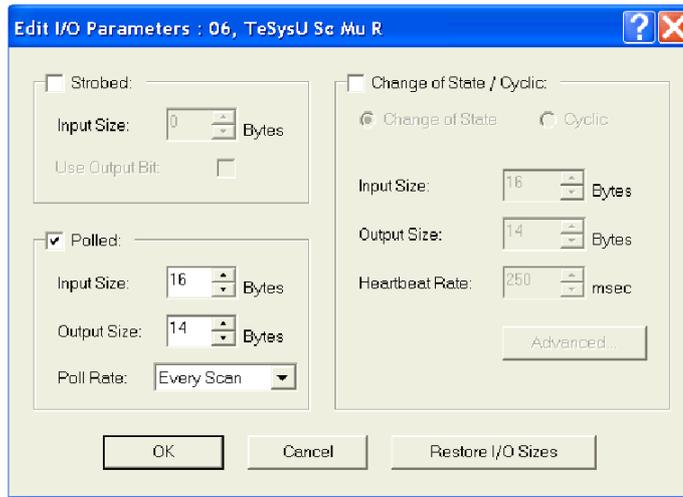
Depending on your requirements, you can select one of three transmission modes:
- Polled
- Change of State
- Cyclic

NOTE: LULC09 does not support Strobed I/O messages used for very simple I/O devices.

About the Motor Starter's Input and Output Data Lengths

In the above procedure, you entered the number of input and output bytes produced by the motor starter. The master device needs this information to allocate data space for each network node.

The number of input and output bytes the motor starter produces depends on the instances you select for the DeviceNet Interface object and the TeSys U variant.

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

Output assembly data size (consumed by the motor starter):

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

Input assembly data size (produced by the motor starter):

Instance	Name	TeSys U variant	Number of bytes
50	Basic Overload	All	1
51	Extended Overload	All	1
52	Basic Motor Starter	All	1
53	Extended Motor Starter 1	All	1
54	Extended Motor Starter 2	All	1

Instance	Name	TeSys U variant	Number of bytes
110	TeSys U Monitoring Registers (with dynamic configuration)		
		TeSys U Sc St	6
		TeSys U Sc Ad	7
		TeSys U Sc Mu R/L	8
		TeSys U C Ad	8
		TeSys U C Mu R/L	8
111	PKW Response Object	All	8
112	PKW Response and Extended Motor Starter	All	10
113	PKW Response and TeSys U Monitoring Registers		
		TeSys U Sc St	14
		TeSys U Sc Ad	15
		TeSys U Sc Mu R/L	16
		TeSys U C Ad	18
		TeSys U C Mu R/L	16

I/O Messages Description

I/O Messages are used to exchange periodic I/O data between the PLC and the TeSys U Motor Starter. The tables below describe the data exchanges, depending on the instance selected in the Assembly Object .

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: TeSys U Control Registers

This assembly contains several control registers commonly used with a TeSys U 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

The following table gives a description of the **Control Register 704:**

⚠ **WARNING**

AUTOMATIC RESTART OF THE MOTOR

The motor will automatically restart if the control bits 704.0 or 704.1 were not previously overwritten to zero by the PLC application, in case of a cyclic writing to register 704 and on the occurrence of one of the following events:

● Loss followed by restoration of the outputs power supply 24 VDC.
● Change in position of rotary knob on power base followed by return to Ready position.
● Communication break followed by restoration.

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

Word 1	DeviceNet path 6C : 1 : 5	Sc St	C Ad Sc Ad	C Mu L/R Sc Mu L/R
bit 0	Run forward	√	√	√
bit 1	Run reverse	√	√	√
bit 2	(Reserved)			
bit 3	Fault reset: if register 451=102 or 104, fault acknowledgment causes a return to communication module factory settings This bit is active on rising edge and must be reset to 0 by programming	√	√	√
bit 4	(Reserved)			
bit 5	Launch automatic thermal overload fault test This bit is active on rising edge and must be reset to 0 by programming			√
bit 6-11	(Reserved)			
bit 12	Launch trip test via communication bus This bit is active on rising edge and must be reset to 0 by programming			√
bit 13-15	(Reserved)			

NOTE: Fault reset bit must be set to 1 and reset to 0 to acknowledge a fault.

The following table gives a description of the Control of Communication Module, **Register 703:**

Word 2	DeviceNet path 6C : 1 : 4	Sc St	C Ad Sc Ad	C Mu L/R Sc Mu L/R
bit 0-2	(Reserved)			
bit 3	Reset warning (communication loss,...) This bit is active on rising edge and must be reset to 0 by programming	√	√	√
bit 4-15	(Reserved)			

NOTE: Reset warning bit must be set to 1 and reset to 0 to acknowledge a warning (loss of communication).

The following table gives a description of the Output Control, **Register 700**:

Word 3	DeviceNet path 6C : 1 : 1	Sc St	C Ad Sc Ad	C Mu L/R Sc Mu L/R
bit 0	Control of output LO1 (if 685=2)	√	√	√
bit 1	Control of output OA1 (if 686 LSB=2)	√	√	√
bit 2	Control of output OA3 (if 686 MSB=2)	√	√	√
bit 3-15	(Reserved)			

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
See *PKW Objects, page 58* for details.							

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 TeSys U 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*	*Reserved*

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: TeSys U Monitoring Registers (with dynamic configuration)

This assembly contains several monitoring registers commonly used with a TeSys U device.

Conf.	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Sc St	Path: 68:01:06 (Register [455])		Path: 68:01:09 (Register [458])		Empty[1]		Empty[1]	
Sc Ad	Path: 68:01:06 (Register [455])		Path: 68:01:09 (Register [458])		Path: 68:01:0C (Register [461])		Empty[1]	
Sc Mu L/R	Path: 68:01:06 (Register [455])		Path: 68:01:09 (Register [458])		Path: 68:01:0C (Register [461])		Path: 68:01:08 (Register [457])	
C Ad	Path: 68:01:06 (Register [455])		Path: 68:01:09 (Register [458])		Path: 68:01:0C (Register [461])		Path: 68:01:0A (Register [459])	
C Mu L/R	Path: 68:01:06 (Register [455])		Path: 68:01:09 (Register [458])		Path: 68:01:0C (Register [461])		Path: 68:01:0A (Register [459])	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB

[1] Any byte left vacant at the end of this assembly will not be sent to the bus; so if no parameter is attributed to bytes 6 and 7, the assembly data length will be 6 bytes instead of 8 bytes.

The following table gives a description of the **Status Register 455** of the starter-controller

Word 1	DeviceNet path 68 : 1 : 6	Sc St	Sc Ad	Sc Mu L/R
bit 0	Ready: LUB••/2B•• = the rotary handle is turned to 'On' position and there is no fault. LUS••/2S•• = the push-button is pressed and there is no fault.	√	√	√
bit 1	Pole status: closed	√	√	√
bit 2	All faults	√	√	√
bit 3	All warnings	√	√	√
bit 4	Tripped: LUB••/2B•• = the rotary handle is turned to 'Trip' position. LUS••/2S•• = the push-button is depressed.	√	√	√
bit 5	Fault reset authorized		√	√
bit 6	A1/A2 terminals powered up			√
bit 7	Motor running with detection of current, if greater than 10% FLA		√	√
bit 8-13	Average motor current: 32 = 100% FLA 63 = 200% FLA		√	√
bit 14	(Non significant)	√	√	√
bit 15	Start in progress: 1 = ascending current is greater than 10% FLA 0 = descending current is lower than 150% FLA		√	√

The following table gives a description of the **Status Register 455** of the Controller

Word 1	DeviceNet path 68 : 1 : 6	C Ad	C Mu L/R
bit 0	Ready: LUTM is powered-on and there is no fault with the communication module (I.7 powered-on)	√	√
bit 1	Input I.3 or I.4 powered-on	√	√
bit 2	All faults	√	√
bit 3	All warnings	√	√
bit 4	Tripped, if thermal overload fault reset mode = manual	√	√
bit 5	Fault reset authorized	√	√
bit 6	I.1 and I.2 powered-on		√
bit 7	Motor running with detection of a current, if greater than 10% FLA	√	√
bit 8-13	Average motor current: 32 = 100% FLA 63 = 200% FLA	√	√
bit 14	In local control	√	√
bit 15	Start in progress: 1 = ascending current is greater than 10% FLA 0 = descending current is lower than 150% FLA For LUCBT/DT, timeout is 10s. For LUCMT, refer to LUCM/MT User's Guide.	√	√

