LXM32M

Module DeviceNet
Fieldbus manual

V1.02, 12.2009

0198441113808, V1.02, 12.2009

www.schneider-electric.com

Schneider Electric

Important information

This manual is part of the product.

Carefully read this manual and observe all instructions.

Keep this manual for future reference.

Hand this manual and all other pertinent product documentation over to all users of the product.

Carefully read and observe all safety instructions and the chapter "Before you begin - safety information".

Some products are not available in all countries.
For information on the availability of products, please consult the catalog.

Subject to technical modifications without notice.

All details provided are technical data which do not constitute warranted qualities.

Most of the product designations are registered trademarks of their respective owners, even if this is not explicitly indicated.

0198441113808, V1.02, 12.2009

Table of contents

0198441113808, V1.02, 12.2009

0198441113808, V1.02, 12.2009

0198441113808, V1.02, 12.2009

0198441113808, V1.02, 12.2009

About this manual

This manual applies to the module DeviceNet for the product LXM32M, module identification DNT.

The information provided in this manual supplements the product manual.

Source manuals

The latest versions of the manuals can be downloaded from the Internet at:

http://www.schneider-electric.com

Source EPLAN Macros

For easier engineering, macro files and product master data are available for download from the Internet at:

http://www.schneider-electric.com

Corrections and suggestions

We always try to further optimize our manuals. We welcome your suggestions and corrections.

Please get in touch with us by e-mail:
techcomm@schneider-electric.com.

Work steps

If work steps must be performed consecutively, this sequence of steps is represented as follows:

- ■ Special prerequisites for the following work steps
- ▶ Step 1
- ◁ Specific response to this work step
- ▶ Step 2

If a response to a work step is indicated, this allows you to verify that the work step has been performed correctly.

Unless otherwise stated, the individual steps must be performed in the specified sequence.

Making work easier

Information on making work easier is highlighted by this symbol:

Sections highlighted this way provide supplementary information on making work easier.

Parameters

In text sections, parameters are shown with the parameter name, for example `_IO_act`. A list of the parameters can be found in the product manual in the chapter Parameters.

SI units

SI units are the original values. Converted units are shown in brackets behind the original value; they may be rounded.

Example:
Minimum conductor cross section: 1.5 mm^2 (AWG 14)

0198441113808, V1.02, 12.2009

Inverted signals Inverted signals are represented by an overline, for example $\overline{\text{ST0\_A}}$ or $\overline{\text{ST0\_B}}$.

Glossary Explanations of special technical terms and abbreviations.

Index List of keywords with references to the corresponding page numbers.

Further reading

Recommended literature for further reading

Reference documents

- The CIP Networks Library
  Volume 1
  Common Industrial Protocol
- The CIP Networks Library
  Volume 3
  DeviceNet Adaption of CIP
- DeviceNet terms of Usage Agreement

  http://www.odva.org

User Association **O**pen **D**eviceNet **V**endor **A**ssociation (ODVA)

http://www.odva.org

0198441113808, V1.02, 12.2009

1 Introduction

1

1.1 DeviceNet technology

The ODVA (**O**pen **D**eviceNet **V**endor **A**ssociation) is in charge of the specifications for the DeviceNet network and DeviceNet devices. For more information on the ODVA see: http://www.odva.org

Number of nodes Up to 64 nodes can operate in a DeviceNet network (0 - 63).

Cable length The maximum cable length in the individual network segments depends on the baud rate and the conductor cross section.

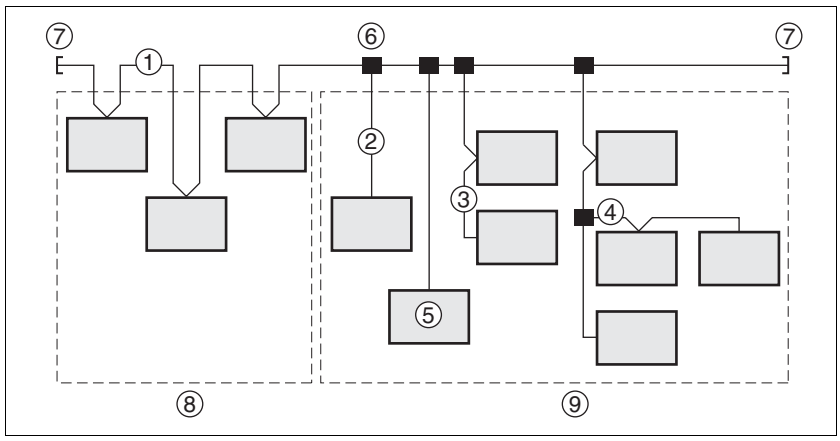

Figure 1.1 A possible DeviceNet structure

(1) Trunk Line
(2) Drop Line, 0 - 6 m
(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

Cable type	125 kbit/s	250 kbit/s	500 kbit/s
Total length of trunk line "Thick" (Thick Trunk)	500 m	250 m	100 m
Total length of trunk line "Thin" (Thin Trunk)	100 m	100 m	100 m
Total length of trunk line "Flat" (Flat Trunk)	420 m	200 m	75 m
Maximum length of a drop line	6 m	6 m	6 m
Maximum length of all drop lines	156 m	78 m	39 m

0198441113808, V1.02, 12.2009

The "thick" trunk line consists of two shielded, twisted lines containing a wire in the center of the cable. The shield is on the outside. The cable is not branched.

The "thin" trunk line has a more flexible design. It is easier to install. The cable is used as a drop line and can also be used as a trunk line for short distances.

Drive profile

The product supports 2 drive profiles:

- "Position Controller Profile" (CIP)
- "Drive Profile Lexium" (vendor-specific)

Properties

The product supports the following features of the DeviceNet specification:

- CIP "Position Controller" (Device Type = 10_h)
- Group 2 servers
- UCMM capable device
- Predefined Master/Slave connections
  - Explicit Message
  - Polled I/O Connection
  - Full "Fragmentation Protocol" support
  - Dynamic establishment of two explicit messages and an I/O message
- Heartbeat
- Shutdown
- Offline Connection Set
- Auto Device Replacement

Data Link Layer

The DeviceNet data link layer uses the transmission mechanisms of the CAN specification. This makes it possible to use a wide selection of available CAN controllers.

Physical Layer

The DeviceNet physical layer uses twisted pair cables. The data is transmitted via one wire pair. The second wire pair is connected to the supply voltage. This means that devices with their own power supply and devices that are powered via DeviceNet (for example, I/O nodes) can be connected.

Another feature of DeviceNet is that devices can be hot-plugged during operation. It is not necessary to shut down the bus.