The following table gives a description of the I/O Module Status **Register 458**

Word 2	DeviceNet path 68 : 1 : 9	Sc St	C Ad Sc Ad	C Mu L/R Sc Mu L/R
bit 0	OA1 status	√	√	√
bit 1	OA3 status	√	√	√
bit 2	LO1 status	√	√	√
bit 3-7	(Non significant)			
bit 8	LI1 status	√	√	√
bit 9	LI2 status	√	√	√
bit 10-15	(Non significant)			

The following table gives a description of the Warning **Register 461**

Word 3	DeviceNet path 68 : 1 : C	Sc St	C Ad Sc Ad	C Mu L/R Sc Mu L/R
bit 0-1	Non significant)			
bit 2	Ground fault warning			√
bit 3	Thermal warning			√
bit 4	Long start warning		√	√
bit 5	Jam warning			√
bit 6	Phase imbalance warning			√
bit 7	Under-current warning			√
bit 8-9	Non significant)			
bit 10	Communication loss on LUCMT Modbus port			√
bit 11	Internal temperature warning			√
bit 12	Module identification or internal communication warning			√
bit 13-14	Non significant)			
bit 15	Module warning	√	√	√

The following table gives a description of the I/O Status of the Controller Base, **Register 459**

Word 4	DeviceNet path 68 : 1 : A	C Ad	C Mu L/R
bit 0	I.1 = local control of output 13	√	√
bit 1	I.2 = local control of output 23	√	√
bit 2	I.3 = contactor status on output 13	√	√
bit 3	I.4 = contactor status on output 23	√	√
bit 4	I.5 = input status (reset)	√	√
bit 5	I.6 = input status (external fault)	√	√
bit 6	I.7 = input status (system ready)	√	√
bit 7	I.8 = input status (free)	√	√
bit 8	I.9 = input status (free)	√	√
bit 9	I.10 = input status in local/remote mixed mode if 683=2, otherwise free	√	√
bit 10-11	(Non significant)		
bit 12	Output 13 status (1=O1 closed)	√	√
bit 13	Output 23 status (1=O2 closed)	√	√
bit 14	Outputs 95-96 and 97-98 status (1=95-96 closed and 97-98 opened)	√	√
bit 15	Output 05-06 status (1=05-06 closed)	√	√

The following table gives a description of the Mechanical and Power Supply Status, **Register 457**

Word 4	DeviceNet path 68 : 1 : 8	Sc St	C Ad Sc Ad	C Mu L/R Sc Mu L/R
bit 0	Button position 'On' (0='Off')	√	√	√
bit 1	Button position 'Trip' (0='Not tripped')	√	√	√
bit 2	Contactor state 'On'	√	√	√
bit 3	24 VDC power supply present on outputs	√	√	√
bit 4-15	(Non significant)			

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
See *PKW Objects, page 58* for details.							

Instance 112: PKW Response 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 Response and TeSys U Monitoring Registers

This assembly is vendor specific.

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

Create a Customized EDS File

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

Step	Action	Comment
1	In the project view, double-click the motor starter.	You will be asked if you want to register the motor starter 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 motor starter's identity information, displayed in the Device Description screen.
5	Record the product name string, *TeSysU1*, 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 motor starter has been registered and click Next.	The completion screen appears.
9	Click Finish.	The EDS Wizard closes.
10	Add the motor starter to the Scanlist. See *Add the Motor Starter to the Scanlist, page 40*.	

Saving the Configuration

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

Inserting TeSys U in the DeviceNet Network

Introduction

To insert TeSys U in the DeviceNet network you must select one of the seven variants described below

Selection Criteria of a TeSys U Variant

Choose TeSys U variant...	When you need...
TeSys U C Ad	a controller up to 450kW, for a 3-phase motor class 10-20, with an advanced control unit that protects against overloads and short-circuits, against phase imbalance and insulation breaks, and offers a manual or remote/automatic reset.
TeSys U C Mu R or TeSys U C Mu L	a controller up to 450kW, for a 3-phase motor class 5-30, with a multifunction control unit that protects against overloads and short-circuits, against phase imbalance and insulation breaks, has function warnings, log and monitoring functions, fault differentiation, overtorque and no-load running monitoring, and offers a manual/automatic reset.
TeSys U Sc Ad	a starter or a starter-controller up to 15kW, for a 3-phase motor class 10 or 20 or a 1-phase motor class 10, 0-12 or 0-32A rating, with an advanced control unit that protects against overloads, short-circuits, phase imbalance, and insulation breaks, and offers a manual or remote/automatic reset.
TeSys U Sc Mu R or TeSys U Sc Mu L	a starter-controller up to 15kW, for a 1 phase or a 3-phase motor class 5-30, 0-12 or 0-32A rating, with a multifunction control unit that protects against overloads and short-circuits, against phase imbalance and insulation breaks, has function warnings, log and monitoring functions, fault differentiation, overtorque and no-load running monitoring, and offers a manual/automatic reset.
TeSys U Sc St	a starter or a starter-controller up to 15kW, for a 3-phase motor class 10, 0-12 or 0-32A rating, with a standard control unit that protects against overloads, short-circuits, phase imbalance, and insulation breaks, and offers a manual reset.

Local (L) / Remote (R) configuration modes refer to Configuration register 601 (read/write with motor off), supported by Multifunction Control Unit ≥ V3.x.

If in a local configuration mode...	It means that 601.7 = 1. This mode preserves the local configuration made with the embedded HMI of the multifunction control unit. It forbids any configuration managed by PLC application through the network, thus preserving your local configuration.
If in a remote configuration mode...	It means that 601.7 = 0. This mode enables the PLC application to remotely configure the TeSys U device. **Note:** The parameters overwritten by the PLC application will be lost. This mode is useful in case of faulty device replacement.

By default, the TeSys U device equipped with a Multifunction Control Unit ≥ V3.x is in Remote configuration mode.

Setting TeSys U Parameters

Depending on the TeSys U variant, parameter settings can be managed through different channels:

	Configuration managed by RSNetworx configuration tool	Configuration managed through PKW	Configuration locally by embedded HMI of multifunction control unit
TeSys U C Ad V1.xx	√	√	
TeSys U C Mu L V1.xx			√
TeSys U C Mu R V1.xx	√	√	*
TeSys U Sc Ad V1.xx	√	√	
TeSys U Sc Mu L V1.xx			√
TeSys U Sc Mu R V1.xx	√	√	*
TeSys U Sc St V1.xx	√	√	

* The parameters can be set locally by embedded HMI of multifunction control unit on "TeSys U Sc Mu R V2.xx" and "TeSys U C Mu R V1.xx" variants if the PLC application does not modify any parameters through the network.

Parameters for TeSys U C Ad

The following table gives the description of parameters for TeSys U C Ad:

DeviceNet path	Parameter	Description
6A : 1 : 03	602	Control configuration (thermal fault reset mode)
6B : 1 : 21	682	Communication loss fallback strategy
6B : 1 : 22	683	Controller Local/Remote control mode
6B : 1 : 23	684	Inversion of output configuration
6B : 1 : 24	685	Output LO1 configuration
6B : 1 : 25	686	Outputs OA1 and OA3 configuration
6B : 1 : 26	687	Outputs 13 and 23 configuration
6B : 1 : 27	688	Recovery mode
6B : 1 : 29	690	Disable auto-identification

Parameters for TeSys U C Mu L/R

The following table gives the description of parameters for TeSys U C Mu L/R:

DeviceNet path	Parameter	Description
6A : 1 : 01	600	Define an access code to lock LUCMT keypad
6A : 1 : 02	601	Configuration
6A : 1 : 03	602	Control configuration
6A : 1 : 04	603	Control unit communication on LUCMT port address
6A : 1 : 05	604	Control unit communication on LUCMT port baud rate
6A : 1 : 07	606	Load class
6A : 1 : 08	607	Thermal reset time
6A : 1 : 09	608	Thermal reset threshold
6A : 1 : 0A	609	Thermal warning threshold
6A : 1 : 0B	610	Ground fault trip timeout
6A : 1 : 0C	611	Ground fault trip threshold
6A : 1 : 0D	612	Ground fault warning threshold
6A : 1 : 0E	613	Phase imbalance trip timeout at start-up
6A : 1 : 0F	614	Phase imbalance trip timeout while running
6A : 1 : 10	615	Phase imbalance trip threshold
6A : 1 : 11	616	Phase imbalance warning threshold
6A : 1 : 12	617	Jam trip timeout
6A : 1 : 13	618	Jam trip threshold
6A : 1 : 14	619	Jam warning threshold
6A : 1 : 15	620	Undercurrent trip timeout
6A : 1 : 16	621	Undercurrent trip threshold
6A : 1 : 17	622	Undercurrent warning threshold
6A : 1 : 18	623	Long start trip timeout
6A : 1 : 19	624	Long start trip threshold
6A : 1 : 1A	625	Long start warning threshold
6A : 1 : 1B-1C	626-627	Reserved
6A : 1 : 1D	628	Current transformer primary
6A : 1 : 1E	629	Current transformer secondary
6A : 1 : 1F	630	Current transformer external passes
6A : 1 : 20-23	631-634	Reserved
6B : 1 : 01	650	Display language
6B : 1 : 02	651	Display of running items
6B : 1 : 03	652	Full Load Amps setting (%FLA max)
6B : 1 : 04-1E	653-679	Reserved

DeviceNet path	Parameter	Description
6B : 1 :1F	680	Communication module identification code setting
6B : 1 : 21	682	Communication loss fallback strategy
6B : 1 : 22	683	Controller Local/Remote control mode
6B : 1 : 23	684	Inversion of output configuration
6B : 1 : 24	685	Output LO1 configuration
6B : 1 : 25	686	Outputs OA1 and OA3 configuration
6B : 1 : 27	687	Outputs 13 and 23 configuration
6B : 1 : 28	688	Recovery mode
6B : 1 : 29	690	Disable auto-identification

Parameters for TeSys U Sc Ad

The following table gives the description of parameters for TeSys U Sc Ad:

DeviceNet path	Parameter	Description
6A : 1 : 03	602	Control configuration (thermal fault reset mode)
6B : 1 : 21	682	Communication loss fallback strategy
6B : 1 : 23	684	Inversion of output configuration
6B : 1 : 24	685	Output LO1 configuration
6B : 1 : 25	686	Outputs OA1 and OA3 configuration
6B : 1 : 27	688	Recovery mode
6B : 1 : 29	690	Disable auto-identification

Parameters for TeSys U Sc Mu L/R

The following table gives the description of parameters for TeSys U Sc Mu L/R:

DeviceNet path	Parameter	Description
6A : 1 : 01	600	Define an access code to lock LUCMT keypad
6A : 1 : 02	601	Configuration
6A : 1 : 03	602	Control configuration
6A : 1 : 04	603	Control unit communication on LUCMT port address
6A : 1 : 05	604	Control unit communication on LUCMT port baud rate
6A : 1 : 06	605	Overcurrent trip threshold
6A : 1 : 07	606	Load class
6A : 1 : 08	607	Thermal reset time
6A : 1 : 09	608	Thermal reset threshold
6A : 1 : 0A	609	Thermal warning threshold
6A : 1 : 0B	610	Ground fault trip timeout
6A : 1 : 0C	611	Ground fault trip threshold
6A : 1 : 0D	612	Ground fault warning threshold
6A : 1 : 0E	613	Phase imbalance trip timeout at start-up
6A : 1 : 0F	614	Phase imbalance trip timeout while running
6A : 1 : 10	615	Phase imbalance trip threshold
6A : 1 : 11	616	Phase imbalance warning threshold
6A : 1 : 12	617	Jam trip timeout
6A : 1 : 13	618	Jam trip threshold
6A : 1 : 14	619	Jam warning threshold
6A : 1 : 15	620	Undercurrent trip timeout
6A : 1 : 16	621	Undercurrent trip threshold
6A : 1 : 17	622	Undercurrent warning threshold
6A : 1 : 18	623	Long start trip timeout
6A : 1 : 19	624	Long start trip threshold

DeviceNet path	Parameter	Description
6A : 1 : 1A	625	Long start warning threshold
6A : 1 : 1B-23	626-634	Reserved
6B : 1 : 01	650	Display language
6B : 1 : 02	651	Display of running items
6B : 1 : 03	652	Full Load Amps setting (%FLA max)
6B : 1 : 04-1E	653-679	Reserved
6B : 1 :1F	680	Communication module identification code setting
6B : 1 : 21	682	Communication loss fallback strategy
6B : 1 : 22	683	Local/Remote control
6B : 1 : 23	684	Inversion of output configuration
6B : 1 : 24	685	Output LO1 configuration
6B : 1 : 25	686	Outputs OA1 and OA3 configuration
6B : 1 : 26	687	Reserved
6B : 1 : 27	688	Recovery mode
6B : 1 : 29	690	Disable auto-identification

Parameters for TeSys U Sc St

The following table gives the description of parameters for TeSys U Sc St:

DeviceNet path	Parameter	Description
6B : 1 : 21	682	Communication loss fallback strategy
6B : 1 : 23	684	Inversion of output configuration
6B : 1 : 24	685	Output LO1 configuration
6B : 1 : 25	686	Outputs OA1 and OA3 configuration
6B : 1 : 27	688	Recovery mode
6B : 1 : 29	690	Disable auto-identification

Factory Configuration and Setting

Parameter Types

Setting communication module parameters allows you to determine:
- the operation mode,
- the reset mode on thermal overload fault,
- the correspondence between the communication module outputs and the LUTM controller inputs.

Default Configuration and Setting Registers

Configuration registers (6A : 1 : xx) and Setting registers (6B : 1 : xx) are in a read/write access. Factory default values are:

DeviceNet path	Register	Subject	Factory value	Meaning
6A : 1 : 03	602.0	Reset mode after thermal overload fault	1	"**Manual**" mode
	602.4	Validate the communication between LUCM and LULC09	1	Forced to 0 (zero), this bit forbids any communication between LUCM multifunction control unit and LULC09 communication module.
6B : 1 : 21	682	Fallback mode of control outputs on communication loss	2	**Forced stop** Power base: OA1 and OA3 to 0 Controller base: 13 and 23 to 0
6B : 1 : 22	683	Local or bus operation mode with LUTM and LULC09	0	LUTM output control mode "**remote via the bus**"
6B : 1 : 23	684	LULC09 outputs inversion	0	Outputs status reflects control bits
6B : 1 : 24 LSB 6B : 1 : 25 LSB 6B : 1 : 25 MSB 6B : 1 : 26 LSB 6B : 1 : 26 MSB	685 LSB 686 LSB 686 MSB 687 LSB 687 MSB	Assignment of: - output LO1 - output OA1 - output OA3 (on a controller base) - output 13 - output 23	 2 12 13 12 13	 LO1 reflects control bit 700.0 OA1 reflects control bit 704.0 OA3 reflects control bit 704.1 13 reflects control bit 704.0 23 reflects control bit 704.1
6B : 1 : 27	688	Recovery mode after power-off	0	The outputs recover the status they had before power-off
6B : 1 : 29	690	Disable auto-identification	0	Automatic identification of control unit

NOTE: For more details, refer to *TeSys U Communication Variables User's Manual*.

Customizing your Configuration

Parameter Types

You can either use the factory settings or customize your configuration.

Parameters concerning the communication module are described below.

For other parameters concerning the Control Unit, refer to the "TeSys U communication variables User's manual."

Control Configuration

Bits 0, 1 and 2 of this register are used to configure the reset mode after thermal overload fault. Only one of these bits must be set to 1 to select the reset mode. Other bits (3 to 8) are dedicated to the configuration of the Modbus port of the multifunction control unit.

DeviceNet path	Bit	Description/Possible values	Advanced Control Unit	Multifunction Control Unit
6A : 1 : 03	Reset mode after thermal overload fault bits 0-2 (1 bit is set to 1)			
	602.0	Manual (default value =1)	√	√
	602.1	Remote (or multifunction control unit keypad)	√	√
	602.2	Automatic	√	√
	602.3	Control Unit communication parity 0 = none (default) - 1 = even		√
	602.4	Communication control enabled/disabled 0 = disabled - 1 = enabled (default)		√
	Multifunction control unit port watchdog bits 5-8 (1 bit is set to 1)			√
	602.5	Ignored (default value =1)		√
	602.6	Warning		√
	602.7	Drop-out		√
	602.8	Trip		√
	602.9 to 602.15	Reserved		√

Communication Loss Fallback strategy

Communication loss fallback strategy parameter (register 682 or DeviceNet path 6B:1:21) is used to adjust the fallback mode in case of a communication loss with the PLC.

Register 682 Value	Fallback Mode
0	Ignored
1	Freeze outputs
2	Stop
3	Signal comm loss warning
4	Force run forward
5	Force run reverse

⚠ WARNING

AUTOMATIC RESTART OF THE MOTOR

If communication is stopped, the outputs OA1-OA3 take the status corresponding to the selected fallback mode (register 682), but the control bits 704.0 and 704.1 are not modified.

When a loss of communication warning is acknowledged (register 703 or pushbutton on the controller), the motor will automatically restart if the control bits 704.0 or 704.1 were not previously overwritten to zero by the PLC application.

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

Description of the different fallback modes:

Fallback Mode	Loss of Communication	Communication Recover	Loss of Communication Acknowledgement
Ignored (reg 682 = 0)	No detection of the loss of communication	No detection of the loss of communication	No acknowledgement of the loss of communication
	OA1 and OA3 keep their status	OA1 and OA3 keep their status	
Freeze outputs (reg 682 = 1)	OA1 and OA3 keep their status	OA1 and OA3 keep their status	On rising edge by bit 703.3 (do not leave set to 1)
	ERR LED blinking on the front face	ERR LED blinking on the front face	Once the acknowledgement is done, the last command stored in register 704 is enabled
		Every new On/Off command is stored but with no impact on OA1 and OA3	ERR LED switches off
Stop (reg 682 = 2)	OA1 and OA3 are forced to 0	OA1 and OA3 are forced to 0	On rising edge by bit 703.3 (do not leave set to 1)
	ERR LED blinking on the front face	ERR LED blinking on the front face	Once the acknowledgement is done, the last command stored in register 704 is enabled
		Every new On/Off command is stored but with no impact on OA1 and OA3	ERR LED switches off
Signal comm loss warning (reg 682 = 3)	OA1 and OA3 keep their status	OA1 and OA3 keep their status	On rising edge by bit 703.3 (do not leave set to 1)
	ERR LED blinking on the front face	ERR LED blinking on the front face	ERR LED switches off
		Every new On/Off command is considered and has an impact on OA1 and OA3	
Force run forward (reg 682 = 4)	OA1 is forced to 1 OA3 is forced to 0	OA1 is forced to 1 OA3 is forced to 0	On rising edge by bit 703.3 (do not leave set to 1)
	ERR LED blinking on the front face	ERR LED blinking on the front face	Once the acknowledgement is done, the last command stored in register 704 is enabled
		Every new On/Off command is stored but with no impact on OA1 and OA3	ERR LED switches off
Force run reverse (reg 682 = 5)	OA1 is forced to 0 OA3 is forced to 1	OA1 is forced to 0 OA3 is forced to 1	On rising edge by bit 703.3 (do not leave set to 1)
	ERR LED blinking on the front face	ERR LED blinking on the front face	Once the acknowledgement is done, the last command stored in register 704 is enabled
		Every new On/Off command is stored but with no impact on OA1 and OA3	ERR LED switches off

Controller Local/Remote Control Mode

Controlling LUTM outputs 13 and 23 depends on the operating mode selected in register Controller Local/Remote Mode.

DeviceNet path	Register	Control Mode	Value	Comment	
6B : 1 : 22	683	Remote	0	Outputs 13 and 23 are controlled only by the bus (default value). *Status of inputs I.1 and I.2 does not affect outputs 13 and 23.*	
		Local	1	Outputs 13 and 23 are controlled only by inputs **I.1** and **I.2**. *Controlling outputs via the bus is not taken into account.*	
		Mixed Input I.10 takes priority	2	If I.10 = 1: Local mode	Outputs 13 and 23 are controlled by inputs **I.1** and **I.2**.
				If I.10 = 0: Remote Mode	Outputs 13 and 23 are controlled only by the bus. Status of inputs **I.1** and **I.2** does not affect outputs 13 and 23.

Inversion of Outputs Configuration

Depending on your needs (signalling, run, stop, etc.), you can assign a NO or NC status to outputs OA1, OA3 and LO1, by configuring Inversion of outputs configuration register.

DeviceNet path	Register	Bit	Value	Comment
6B : 1 : 23	684	0	0	No inversion of output OA1 (default value)
			1	Inversion of output OA1
		1	0	No inversion of output OA3 (default value)
			1	Inversion of output OA3
		2	0	No inversion of output LO1 (default value)
			1	Inversion of output LO1

Output LO1 Configuration

To change the assignment (factory setting), write another value (0 to 45), as described in Assignment of outputs LO1, OA1, OA3, 13, 23.

Assignment/control (factory setting) of LULC09 output LO1 is:

DeviceNet path	Register	Value	Factory setting	Comment
6B : 1 : 24 - LSB	685 - LSB	0 to 45	2	Output LO1 = image of register 700.0

Output OA1 Configuration

To change the assignment (factory setting), write another value (0 to 45), as described in Assignment of outputs LO1, OA1, OA3, 13, 23.

Assignment/control (factory setting) of LULC09 output OA1 is:

DeviceNet path	Register	Value	Factory setting	Comment
6B : 1 : 25 - LSB	686 - LSB	0 to 45	12	Output OA1 = image of register 704.0

Output OA3 Configuration

To change the assignment (factory setting), write another value (0 to 45), as described in Assignment of outputs LO1, OA1, OA3, 13, 23.

Assignment/control (factory setting) of LULC09 output OA3 is:

DeviceNet path	Register	Value	Factory setting	Comment
6B : 1 : 25 - MSB	686 - MSB	0 to 45	13	Output OA3 = image of register 704.1

Output 13 Configuration

To change the assignment (factory setting), write another value (0 to 45), as described in Assignment of outputs LO1, OA1, OA3, 13, 23.

Assignment/control (factory setting) of LULC09 output 13 is:

DeviceNet path	Register	Value	Factory setting	Comment
6B : 1: 26 - LSB	687 - LSB	0 to 45	12	Output 13 = image of register 704.0

Output 23 Configuration

To change the assignment (factory setting), write another value (0 to 45), as described in Assignment of outputs LO1, OA1, OA3, 13, 23.

Assignment/control (factory setting) of LULC09 output 23 is:

DeviceNet path	Register	Value	Factory setting	Comment
6B : 1 : 26 - MSB	687 - MSB	0 to 45	13	Output 23 = image of register 704.1

Recovery Mode After Stopping (Reg 688)

If you use register 704 to control outputs OA1-OA3, writing value 1 to register 688 locks the motor and prevents it from restarting after the occurrence of certain events:
● Loss followed by restoration of 24 VDC (outputs OA1-OA3).
● Change in position of rotary knob on power base followed by return to Ready position.

When one of these events occurs, control bits 704.0 and 704.1 (outputs OA1-OA3) are forced to 0 automatically. Once these conditions have disappeared, control of the motor can be restored by sending a new run command.

⚠ WARNING

AUTOMATIC RESTART OF THE MOTOR

In case of a cyclic writing to register 704 (e.g., an LUFP• gateway in its predefined configuration), this monitoring function must be used with caution. The application program must take this state into account and request that bits 704.0 or 704.1 are written to 0. Otherwise, when this event disappears, the motor will restart automatically.

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

Disable Auto- Identification

Disable auto-identification can be automatic or forced.

DeviceNet path	Register	Value	Comment
6B : 1 : 29	690	0	Automatic identification
		1	Forced to LUCB/C/D
		2	Forced to LUCM

Assignment of Outputs LO1, OA1, OA3, 13, 23

This table describes the assigned values to configure outputs LO1, OA1, OA3, 13 and 23.