0198441113808, V1.02, 12.2009

Object model

The following object classes from the CIP object model are available:

Object class	Class ID	Instance ID
Identity Object	1	1
Message Router Object	2	1
DeviceNet Object	3	1
Assembly Object	4	101 Standard Output Assembly 111 Standard Input Assembly 102 Extended Output Assembly 112 Extended Input Assembly
Connection Object	5	1: Explicit Message 2: Poll Connection 5 ... 6: Dynamic Explicit Connection 7: Dynamic I/O Connection
Position Controller Supervisor Object	36	1
Position Controller Object	37	1
Acknowledge Handler Object	43	1
Vendor-specific objects	101...199	1

The vendor-specific class IDs 101 to 199 correspond to the object dictionary (class ID = object group + 100). The attributes of a class correspond to the subindex entry within the object group.

Communication model

DeviceNet uses the producer-consumer communication model. All nodes check the bus as to whether a data packet with the identifiers they support is available. Data packets that are sent by producers can only be received by the consumers of these packets.

DeviceNet allows for the implementation of master-slave, multi-master and peer-to-peer topologies.

Groups of connections

DeviceNet is a connection-oriented network. Connections must be established and managed between two nodes. The connection ID is contained in the 11 bit CAN identifier. There are 4 connection groups with different priorities:

Group 1	Top-priority process data (highest priority)
Group 2	For simple master-slave connections
Group 3	For explicit messages
Group 4	Reserved group (lowest priority)

Device profile

Device profiles are definitions for various types of nodes. The device profile of a node is described in an Electronic Data Sheet (EDS).

Electronic Data Sheet

An EDS file is a file in ASCII format. This file contains device-specific and vendor-specific descriptions of all parameters for a device. The EDS file also contains the fieldbus-specific communication parameters. The EDS file is required for commissioning.

http://www.schneider-electric.com

0198441113808, V1.02, 12.2009

0198441113808, V1.02, 12.2009

2 Before you begin - safety information

2

The information provided in this manual supplements the product manual. Carefully read the product manual before using the product.

2.1 Qualification of personnel

Only appropriately trained persons who are familiar with and understand the contents of this manual and all other pertinent product documentation are authorized to work on and with this product. In addition, these persons must have received safety training to recognize and avoid hazards involved. These persons must have sufficient technical training, knowledge and experience and be able to foresee and detect potential hazards that may be caused by using the product, by changing the settings and by the mechanical, electrical and electronic equipment of the entire system in which the product is used.

All persons working on and with the product must be fully familiar with all applicable standards, directives, and accident prevention regulations when performing such work.

2.2 Intended use

The functions described in this manual are only intended for use with the basic product; you must read and understand the appropriate product manual.

The product may only be used in compliance with all applicable safety regulations and directives, the specified requirements and the technical data.

Prior to using the product, you must perform a risk assessment in view of the planned application. Based on the results, the appropriate safety measures must be implemented.

Since the product is used as a component in an entire system, you must ensure the safety of persons by means of the design of this entire system (for example, machine design).

Operate the product only with the specified cables and accessories. Use only genuine accessories and spare parts.

Any use other than the use explicitly permitted is prohibited and can result in hazards.

Electrical equipment should be installed, operated, serviced, and maintained only by qualified personnel.

The product must NEVER be operated in explosive atmospheres (hazardous locations, Ex areas).

0198441113808, V1.02, 12.2009

2.3 Hazard categories

Safety instructions to the user are highlighted by safety alert symbols in the manual. In addition, labels with symbols and/or instructions are attached to the product that alert you to potential hazards.

Depending on the seriousness of the hazard, the safety instructions are divided into 4 hazard categories.

⚠ 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, serious injury, or equipment damage.

⚠ CAUTION

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

CAUTION

CAUTION used without the safety alert symbol, is used to address practices not related to personal injury (e.g. **can result** in equipment damage).

0198441113808, V1.02, 12.2009

2.4 Basic information

⚠ DANGER

UNINTENDED CONSEQUENCES OF EQUIPMENT OPERATION

When the system is started, the drives are usually out of the operator's view and cannot be visually monitored.

- Only start the system if there are no persons in the hazardous area.

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

⚠ WARNING

LOSS OF CONTROL

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

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

1) For USA: Additional information, refer to NEMA ICS 1.1 (latest edition), Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control and to NEMA ICS 7.1 (latest edition), Safety Standards for Construction and Guide for Selection, Installation for Construction and Operation of Adjustable-Speed Drive Systems.

0198441113808, V1.02, 12.2009

2.5 Standards and terminology

Technical terms, terminology and the corresponding descriptions in this manual are intended to use the terms or definitions of the pertinent standards.

In the area of drive systems, this includes, but is not limited to, terms such as "safety function", "safe state", "fault", "fault reset", "failure", "error", "error message", "warning", "warning message", etc.

Among others, these standards include:

- IEC 61800 series: "Adjustable speed electrical power drive systems"
- IEC 61158 series: "Industrial communication networks - Fieldbus specifications"
- IEC 61784 series: "Industrial communication networks - Profiles"
- IEC 61508 series: "Functional safety of electrical/electronic/programmable electronic safety-related systems"

Also see the glossary at the end of this manual.

0198441113808, V1.02, 12.2009

3 Basics

3

3.1 Message types

DeviceNet defines several message types for communication. The product described here uses the "Explicit Message" and "I/O Message" message types.

Explicit Message

An "Explicit Message" is used for a write or read access to a single parameter. This chapter describes the use of the "Explicit Messages" on the basis of just a small number of messages since this type of communication can be used with all available parameters in the same way.

I/O Message

An "I/O Message" is used to transmit motion information. "I/O Messages"contain time-critical data that is specifically assembled for a use case. "I/O Messages" have a high-priority identifier so their transmission via the bus has top priority.

Command processing: Transmit data and receive data

The master sends a command to the drive system (slave) to execute a motion command, activate operating modes or request information from the slave. The slave executes the command and acknowledges it with a response message that may contain an error message if an error occurred.

The master device can send new commands as soon as it has received acknowledgement concerning the current command. Acknowledgement information and error messages are included in the transmitted data in bit-coded form.

The master must then continuously monitor for completion of the processing command by evaluating the receive data of the slave.

0198441113808, V1.02, 12.2009

3.2 Data structure

The data frame with transmit and receive data and all byte, word and double-word values is shown in hexadecimal notation. Hexadecimal values are indicated as such by means of an "h" behind the numerical value, for example, "31_h", decimal values have no special identification. Note the different counting format of bits (0 ... 7, right to left) and bytes (0-xx, left to right)

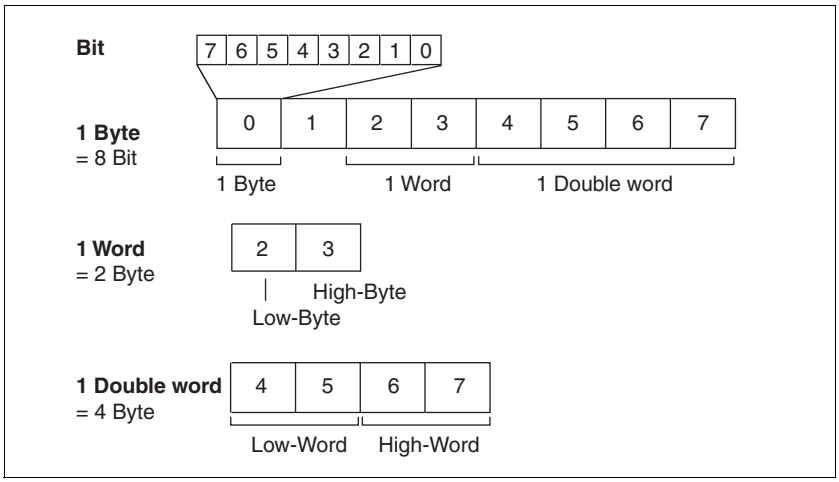

Figure 3.1 General data structure from the bit to the double word

The figure shows the bus view.

In the case of the Big Endian format, the bytes are transmitted one after the other from 0 to 7.
In the case of the Little Endian format, low-byte / high-byte and low-word / high-word are transmitted in a reversed way.

Byte sequence

There are two type of byte sequence: Little Endian / Intel format and Big Endian / Motorola format.

Unless explicitly specified otherwise, this manual uses the Big Endian format from the perspecitive of the DeviceNet scanner.

The byte sequence is entered in "Big Endian" format (Motorola format) in the fieldbus scanner, but is transmitted via the bus in "Little Endian" format (Intel format). The representation is therefore different on the fieldbus scanner and on the fieldbus monitor!

The data is transmitted via the bus left-aligned in Little Endian format, i.e. numerical values greater than one byte are transmitted with the lowest-value byte first.

0198441113808, V1.02, 12.2009

3.3 Communication via "Explicit Message"

A single parameter (DeviceNet-specific or vendor-specific) is read or written using an Explicit Message.

This chapter provides an example of communication via Explicit Messages in the form of a read request and a write request. This type of communication is identical for all parameters.

3.3.1 Reading a parameter

Task The master (MAC ID 2) is to read the parameter `n_act` of the slave (MAC ID 10).

- Class.Instance.Attribute = 130.1.8 = 82_h.01_h.08_h

Transmit data

Data 0	Data 1	Data 2	Data 3	Data 4
$0A_h$	$0E_h$	82_h	01_h	08_h

Data 0: $0A_h$ = Frag = 0, XID = 0, Destination MAC ID 0A
Data 1: $0E_h$ = Service Code = Get_Attribute_Single Request
Data 2: 82_h = Class 130
Data 3: 01_h = Instance 1
Data 4: 08_h = Attribute 8

Receive data

Data 0	Data 1	Data 2	Data 3
02_h	$8E_h$	02_h	58_h

Data 0: 02_h = Frag = 0, XID = 0, Destination MAC ID 2
Data 1: $8E_h$ = Get_Attribute_Single successful Response
Data 2 ... 3: 0258_h = 600

3.3.2 Writing a parameter

Task The master (MAC ID 2) is to set the parameter `RAMPsym` of the slave (MAC ID 10) to the value 1000.

- Class.Instance.Attribute = 106.1.1 = $6A_h$.01_h.01_h
- Value = 1000 = $03E8_h$

Transmit data

Data 0	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6
$0A_h$	10_h	$6A_h$	01_h	01_h	03_h	$E8_h$

Data 0: $0A_h$ = Frag = 0, XID = 0, Source MAC ID 10
Data 1: 10_h = Service Code = Set_Attribute_Single
Data 2: $6A_h$ = Class 106
Data 3: 01_h = Instance 1
Data 4: 01_h = Attribute 1
Data 5 ... 6: $03E8_h$ = Value 1000

Receive data

Data 0	Data 1
02_h	90_h

Data 0: 02_h = Frag = 0, XID = 0, Destination MAC ID = 02
Data 1: 90_h = Set_Attribute_Single successful Response

0198441113808, V1.02, 12.2009

3.3.3 Synchronous errors

If a write or read command is unsuccessful, the product responds with an error message (Error Response). The transmitted error number provides information on the exact cause.

Task The master (MAC ID 2) is to set the parameter `n_act` of the slave (MAC ID 10) to a random value.

- Class.Instance.Attribute = 130.1.8 = 82_h.01_h.08_h
- Value = any

Transmit data

Data 0	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6
$0A_h$	10_h	82_h	01_h	08_h	xx_h	xx_h

Data 0: $0A_h$ = Frag = 0, XID = 0, Source MAC ID 10
Data 1: 10_h = Service Code = Set_Attribute_Single
Data 2: 82_h = Class 130
Data 3: 01_h = Instance 1
Data 4: 08_h = Attribute 8
Data 5 ... 6: any

Receive data

Data 0	Data 1	Data 2	Data 3
02_h	94_h	$0E_h$	FF_h

Data 0: 02_h = Frag = 0, XID = 0, Destination MAC D 2
Data 1: 94_h = Set_Attribute_Single Error Response
Data 2: $0E_h$ = Error Code
Data 3: FF_h = Additional code (object-specific)
Data 2 ... 3: $0EFF_h$ = Attribute not settable

A list of synchronous error messages in data byte 2 can be found in chapter 7 "Diagnostics and troubleshooting".

0198441113808, V1.02, 12.2009

3.4 Communication via "I/O Message"

An "I/O Message" is used for realtime exchange of process data. This type of connection is ideal for positioning mode. Transmission is very fast because the data is sent without administration data and an acknowledgement from the recipient is not required.

The master can control the operating state of the slave by means of "I/O Messages", for example, enable and disable the power stage, trigger a "Quick Stop", reset errors and activate operating modes.

Changing operating states and activating operating modes must be done separately. An operating mode can usually only be activated if the operating state is already **6** Operation Enabled.
A new operating mode can only be activated when the motor is at a standstill.

Assembly

I/O messages contain an assembly of various parameters that are transmitted in a single message.

The following assemblies are permanently defined in the "Position Controller Profile":

- Command Message (100)
- Response Message (110)

The following assemblies are permanently defined in the "Drive Profile Lexium":

- Standard Output Assembly 101
- Standard Input Assembly 111
- Extended Output Assembly 102
- Extended Input Assembly 112

Polled I/O Connection

The assemblies are used in a polled I/O connection. A polled I/O connection is initiated by the master with a poll command. The slave responds with a poll response.

0198441113808, V1.02, 12.2009

3.4.1 "Position Controller Profile" assemblies

The "Position Controller Profile" supports the following formats:

Command Message

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	Enable	-	Hard Stop	Smooth Stop	Direction	Incre-mental	-	LoadData/ StartProfile
Byte 1	Command Data 1							
Byte 2	Command Axis Number			Command Message Type				
Byte 3	Command Data 2							
Byte 4	Command Data 3							
Byte 5	Command Data 4							
Byte 6	Command Data 5							
Byte 7	Command Data 6							

Figure 3.2 Command Message

The following table shows the structure of byte 0:

bit	Name	Function
0	LoadData / Start-Profile	Handshake for a "Command Message" To trigger a "Command Message", you require a rising edge at the "LoadData / StartProfile" bit. To obtain data via a "Response message", you do not require a rising edge at the "LoadData / StartProfile" bit.
1	Start Block	No function
2	Incremental	0: Absolute movement 1: Relative movement
3	DIR	0: Negative direction of movement 1: Positive direction of movement
4	Smooth Stop	0: Clear "Halt" 1: Set "Halt"
5	Hard Stop	0->1: Set "Quick Stop" 1->0: Reset "Quick Stop" 1)
6	RegArm	No function
7	Enable	Enable and disable the power stage

1) Only possible when the drive is at a standstill and if the "Quick Stop was triggered by a 0->1 edge."