Value	Description of assigned value	LUCBT / LUCDT	LUCMT
0	The corresponding output is forced to 0 (0V)	√	√
1	The corresponding output is forced to 1 (24V)	√	√
2	State of register 700, bits 0-4: - 700.0 --> LO1 - 700.1 --> OA1 - 700.2 --> OA3 - 700.3 --> 13 - 700.4 --> 23	√	√
3	452.3 (Thermal overload fault)	√	√
4	461.3 (Thermal overload warning)	√	√
5	457.0 (System ready)	√	√
6	457.1	√	√
7	State of bit 457.2	√	√
8	The corresponding output copies the result of "Reflex stop 1: forward"	√	√
9	The corresponding output copies the result of "Reflex stop 1: reverse"	√	√
10	The corresponding output copies the result of "Reflex stop 2: forward"	√	√
11	The corresponding output copies the result of "Reflex stop 2: reverse"	√	√
12	The corresponding output copies the result of "Forward direction" (default OA1 value)	√	√
13	The corresponding output copies the result of "Reverse direction" (default OA3 value)	√	√
14	452.0 (Short-circuit fault)	√	√
15	452.1 (Overcurrent fault)	√	√
16	452.2 (Ground fault)		√
17	452.3 (Thermal overload fault)	√	√
18	452.4 (Long start fault)		√
19	452.5 (Mechanical locking (jam) fault)		√
20	452.6 (Phase imbalance fault)		√
21	452.7 (Underload fault)		√
22	452.8 (Shunt trip)		√
23	452.9 (Test trip)		√
24	452.10 (Communication loss fault on LUCMT Modbus port)		√

Value	Description of assigned value	LUCBT / LUCDT	LUCMT
25	452.11 (Control unit internal fault)	√	√
26	452.12 (Module identification or internal communication fault)		√
27	452.13 (Module internal fault)	√	√
28-31	*(Reserved)*		
32	461.2 (Ground fault warning)		√
33	461.3 (Thermal overload warning)	√	√
34	461.4 (Long start warning)		√
35	461.5 (Mechanical locking (jam) warning)		√
36	461.6 (Phase imbalance warning)		√
37	461.7 (Under-current warning)		√
38-39	*(Reserved)*		
40	461.10 (Communication loss on LUCMT Modbus port)		√
41	461.11 (Internal temperature warning)		√
42	461.12 (Module identification or internal communication warning)		√
43-44	*(Reserved)*		
45	461.15 (Module warning)	√	√

PKW Objects

Overview

The TeSys U motor starter supports PKW (Periodically Kept in acyclic Words). The PKW feature consists of two 8-byte objects: the Periodic Registers Service Objects (C5h). These objects enable a DeviceNet™ master to read or write any manufacturer-specific register (class 0x64 to 0x81) using cyclic I/O messaging.

The 8 bytes of the objects are interpreted as a request telegram or response telegram encapsulated in I/O messaging data.

Examples of I/O Messages without PKW

The table below shows the byte arrangement of I/O messages for the Te Sys U Sc Mu R/L, when Output Assembly Instance 100 = TeSys U Control Register and Input Assembly Instance 110 = TeSys U Monitoring Registers:

IN		OUT
0		0
1		1
2		2
3		3
4		4
5		5
6		
7		

The table below shows the corresponding double word arrangement:

IN						OUT			
DW0	3	2	1	0		3	2	1	0
DW1	7	6	5	4		-	-	5	4
	Reg 458		Reg 455			Reg 703		Reg 704	
	Reg 457		Reg 461			-		Reg 700	

Examples of I/O Messages with PKW

The table below shows the byte arrangement of I/O messages for theTeSys U Sc Mu R/L when Output Assembly Instance 103 = PKW Request and TeSys U Control Register and Input Assembly Instance 113 = PKW Response and TeSys U Monitoring Registers:

IN		OUT
0 PKW IN 0		0 PKW OUT 0
1 PKW IN 1		1 PKW OUT 1
2 PKW IN 2		2 PKW OUT 2
3 PKW IN 3		3 PKW OUT 3
4 PKW IN 4		4 PKW OUT 4
5 PKW IN 5		5 PKW OUT 5
6 PKW IN 6		6 PKW OUT 6
7 PKW IN 7		7 PKW OUT 7
8		8
9		9
10		10
11		11
12		12
13		13
14		
15		

The table below shows the corresponding double word arrangement:

IN				OUT			
3 PKW IN 3	2 PKW IN 2	1 PKW IN 1	0 PKW IN 0	3 PKW OUT 3	2 PKW OUT 2	1 PKW OUT 1	0 PKW OUT 0
7 PKW IN 7	6 PKW IN 6	5 PKW IN 5	4 PKW IN 4	7 PKW OUT 7	6 PKW OUT 6	5 PKW OUT 5	4 PKW OUT 4
11	10	9	8	11	10	9	8
15	14	13	12	-	-	13	12

Function code	Object address	Object address	Function code	Object address	Object address
Value 2		Value 1	Value 2		Value 1
Reg 458		Reg 455	Reg 703		Reg 704
Reg 457		Reg 461	-		Reg 700

PKW OUT Data

PKW OUT Data request (DeviceNet Master -> TeSys U device) are mapped in modules supporting PKW.

To access a register, you must select 1 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.

Register numbers are given in TeSys U Communication Variables User's Manual.

Word 1	Word 2			Word 3	Word 4
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

Depending on the PLC platform used, refer to the PKW OUT description in Little and Big endian formats to know the positioning of each field inside each word.

Any changes in the function field will trigger the handling of the request (except if Function code = 0x00).

Toggle bit must change at each consecutive request. This mechanism allows the request initiator to detect that a response is ready by polling the toggle bit in response. When this bit in the OUT data becomes equal to the response emitted toggle bit in the IN data, then the response is ready.

PKW IN Data

PKW IN Data Response (TeSys U device -**>** DeviceNet Master) are mapped in modules supporting PKW. The TeSys U device echoes the same register address and function code or eventually an error code:

Word 1	Word 2			Word 3	Word 4
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 request	Same as 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		_	_

Depending on the PLC platform used, refer to the PKW IN description in Little and Big endian formats to know the positioning of each field inside each word.

If the initiator tries to write a TeSys U object or register to an unauthorized value, or tries to access an inaccessible register, an error code is answered (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 or register remains at the old value.

If you want to re-trigger exactly the same command, you must:
● reset the Function code to 0x00,
● wait for the response frame with the function code equal to 0x00, then
● set it again 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

Error Code	Error Name	Explanation

Using of main registers for a simplified management

Before commissioning a motor starter, it is interesting to know which registers you access, and in which order.

Illustration of registers used

The following illustration gives you basic information about commissioning, through registers: configuration, control and monitoring (state of the system, measurements, faults and warnings, acknowledgment). Starting from the predefined factory configuration, you will be able to visualize, and even anticipate the behavior of your system.
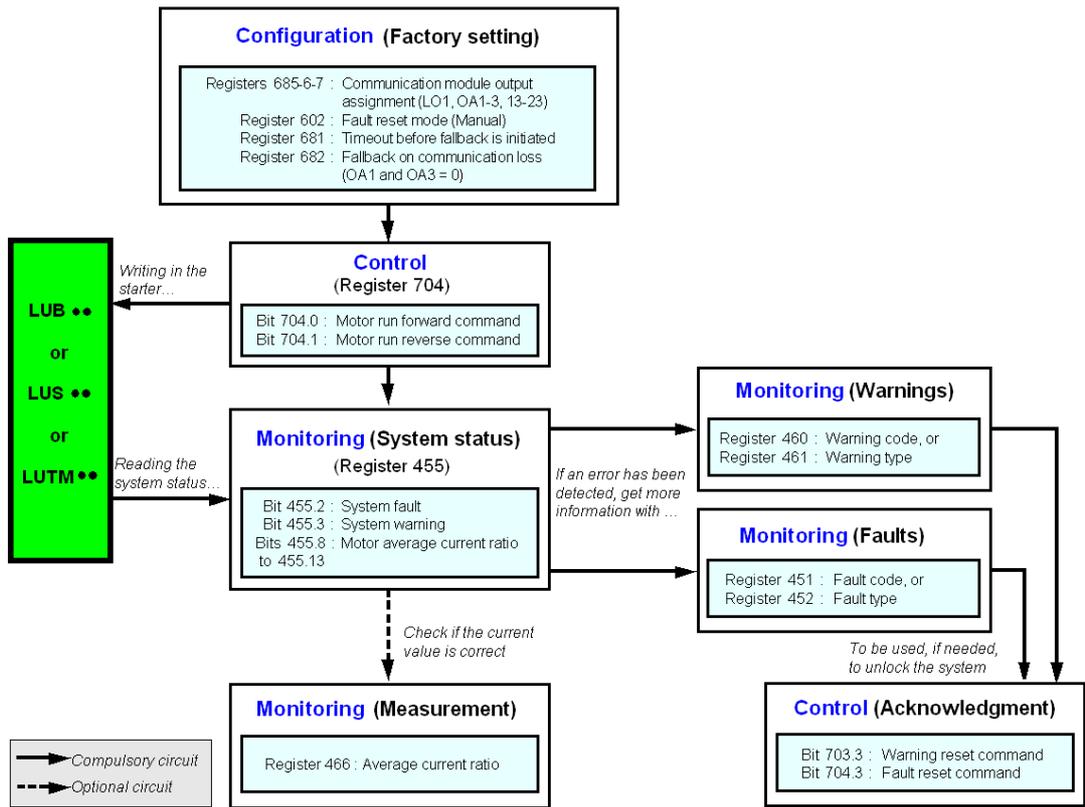
Chapter 4
Managing faults and warnings

This chapter explains how to manage the different types of faults and warnings that may occur.