The following table shows the available types of "Command Messages":

Type	Command Message
01_h	Target position
02_h	Target velocity
03_h	Acceleration
04_h	Deceleration
$1A_h$	Position Controller Supervisor Attribute
$1B_h$	Position Controller Attribute

0198441113808, V1.02, 12.2009

Response Message

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 0	Enable	-	-	Current Direction	General Fault	On Target Position	-	Profile in Progress
Byte 1	Response Data 1							
Byte 2	Load Complete	-	FE Fault	Negative Limit	Positive Limit	-	-	-
Byte 3	Response Axis Number			Response Message Type				
Byte 4	Response Data 2							
Byte 5	Response Data 3							
Byte 6	Response Data 4							
Byte 7	Response Data 5							

Figure 3.3 Response Message

The following table shows the structure of byte 0:

bit	**bit**	**Function**
0	ProfileInProgress	1: Operating mode active (x_end=0)
1	BlockInExecution	No function
2	OnTargetPosition	1: Target position reached
3	General Fault	Is set in operating states 2, 3, 7 [1)], 8 and 9 [2)]
4	Current Direction	0: Negative direction of movement 1: Positive direction of movement
5	HomeLevel	No function
6	RegLevel	No function
7	Enable	1: Power stage enabled

1) Not for "Quick Stop via the fieldbus."
2) The operating states are described in the product manual.

The following table shows the structure of byte 1:

bit	**Name**	**Function**
0	LoadData / Start-Profile	No function
1	FwdLimit	No function
2	RevLimit	No function
3	PositiveLimit	1: Error positive limit switch (LIMP)
4	NegativeLimit	1: Error negative limit switch (LIMN)
5	FE Fault	1: Following error
6	BlockFault	No function
7	LoadComplete	Handshake

0198441113808, V1.02, 12.2009

The following table shows the available types of "Response Messages":

Type	Response Message
01_h	Actual position
03_h	Actual Velocity
14_h	Command/Response Error
$1A_h$	Position Controller Supervisor Attribute
$1B_h$	Position Controller Attribute

0198441113808, V1.02, 12.2009

3.4.2 "Drive Profile Lexium" assemblies

Output - Input Output and input refer to the direction of data transmission from the perspective of the master.

- Output: Commands from the master to the slave
- Input: Status messages from the slave to the master

3.4.2.1 Output Assemblies

Standard Output Assembly

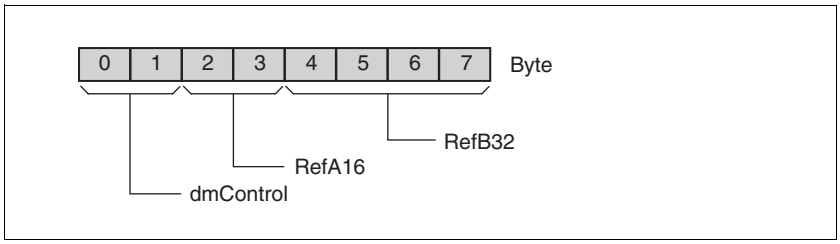

Figure 3.4 Output Assembly 101

Extended Output Assembly

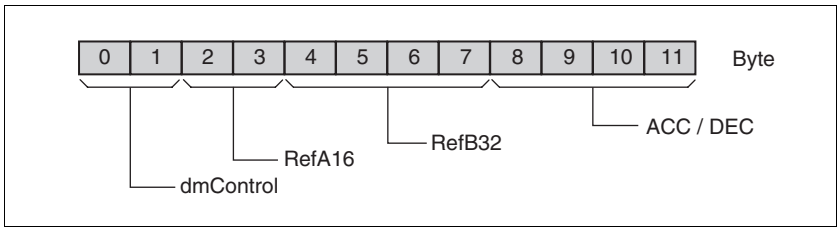

Figure 3.5 Output Assembly 102

dmControl The word "dmControl" is used to set the operating state and the operating mode.

See chapters 6.1.2 "Changing the operating state" and 6.2.2 "Starting and changing an operating mode" for a detailed description of the bits.

RefA16 The word "RefA16" is used to set the first value for the operating mode. The meaning depends on the operating mode; it is described in the chapters on the individual operating modes.

RefB32 The double word "RefB32" is used to set the second value for the operating mode. The meaning depends on the operating mode; it is described in the chapters on the individual operating modes.

ACC/DEC The double word "ACC/DEC" is used to set the acceleration and deceleration. The value corresponds to the parameter `RAMPaccdec`.

0198441113808, V1.02, 12.2009

3.4.2.2 Input Assemblies

Standard Input Assembly

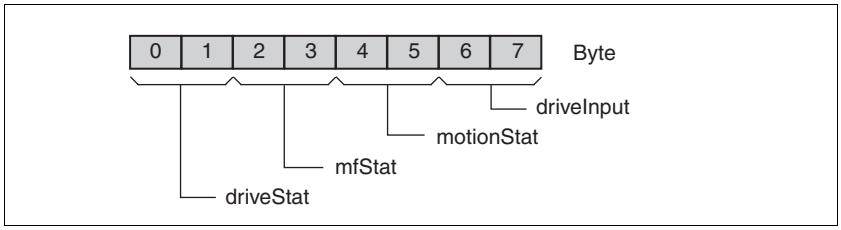

Figure 3.6 Input Assembly 111

Extended Input Assembly

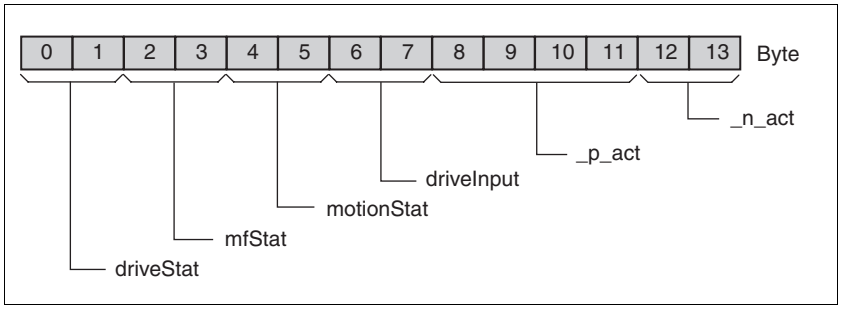

Figure 3.7 Input Assembly 112

driveStat The current operating state is indicated with the "driveStat" word.

For a detailed description of the bits, see chapter 6.1.1 "Indication of the operating state".

mfStat The word "mfStat" is used to indicate the current operating mode.

For a detailed description of the bits, see chapter 6.2.1 "Indicating and monitoring the operating mode".

motionStat The word "motionStat" is used to provide information on the motor and profile generator.

Bit	Meaning
0 ... 5	Reserved
6	MOTZ: Motor at a standstill
7	MOTP: Motor movement in positive direction
8	MOTN: Motor movement in negative direction
9	PWIN: Motor within position window
10	Reserved
11	TAR0: Profile generator at standstill
12	DEC: Profile generator decelerates
13	ACC: Profile generator accelerates
14	CNST: Profile generator moves at constant velocity
15	Reserved

0198441113808, V1.02, 12.2009

driveInput The word "driveInput" is used to indicate the status of the digital signal inputs.

bit	Signal	Factory setting
0	DI0	Signal input function Freely Available
1	DI1	Signal input function Reference Switch (REF)
2	DI2	Signal input function Positive Limit Switch (LIMP)
3	DI3	Signal input function Negative Limit Switch (LIMN)
4	DI4	Signal input function Freely Available
5	DI5	Signal input function Freely Available
6 ... 15	-	Reserved

_p_act The double word "_p_act" is used to provide information on the current motor position. The value corresponds to the parameter _p_act.

_n_act The word "_n_act" is used to provide information on the current velocity. The value corresponds to the parameter _n_act.

0198441113808, V1.02, 12.2009

3.5 Handshake with the "Mode Toggle" bit

Availability Handshake with the "Mode Toggle" bit is only available with the "Drive Profile Lexium".

Mode Toggle Synchronized processing can be carried out with the transmit data in byte "modeCtrl" bit "Mode Toggle" and the receive data in byte "modeStat" bit "Mode Error" and bit "Mode Toggle". Synchronized processing means that the master waits for feedback messages from the slave so it can respond appropriately.

Example 1: Positioning The master starts a positioning movement. At points in time t_1, t_2 ..., the master checks the responses from the slave. It waits for the end of positioning. The end is identified by bit "x_end" = 1.

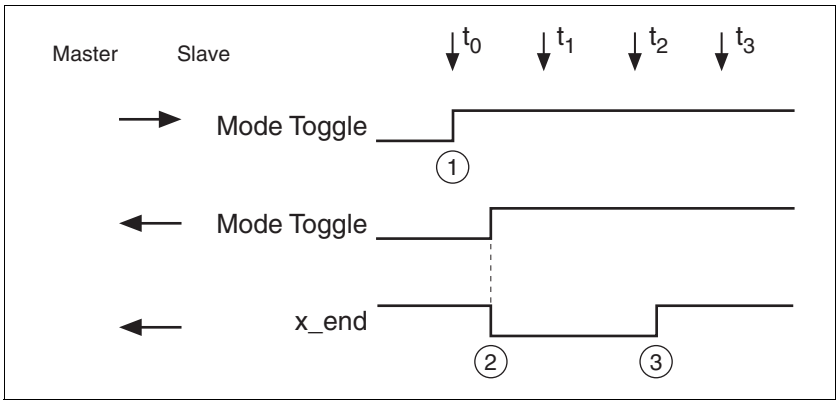

Figure 3.8 Mode Toggle Handshake

(1) Master starts positioning with "Mode Toggle" = 1
(2) Slave signals that positioning is running with "Mode Toggle" = 1; at the same time "x_end" = 0
(3) Slave signals end of positioning with "x_end" = 1

Example 2: Short-time positioning The master starts a positioning movement that will only take a very short time. The duration is shorter than the request cycle of the master. At point in time t_1 the movement is already complete. Bit "x_end" does not allow the master to detect whether the movement is already complete or has not yet been started. However, it can identify this with the "Mode Toggle" bit.

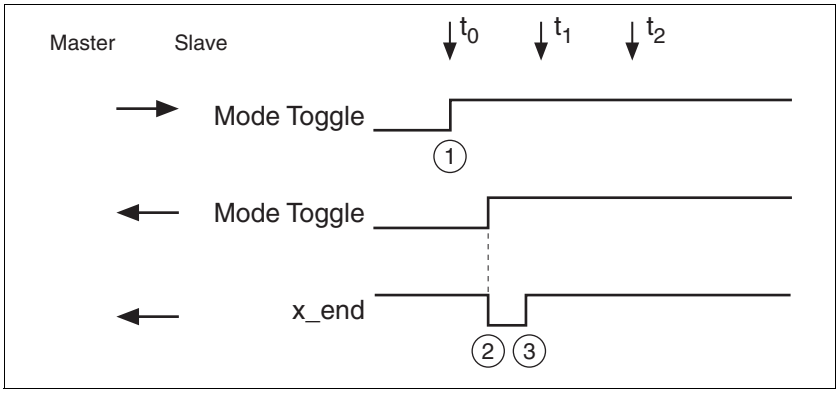

Figure 3.9 Mode Toggle Handshake, short movement

(1) Master starts positioning with "Mode Toggle" = 1
(2) Slave signals that positioning is running with "Mode Toggle" = 1; at the same time "x_end" = 0
(3) Slave signals end of positioning with "x_end" = 1

0198441113808, V1.02, 12.2009

4 Installation

4

⚠ WARNING

SIGNAL AND DEVICE INTERFERENCE

Signal interference can cause unexpected responses of device.

- Install the wiring in accordance with the EMC requirements.
- Verify compliance with the EMC requirements.

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

4.1 Installation of the module

CAUTION

DESTRUCTION DUE TO ESD

Electrostatic discharge (ESD) can cause immediate or later destruction of the module or the device.

- Use suitable ESD measures (IEC 61340-5-2) when handling the module.
- Do not touch any internal components.

Failure to follow these instructions can result in equipment damage.

▶ Install the module according to the instructions in the product manual.

Description	Order no.
DeviceNet fieldbus module with Open Style Connection (female)	VW3M3301

0198441113808, V1.02, 12.2009

4.2 Electrical installation

Cable specifications

For details on the cable specification, see chapter 1.1 "DeviceNet technology".

- ▶ Use equipotential bonding conductors.
- ▶ Use pre-assembled cables to reduce the risk of wiring errors.
- ▶ Verify that wiring, cables and connected interfaces meet the PELV requirements.

Terminating resistor

Both ends of the entire bus system must be terminated with a terminating resistor of 120 Ω between CAN_H and CAN_L each.