What Is in This Chapter?

This chapter contains the following topics:

Viewing a fault

A fault is signalled by various different indicators:

● status of LULC09 communication module's LEDs,

with a power base:

● status of rotary button on the power base (0 or "trip"),
● status of outputs,

with a controller base:

● status of the LEDs on the controller base,
● status of output relays,

with a standard or advanced control unit:

● internal signals sent to LULC09 communication module,
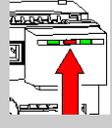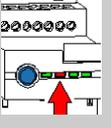
with a multifunction control unit:

● warning,
● message(s) displayed on screen,
● internal communication with the LULC09 communication module,
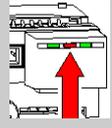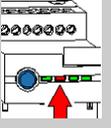● presence of an exception code (PLC report).

NOTE: Warnings and faults are stored in specific registers. Refer to *Communication Variables User's Manual*: fault monitoring registers (450 to 452) and warning monitoring registers (460 to 461).

Application faults

Application fault acknowledgment

Possible application faults are listed below. They can be reset (acknowledged): manually / automatically / remotely.

Application faults	Registers		LULC09	LUCM•	LUTM	Fault reset
	451 Fault number	**452 Fault bit**	"ERR"	(line 2)	"FAULT"	
Short-circuit fault	1	_.0 = 1	off	SC	-	Manual reset
Over-current fault	2	_.1 = 1		I>>	-	
Thermal overload fault	4	_.3 = 1		overload	-	Depending on reset mode set in register 602
LUCM• multifunction control unit application fault	3 and 5 set to 12		*See LUCM••BL - LUCMT1BL Multifunction Control Unit User's Manual*			

Application faults	Registers		LULC09	LUCM•	LUTM	Fault reset
	451 Fault number	**452 Fault bit**	"ERR"	(line 2)	"FAULT"	
LUTM external fault signalled by I.6 set to 0	201	_.0 = 1	-	-	See LUTM Controller User's Guide	Automatic with I.6 set back to 1

Overload fault with LU•B•/LU•S• power base

After a thermal overload fault, rotary button or blue push-button on the front can be used, whatever the reset mode that was set.

Configuration register	Acknowledgment (reset)	Means
602.0 = 1	**"manual"** local	With rotary button on LU•B• With blue push-button on LU•S•
	"manual" remote	With kit LU9 AP•• on LU•B• With kit LU9 •• on LU•S•
602.1 = 1	**"remote"**	Acknowledged by setting bit 704.3 to 1
602.2 = 1	**"automatic"**	Managed by control unit

Overload fault with LUTM controller base

After a thermal overload fault, blue push-button on the front or input I.5 can be used, whatever the reset mode that was set.

Configuration register	Acknowledgment (reset)	Means
602.0 = 1	**"manual"** local	With blue push-button on the front
	"manual" remote	With reset mode on front of the rack or table (via input **I.5**)
602.1 = 1	**"remote"**	Acknowledged by bit 704.3. This bit is active on rising edge and must be reset to 0 by programming.
602.2 = 1	**"automatic"**	Managed by control unit

NOTE: Reset mode must be set.
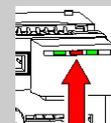
Internal Faults

Internal Fault Acknowledgement

Here is the list of possible internal faults:

Internal faults	Registers		LULC09	LUCM•	LUTM	Fault acknowledgement
	451 Fault number	452 Fault bit	"ERR"	(line 2)	"FAULT"	
LULC09 DeviceNet communication module fault	14	-	Off	M14	-	LULC09 and LUCM• power off then on
LULC09 DeviceNet communication module not installed or not powered, or communication loss with the module	15	-		M15	-	
LUC•• control unit internal fault	54	_.11 = 1		M54	-	
LUCM• multifunction control unit internal fault	51 to 53, 55 to 63	*See LUCM - LUCMT Multifunction Control Unit User's Manual*				
Write-to-EEPROM fault	100	_.13 = 1	On	M100	-	LULC09 power off then on
Communication fault with LUCM• multifonction control unit	101	_.12 = 1	On	M101	Blinking	LULC09 power off then on
Checksum-on-EEPROM fault	102	_.13 = 1	On	M102		Rising edge on 704.3
EEPROM configuration fault	104	_.13 = 1	On	M104		Rising edge on 704.3
Communication fault with LUTM controller base	105	_.13 = 1	On	M105		LULC09 power off then on
Communication fault with LULC09 module	205	_.13 = 1	On	-	See LUTM User's Manual	LUTM power off then on
No control unit	206					LUTM power off then on

Warnings - Communication Loss

Warning Acknowledgement

List of possible warnings.

Warnings	Registers		LULC09	LUCM•	LUTM	warning acknowledgement
	460 warning number	461 warning bit	"ERR"	(line 1)	"FAULT"	
Warning on thermal overload	4	_.3 = 1	-	Wrng overload	-	Automatic when overload is lower than 85%
Warning on communication loss with the master	109	_.15 = 1	Flashing	Comm loss	-	Acknowledgement by bit 703.3. This bit is active on rising edge and must be reset to 0 by programming.
Warning on LUCM• multifunction control unit	2 and 4 to 13	See LUCM - LUCMT Multifunction Control Unit User's Manual				
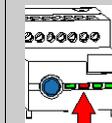
Recovery after Communication Loss

Following an acknowledgement by setting bit 703.3 to 1, recovery occurs depending on the status of control bits 704.0 and 704.1.

Chapter 5
Configuration of Predefined Functions

Overtravel Limit Switch (Communication Module Reflex Functions)

Introduction

The Overtravel Limit Switch allows you to perform precise and iterative positionings, without any interaction of bus or PLC cycle time. It is a module function initiated at the LULC09 communication module level.

There are two types of functions:
● Reflex1: Communication module "reflex stop 1" function
● Reflex2: Communication module "reflex stop 2" function.

Description of Reflex1

Sensor 1 (logic input LI1) directly controls the motor stop.

After a new running order (stop then run order), the motor restarts even with sensing (LI1=1).

1 Bus   **2** Sensor 1 (LI1)
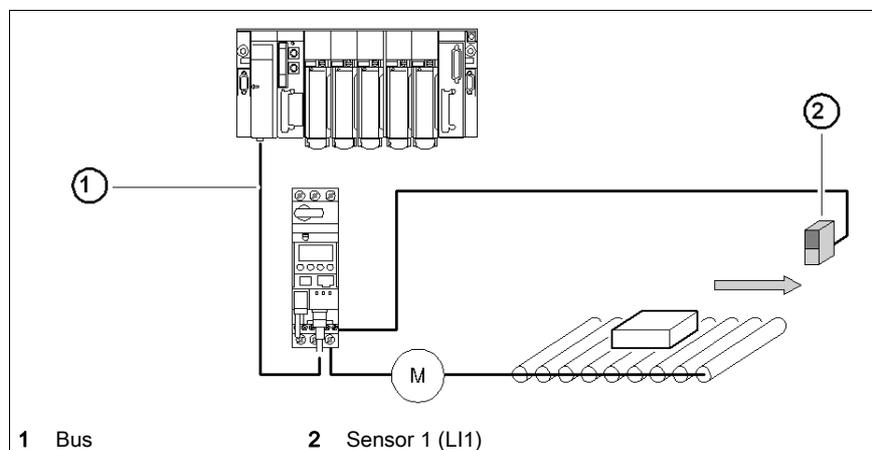
NOTE: In case of a reversing starter, reflex stop impacts both directions.

Information chaining.
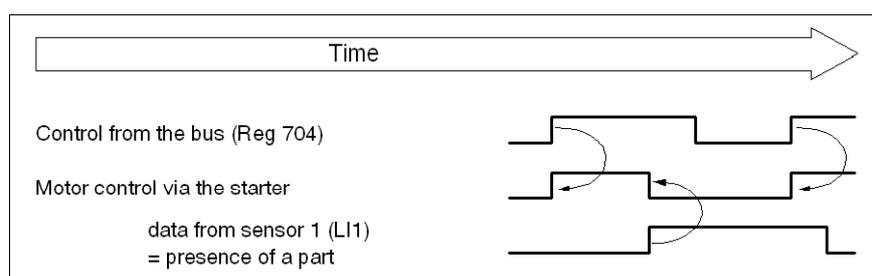
Description of Reflex2

Sensor 1 (logic input LI1) controls the motor stop when running forward.

Sensor 2 (logic input LI2) controls the motor stop when running reverse.

After a new running order (stop then run order), the motor restarts even with sensing (LI1=1 or LI2 = 1).

| **1** | Bus | **2** | Sensor 1 (LI1) | **3** | Sensor 2 (LI2) |

NOTE: Sensor 2 (LI2) does not impact the forward direction, and sensor 1 (LI1) does not impact the reverse direction.

Reflex Stop Selection

In order to use a "reflex stop" function, you have to select it from the register having an output to monitor.

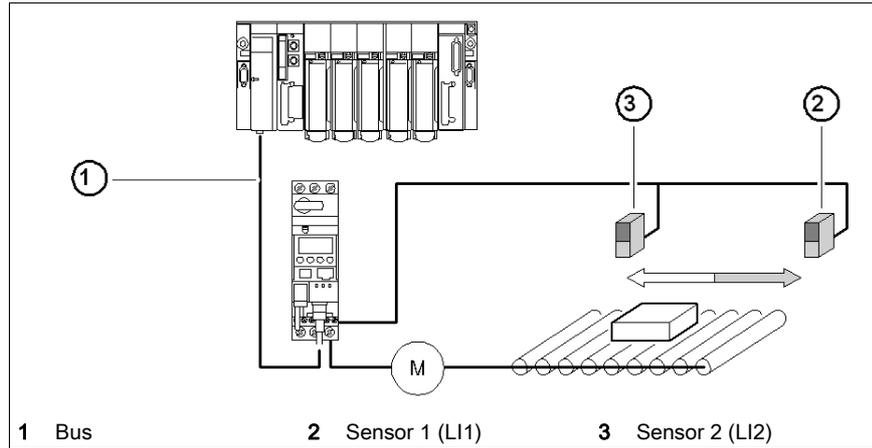| Reflex function	Motor spinning direction	Value of Reg•	LUB••/S•• - LU2B••/2S•• base		LUTM•• base
			Output LO1	Outputs OA1 OA3	Outputs 13 23
Reflex1	Reflex1.Fw = forward run	8	Reg. 685 (2007:24) (LSB)	Reg. 686 (2007:25) (LSB) (MSB)	Reg. 687 (2007:26) (LSB) (MSB)
	Reflex1.Rev = reverse run	9			
Reflex2	Reflex2.Fw = forward run	10			
	Reflex2.Rev = reverse run	11			

NOTE: Before using a "reflex stop" function, you need to assign outputs OA1/OA3 to forward/reverse run. Do it in register **686** (2007:25). By default, OA1 is assigned to forward run and OA3 to reverse run.

Reflex1.Fw

This function is enabled on rising edge, not on level.

.Fw	LI1 = 1 makes the motor stop, whatever the running direction.
	After a new running order (a stop order followed by a run order),
	even though logic input LI1 = 1,
	the motor starts again in the chosen direction.

NOTE: Logic input LI2 is not used.

Reflex1.Rev

This function is enabled on rising edge, not on level.

.Rev	LI1 = 1 makes the motor stop, whatever the running direction.
	After a new running order (a stop order followed by a run order),
	even though logic input LI1 = 1,
	the motor starts again in the chosen direction.

NOTE: Logic input LI2 is not used.

Reflex2.Fw

This function is enabled on rising edge, not on level.

.Fw	Logic input LI1 = 1 makes the motor stop in forward run. Logic input LI2 = 1 makes the motor stop in reverse run. After a new running order (a stop order followed by a run order), even though logic input LI2 = 1, the motor starts again.

NOTE: Logic input LI2 does not impact the forward run and logic input LI1 does not impact the reverse run.

Reflex2.Rev

This function is enabled on rising edge, not on level.

.Rev	Logic input LI2 = 1 makes the motor stop in reverse run. Logic input LI1 = 1 makes the motor stop in forward run. After a new running order (a stop order followed by a run order), even though logic input LI2 = 1, the motor starts again.

NOTE: Logic input LI2 does not impact the forward run and logic input LI1 does not impact the reverse run.

Appendices

Appendix A
Object Dictionary

Overview

The DeviceNet protocol uses 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.

What Is in This Chapter?

This chapter contains the following topics:

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: • Sc=St → 0x11 • Sc=Ad → 0x12 • Sc=Mu → 0x13 • C+Ad → 0x22 • P2+Mu → 0x23 Local mode: • Sc+Mu → 0x0113 • C+B2 → 0x0123
4	Get	Revision	Struct. of: UInt UInt	Product configuration	Product version
5	Get	Status	Word	01	See the table below.
6	Get	Serial number	UDInt	Unique number	Read from the EEPROM during start-up
7	Get	Product name	Struct. of: USInt String	"LULC09"	Read from the controller during start-up in registers [64] to [69]: *Control Unit Identification*

Attribute 5 status:

Bit	Definition	Values
0	Owned by Master (predefined Master/Slave connection)	Provided by the stack
1	*Reserved*	0
2	Configured	NOT ([601.0])
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

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: TeSys U Control Registers

This assembly contains several control registers commonly used with a TeSys U 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
See *PKW Objects*, *page 58* for details.							

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 TeSys U 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: TeSys U Monitoring Registers (with dynamic configuration)

This assembly contains several monitoring registers commonly used with a TeSys U device.

Configuration	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
LUB/LU2B, LUS/LU2S + LUCA	Path: 68:01:06 *(Register [455])*		Path: 68:01:09 *(Register [458])*		Empty[1]		Empty[1]	
LUB/LU2B, LUS/LU2S + LUCB/C/D	Path: 68:01:06 *(Register [455])*		Path: 68:01:09 *(Register [458])*		Path: 68:01:0C *(Register [461])*		Empty[1]	
LUB/LU2B, LUS/LU2S + LUCM	Path: 68:01:06 *(Register [455])*		Path: 68:01:09 *(Register [458])*		Path: 68:01:0C *(Register [461])*		Path: 68:01:08 *(Register [457])*	

Configuration	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
LUTM + LUCBT/DT	Path: 68:01:06 (Register [455])		Path: 68:01:09 (Register [458])		Path: 68:01:0C (Register [461])		Path: 68:01:0A (Register [459])	
LUTM + LUCMT	Path: 68:01:06 (Register [455])		Path: 68:01:09 (Register [458])		Path: 68:01:0C (Register [461])		Path: 68:01:0A (Register [459])	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB

[1] Any byte left vacant at the end of this assembly will not be sent to the bus; so if no parameter is attributed to bytes 6 and 7, the assembly data length will be 6 bytes instead of 8 bytes.

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
See *PKW Objects*, *page 58* for details.							

Instance 112: PKW Response 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 Response and TeSys U Monitoring Registers

This assembly is vendor specific.

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

DeviceNet Object

Overview

The DeviceNet Object is used to provide the configuration and status of a physical attachment to the DeviceNet network. A produc 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).

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 sec. (TimeOut)
12	Get/Set	Watchdog_timeout_action	UInt	1 or 3	1: Auto-Delete (Default) 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:
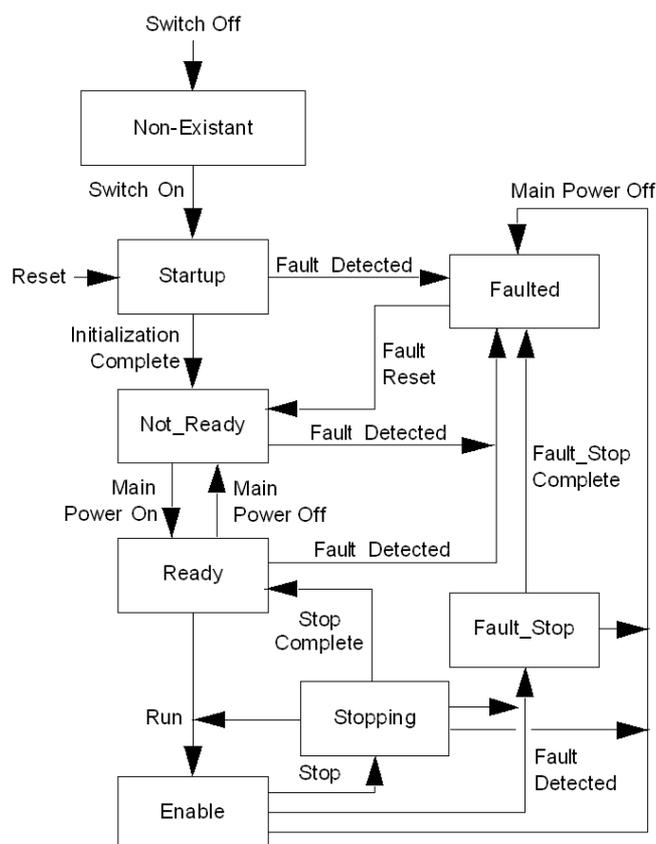
The following table describes the run/stop event matrix:

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
Initialisation 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	

Event	State (N/A = No action)							
	Non-exist	Startup	Not_Ready	Ready	Enabled	Stopping	Fault-Stop	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
2	Get	Max instance	UInt	1	-

Instance Attributes

Attribute ID	Access	Name	Data Type	Value	Only with...	Description
1	Get	NumAttr	UInt		LUCM/LUCMT	Number of Attributes Supported
3	Set/Get	TripFLCSet	UInt	[652]	LUCM/LUCMT	% of FLA max
4	Set/Get	TripClass	USInt	[606]	LUCM/LUCMT	Trip Class Setting (0 to 200)
5	Get	AvgCurrent	Int	[466] x FLA	LUCM/LUCMT	0.1 A
6	Get	%PhImbal	USInt	[471]	LUCM/LUCMT	% Phase Imbalance
7	Get	%Thermal	USInt	[465]	LUCM/LUCMT	% Thermal Capacity
8	Get	IL1 Current	Int	[467] x FLA	LUCM/LUCMT	0.1 A
9	Get	IL2 Current	Int	[468] x FLA	LUCM/LUCMT	0.1 A
10	Get	IL3 Current	Int	[469] x FLA	LUCM/LUCMT	0.1 A
11	Get	Ground Current	Int	[470] x FLA	LUCM/LUCMT	0.1 A
101	Get	IL1 Current	Int	Idem Att. 8	LUCM/LUCMT	0.1 A
102	Get	IL2 Current	Int	Idem Att. 9	LUCM/LUCMT	0.1 A
103	Get	IL3 Current	Int	Idem Att. 10	LUCM/LUCMT	0.1 A
104	Get	Ground Current	Int	Idem Att. 11	LUCM/LUCMT	0.1 A
105	Get	IL1 Current Ratio	UInt	[467]	LUCM/LUCMT	% of FLA
106	Get	IL2 Current Ratio	UInt	[468]	LUCM/LUCMT	% of FLA
107	Get	IL3 Current Ratio	UInt	[469]	LUCM/LUCMT	% of FLA
108	Get	IAV Average Current Ratio	UInt	[466]	LUCB/C/D, LUCBT/DT or LUCM/LUCMT	% of FLA
109	Get	Thermal Capacity Level	UInt	[465]	LUCM/LUCMT	%TripLevel
110	Get	Ground Current	Int	[Idem Att. 11	LUCM/LUCMT	0.1 A
111	Get	Current phase imbalance	UInt	[471]	LUCM/LUCMT	% Imbalance
112	Get	Time to trip	UInt	[511]	LUCM/LUCMT	Seconds
113	Get/Set	Time to Reset	UInt	[450]	LUCM/LUCMT	Seconds
127	Get/Set	Single / Three Ph	Bool	0 if [601.14]=1 1 if [601.13]=1	LUCM/LUCMT	0 = Single phase 1 = Three phases
128	Get/Set	FLA Setting	UInt	[652]	LUCM/LUCMT	Seconds
129	Get/Set	Load Class	UInt	[606]	LUCM/LUCMT	Seconds
132	Get/Set	Thermal Warn Level	UInt	[609]	LUCM/LUCMT	%TripLevel
133	Get/Set	PL Inhibit Time	USInt	[613]	LUCM/LUCMT	Seconds
134	Get/Set	PL Trip Delay	USInt	[614]	LUCM/LUCMT	Seconds
136	Get/Set	GF Trip Delay	USInt	[610]	LUCM/LUCMT	0.1...25.0 Seconds
137	Get/Set	GF Trip Level	USInt	[611]	LUCM/LUCMT	1.0...5.0 A
138	Get/Set	GF Warn Level	USInt	[612]	LUCM/LUCMT	1.0...5.0 A
139	Get/Set	Stall Enabled Time	USInt	[623]	LUCM/LUCMT	0...250 Seconds
140	Get/Set	Stall Trip Level	UInt	[624]	LUCM/LUCMT	100...600
142	Get/Set	Jam Trip Delay	USInt	[617]	LUCM/LUCMT	0.1...25.0 Seconds

Attribute ID	Access	Name	Data Type	Value	Only with...	Description
143	Get/Set	Jam Trip Level	UInt	[618]	LUCM/LUCMT	0...600 %FLA
144	Get/Set	Jam Warn Level	UInt	[619]	LUCM/LUCMT	0...600 %FLA
146	Get/Set	UL Trip Delay	USInt	[620]	LUCM/LUCMT	0.1...25.0 Seconds
147	Get/Set	UL Trip Level	USInt	[621]	LUCM/LUCMT	10...100 %FLA
148	Get/Set	UL Warn Level	USInt	[622]	LUCM/LUCMT	10...100 %FLA
149	Get/Set	CI Inhibit Time	USInt	[613]	LUCM/LUCMT	0...250 Seconds
150	Get/Set	CI Trip Delay	USInt	[614]	LUCM/LUCMT	0.1...25.0 Seconds
151	Get/Set	CI Trip Level	USInt	[615]	LUCM/LUCMT	10...100 %FLA
152	Get/Set	CI Warn Level	USInt	[616]	LUCM/LUCMT	10...100 %FLA
178	Get	CT Ratio	USInt	[628] * [630] / [629]	LUTM + LUCM/LUCMT only	

NOTE: In the table above:

- PL = Current Phase Loss
- GF = Ground Fault
- Stall = Long Start
- UL = Underload
- CI = Current Phase Imbalance

Calculation of current in Amps

In the TeSys U mapping, values of currents are given in %FLA. Here is the formula that must be used to convert them in Amps. This formula must be used for both LUB/2B, LUS/2S and LUTM bases.

I = CurrentInMapping FLA Set FLAmax

and

GroundCurrent = Register_470 FLA Set FLAmin

Where:

- CurrentInMapping: Registers 466 to 470, given in %FLA
- FLA Set: Register 652, given in % of FLAmax
- FLAmax: Register 96, stored in Amps
- FLAmin: Register 96, stored in Amps divided by 4 with an LUB/2B, LUS/2S or divided by 3 with an LUTM.

Registers 96 and 652 can be changed at any time, therefore the module updates their values cyclically.

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) (Default) 5: Instance 110: TeSys U Monitoring registers 6: Instance 111: PKW response object 7: Instance 112: PKW response + EMS2 8:Instance 113: PKW response + TeSys U 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) (Default) * 5: Instance 100: TeSys U control registers 6: Instance 101: PKW Request object 7: Instance 102: PKW Request + EMS 8: Instance 103: PKW Request + TeSys U 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) (Default) 5: Instance 110: TeSys U Monitoring registers 6: Instance 111: PKW response object 7: Instance 112: PKW response + EMS2 8: Instance 113: PKW response + TeSys U monitoring
4	Set/Get	EEPROM Save Parameter	BYTE [0..7]	0: "do nothing" 1: "Save All" 2: "Save Com Parameters" 3: "Save Appli Parameters" Automatically return to 0 when operation is done
5	Set/Get	EEPROM Restore Parameter	BYTE [0..7]	0: "do nothing" 1: "Restore All" 2: " Restore Com Parameters" 3: " Restore Appli Parameters" Automatically return to 0 when operation is done

NOTE: * 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.
** 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.

Instance Service

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

Index

P

Periodic Registers Service Objects, *58*
PKW, *58*
    Periodic Registers Service Objects, *58*
PKW error codes, *60*
Power base, *16*
Power supply, *17*, *18*, *18*, *27*, *28*
Power-up, *17*
producer/consumer model, *24*
product dimensions, *27*
Product functions, *12*

R

Reception of the product, *12*
Reflex1, *69*
Reflex2, *69*
Revert
    output status, *55*
RSNetworx, *36*

S

Setting
    output, *55*
Sub-D 9 connector, *21*

W

Warning, *68*

07/2019