Pin assignment

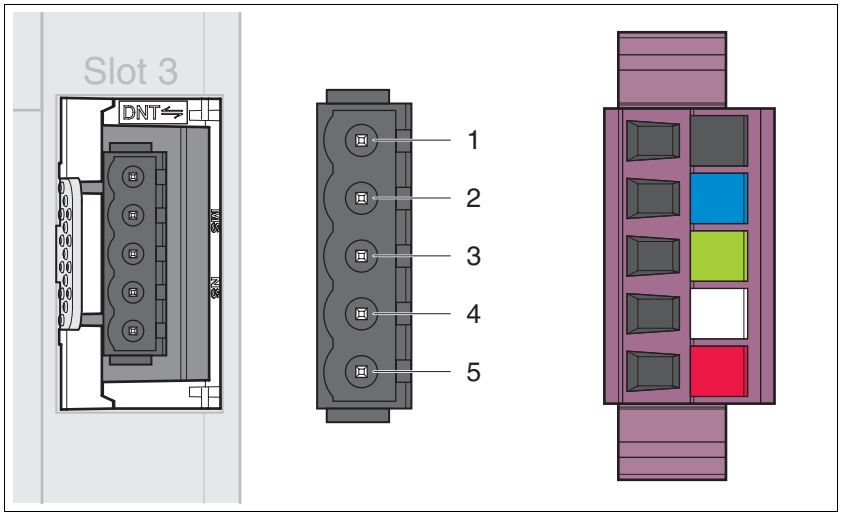

Figure 4.1 Pin assignment

Pin	Signal	Meaning
1	V-	Reference potential to V+
2	CAN_L	Fieldbus
3	SHLD	Shield connection
4	CAN_H	Fieldbus
5	V+	Bus supply voltage

0198441113808, V1.02, 12.2009

5 Commissioning

5

⚠ WARNING

LOSS OF CONTROL

The product is unable to detect an interruption of the network link if connection monitoring is not active.

- Verify that connection monitoring is on.
- The shorter the time for monitoring, the faster the detection of the interruption.

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

⚠ WARNING

UNINTENDED OPERATION

- Do not write values to reserved parameters.
- Do not write values to parameters unless you fully understand the function.
- Run initial tests without coupled loads.
- Verify that the system is free and ready for the movement before changing parameters.
- Verify the use of the word sequence with fieldbus communication.
- Do not establish a fieldbus connection unless you have fully understood the communication principles.

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

Using the library considerably facilitates controlling the device. The library is available for download from the Internet.
http://www.schneider-electric.com

5.1 Commissioning the device

For installation in the network, the device must first be properly installed (mechanically and electrically) and commissioned.

Commission the device as per product manual. This prepares the device for operation in the network.

0198441113808, V1.02, 12.2009

5.2 "First Setup"

A "First Setup" is required when the controller supply voltage is switched on for the first time or after the factory settings have been restored.

Switching on the device

■ The power stage supply voltage is switched off.

▶ Disconnect the product from the the fieldbus during commissioning in order to avoid conflicts by simultaneous access.

▶ Switch on the controller supply voltage.

◁ The device goes through an initialization routine, all LEDs are tested, all segments of the 7-segment display and the LEDs light up.

After the initialization, the fieldbus interface must be configured. You must assign a unique network address to each device. The transmission rate (baud rate) must be the same for all devices in the network.

▶ Set the network address.

The network address is stored in the parameter `DVNaddress` (*dnAd*).

▶ Set the transmission rate.

The transmission rate is stored in the parameter`DVNbaud` (*dnbd*).

Restarting the device

A restart of the device is required for the changes to become effective. After the restart, the device is ready for operation. The device is in the operating mode Jog.

Drive profile

The "Position Controller Profile" is preset.

If you want to use the "Drive Profile Lexium", you must set it using the configuration software RSNetWorx.
In addition, the "Drive Profile Lexium" must be activated by means of a vendor-specific parameter. The parameter must be written via the DeviceNet fieldbus.
Address: 127.1.8
Value: 1

0198441113808, V1.02, 12.2009

6 Operation

6

⚠ WARNING

UNINTENDED OPERATION

- Do not write values to reserved parameters.
- Do not write values to parameters unless you fully understand the function.
- Run initial tests without coupled loads.
- Verify that the system is free and ready for the movement before changing parameters.
- Verify the use of the word sequence with fieldbus communication.
- Do not establish a fieldbus connection unless you have fully understood the communication principles.

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

The chapter "Operation" describes the basic operating states, operating modes and functions of the device.

Using the library considerably facilitates controlling the device. The library is available for download from the Internet.
http://www.schneider-electric.com

0198441113808, V1.02, 12.2009

6.1 Operating states

6.1.1 Indication of the operating state

6.1.1.1 Drive profile "Position Controller Profile"

The current operating state is indicated by means the "Response Message", byte 0, bit 3 and bit 7.

Bit 7 Enable	Bit 3 General Fault	Operating state
0	0	**2** Not Ready To Switch On
0	0	**3** Switch On Disabled
0	0	**4** Ready To Switch On
0	0	**5** Switched On
1	0	**6** Operation Enabled
1	1	**7** Quick Stop Active
0	1	**8** Fault Reaction Active
0	1	**9** Fault

6.1.1.2 "Drive Profile Lexium"

The current operating state is indicated with the "driveStat" word.

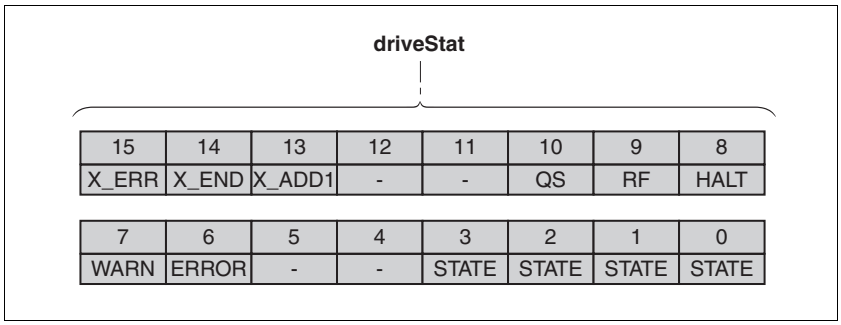

Figure 6.1 Structure of driveStat

Bit	Name	Meaning
0 ... 3	STATE	Current operating state (binary coded)
4 ... 5	-	Reserved
6	ERROR	An error has occurred (error classes 1 ... 3)
7	WARN	A warning has occurred (error class 0)
8	HALT	"Halt" is active
9	RF	Homing valid
10	QS	"Quick Stop" is active
11 ... 12	-	Reserved
13	X_ADD1	Operating mode-dependent
14	X_END	Operating mode terminated
15	X_ERR	Operating mode terminated with error

0198441113808, V1.02, 12.2009

6.1.2 Changing the operating state

6.1.2.1 Drive profile "Position Controller Profile"

You can change the operating state with the "Command Message" byte 0 bit 5 and bit 7.

Byte 0	Operating states
Bit 7 Enable	0->1: Enable power stage Drive switches to operating state **6** Operation Enabled. 1 -> 0: Disable power stage and reset errors Drive switches to operating state **4** Ready To Switch On.
Bit 5 Hard Stop	0->1: Trigger "Quick Stop" Drive switches to operating state **7** Quick Stop Active 1->0: Reset "Quick Stop" Drive switches to operating state **6** Operation Enabled

An error is reset using the vendor-specific attribute 103 of the object "Position Controller 25_h" value =1.

Operating state during error	Transition to operating state
7 Quick Stop Active	**6** Operation Enabled
9 Fault	**4** Ready To Switch On

0198441113808, V1.02, 12.2009

6.1.2.2 "Drive Profile Lexium"

Bits 8 ... 15 of the word "dmControl" are used to set the operating state.

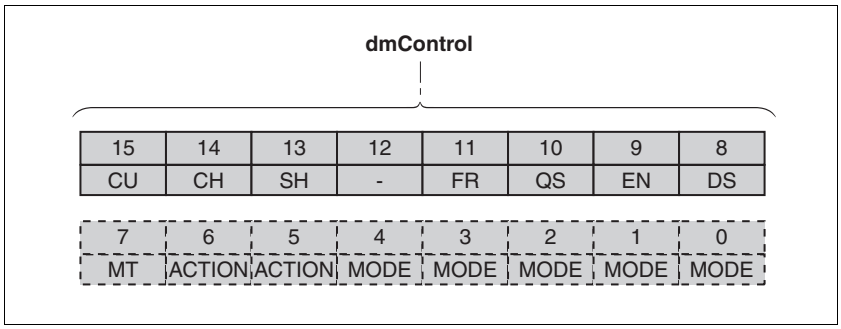

Figure 6.2 Structure dmControl bits 8 ... 15

Bit	Name	Meaning	Operating state
8	DS	Disable power stage	**6** Operation Enabled -> **4** Ready To Switch On
9	EN	Enable power stage	**4** Ready To Switch On -> **6** Operation Enabled
10	QS	Executing a "Quick Stop"	**6** Operation Enabled -> **7** Quick Stop Active
11	FR	Execute "Fault Reset"	**7** Quick Stop Active -> **6** Operation Enabled **9** Fault -> **4** Ready To Switch On
12	-	Reserved	Reserved
13	SH	Execute "Halt"	**6** Operation Enabled
14	CH	Clear "Halt"	**6** Operation Enabled
15	CU	Resume operating mode interrupted by "Halt"	**6** Operation Enabled

In the case of an access, the bits respond to a 0->1 change to trigger the corresponding function.

If a request for changing the operating state is not successful, this request is ignored. There is no error response.

Ambivalent bit combinations are treated in accordance with the following priority list (highest priority bit 8, lowest priority bit 14 and bit 15):

- Bit 8 (disable power stage) prior to bit 9 (enable power stage)
- Bit 10 ("Quick Stop") prior to bit 11 ("Fault Reset")
- Bit 13 (execute "Halt") prior to bit 14 (clear "Halt") and bit 15 (resume operating mode interrupted by "Halt")

0198441113808, V1.02, 12.2009

6.2 Operating modes

6.2.1 Indicating and monitoring the operating mode

6.2.1.1 Drive profile "Position Controller Profile"

You can indicate the current operating mode using attribute 3 and attribute 100 of the object "Position Controller 25_h".

6.2.1.2 "Drive Profile Lexium"

The word "mfStat" is used to indicate the current operating mode.

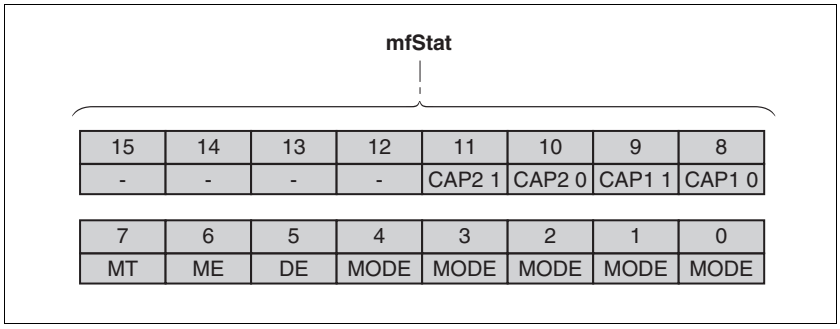

Figure 6.3 Structure mfStat

bit	Name	Description
0 ... 4	MODE	Indicates the current operating mode Value 01_h: Profile Position Value 03_h: Profile Velocity Value 04_h: Profile Torque Value 06_h: Homing Value $1F_h$: Jog Value $1E_h$: Electronic Gear Value $1D_h$: Motion Sequence
5	DE	The "DE" bit relates to parameters that are independent of "Mode Toggle" (MT). The "DE" bit is set if a data value in the process data channel is invalid.
6	ME	The "ME" bit relates to parameters that are dependent on "Mode Toggle" (MT). The "ME" bit is set if a request from a master (starting an operating mode) was rejected.
7	MT	Handshake via "Mode Toggle"
8 ... 9	CAP1	Bit 0 and bit 1 of parameter `_Cap1Count`
10 ... 11	CAP2	Bit 0 and bit 1 of parameter `_Cap2Count`
12 ... 15	-	Reserved

0198441113808, V1.02, 12.2009

6.2.2 Starting and changing an operating mode

6.2.2.1 Drive profile "Position Controller Profile"

The operating mode Profile Position (CIP: Position mode) or the operating mode Profile Velocity (CIP: Velocity mode) can be set via attribute 3 of the object "Position Controller 25_h".

In addition, you can use attribute 100 of the object "Position Controller 25_h" to set the vendor-specific operating mode Homing.

Two separate values are saved internally for the velocity for the operating modes Profile Position and Profile Velocity. This way, the velocity value is not lost when the operating mode is changed.

6.2.2.2 "Drive Profile Lexium"

Bits 0 ... 7 in the word "dmControl" are used to set the operating mode.

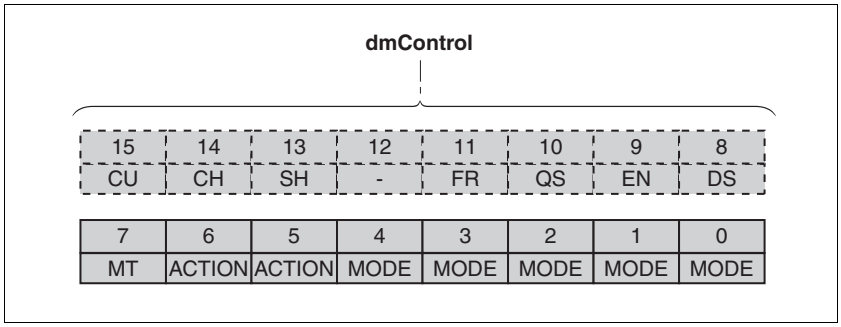

Figure 6.4 Structure dmControl bits 0 ... 7

bit	Name	Description
0 ... 4	MODE	Operating mode Value 01_h: Profile Position Value 03_h: Profile Velocity Value 04_h: Profile Torque Value 06_h: Homing Value $1F_h$: Jog Value $1E_h$: Electronic Gear Value $1D_h$: Motion Sequence
5 ... 6	ACTION	Operating mode-dependent
7	MT	Handshake via Mode Toggle

Handling of errors

If the state of the "Mode Toggle" bit is changed, this is considered as a request to start an operating mode or to change data of the current operating mode.

If the request cannot be processed, the "ME" bit is set in the word "modeStat". This has no effect on the selected operating mode. The corresponding error number can be read with the parameters `_ModeError` and `_ModeErrorInfo`.

The "ME" bit remains set until a new command is triggered.

0198441113808, V1.02, 12.2009

6.2.3 Operating mode Jog

Availability The operating mode is only available with the drive profile "Drive Profile Lexium".

Starting the operating mode The operating mode is set and started by means of an output assembly.

dmControl Bits 0 ... 6 MODE+ACTION	Reference value RefA16	Reference value RefB32
$1F_h$	Value 0: No movement Value 1: Slow movement in positive direction Value 2: Slow movement in negative direction Value 3: Fast movement in positive direction Value 3: Fast movement in negative direction	-

Status information The word "driveStat" provides information on the operating mode.

bit	Name	Meaning
13	X_ADD1	Reserved
14	X_END	0: Operating mode started 1: Operating mode terminated
15	X_ERR	0: No error 1: Error

Terminating the operating mode The operating mode is terminated when the motor is at a standstill and one of the following conditions is met:

- Value 0 RefA
- Stop caused by "Halt" or "Quick Stop"
- Stop caused by an error

0198441113808, V1.02, 12.2009

6.2.4 Operating mode Electronic Gear

Availability The operating mode is only available with the drive profile "Drive Profile Lexium".

Starting the operating mode The operating mode is set and started by means of an output assembly.

Method	dmControl Bits 0 ... 6 MODE+ACTION	Reference value RefA16	Reference value RefB32
Position synchronization without compensation movement	$1E_h$	As GEARdenom	As GEARnum
Position synchronization with compensation movement	$3E_h$	As GEARdenom	As GEARnum
Velocity synchronization	$5E_h$	As GEARdenom	As GEARnum

Status information The word "driveStat" provides information on the operating mode.

bit	Name	Meaning
13	X_ADD1	1: Reference velocity reached [1)]
14	X_END	0: Operating mode started 1: Operating mode terminated
15	X_ERR	0: No error 1: Error

1) Only with method Velocity synchronization and with active velocity window.

Terminating the operating mode The operating mode is terminated when the motor is at a standstill and one of the following conditions is met:

- Stop caused by "Halt" or "Quick Stop"
- Stop caused by an error

0198441113808, V1.02, 12.2009

6.2.5 Operating mode Profile Torque

Availability The operating mode is only available with the drive profile "Drive Profile Lexium".

Starting the operating mode The operating mode is set and started by means of an output assembly.

dmControl Bits 0 ... 6 MODE+ACTION	Reference value RefA16	Reference value RefB32
24_h	As `PTtq_target`	As `RAMP_tq_slope`

Status information The word "driveStat" provides information on the operating mode.

bit	Name	Meaning
13	X_ADD1	0: Target torque not reached 1: Target torque reached
14	X_END	0: Operating mode started 1: Operating mode terminated
15	X_ERR	0: No error 1: Error

Terminating the operating mode The operating mode is terminated when the motor is at a standstill and one of the following conditions is met:

- Stop caused by "Halt" or "Quick Stop"
- Stop caused by an error

0198441113808, V1.02, 12.2009

6.2.6 Operating mode Profile Velocity

6.2.6.1 Drive profile "Position Controller Profile"

The operating mode is started when the target velocity is assigned.

6.2.6.2 "Drive Profile Lexium"

Starting the operating mode

The operating mode is set and started by means of an output assembly.

dmControl Bits 0 ... 6 MODE+ACTION	Reference value RefA16	Reference value RefB32
23_h	As `PVv_target`	-

Status information

The word "driveStat" provides information on the operating mode.

bit	Name	Meaning
13	X_ADD1	0: Target velocity not reached 1: Target velocity reached
14	X_END	0: Operating mode started 1: Operating mode terminated
15	X_ERR	0: No error 1: Error

Terminating the operating mode

The operating mode is terminated when the motor is at a standstill and one of the following conditions is met:

- Stop caused by "Halt" or "Quick Stop"
- Stop caused by an error

0198441113808, V1.02, 12.2009

6.2.7 Operating mode Profile Position

6.2.7.1 Drive profile "Position Controller Profile"

A movement is triggered when you set a target position.

New values entered for velocity, acceleration and deceleration do not take effect while a movement is in process. These values only apply when a target position is set again.

6.2.7.2 "Drive Profile Lexium"

Starting the operating mode

The operating mode is set and started by means of an output assembly.

Method	dmControl Bits 0 ... 6 MODE+ACTION	Reference value RefA16	Reference value RefB32
absolute	01_h	As `PPv_target`	As `PPp_target`
Relative with reference to the currently set target position	21_h	As `PPv_target`	As `PPp_target`
Relative with reference to the current motor position	41_h	As `PPv_target`	As `PPp_target`

Status information

The word "driveStat" provides information on the operating mode.

bit	Name	Meaning
13	X_ADD1	0: Target position not reached 1: Target position reached
14	X_END	0: Operating mode started 1: Operating mode terminated
15	X_ERR	0: No error 1: Error

Terminating the operating mode

The operating mode is terminated when the motor is at a standstill and one of the following conditions is met:

- Target position reached
- Stop caused by "Halt" or "Quick Stop"
- Stop caused by an error

0198441113808, V1.02, 12.2009

6.2.8 Operating mode Homing

6.2.8.1 Drive profile "Position Controller Profile"

- Position setting

  Use attribute 13 of the object "Position Controller 25_h" for homing by means of position setting. The operating mode Homing does not have to be set when you do this.

- Reference movement

  Homing by means of a reference movement is performed via the vendor-specific attribute 100.

  Use the vendor-specific attribute 101 to select the homing method and start the operating mode.

6.2.8.2 "Drive Profile Lexium"

Starting the operating mode The operating mode is set and started by means of an output assembly.

Method	**dmControl Bits 0 ... 6 MODE+ACTION**	**Reference value RefA16**	**Reference value RefB32**
Position setting	06_h	-	As `HMp_setP`
Reference movement	26_h	As `HMmethod`	-

Status information The word "driveStat" provides information on the operating mode.

bit	**Name**	**Meaning**
13	X_ADD1	Reserved
14	X_END	0: Operating mode started 1: Operating mode terminated
15	X_ERR	0: No error 1: Error

Terminating the operating mode The operating mode is terminated when the motor is at a standstill and one of the following conditions is met:

- Homing successful
- Stop caused by "Halt" or "Quick Stop"
- Stop caused by an error

0198441113808, V1.02, 12.2009

6.2.9 Operating mode Motion Sequence

Availability The operating mode is only available with the drive profile "Drive Profile Lexium".

Starting the operating mode The operating mode is set and started by means of an output assembly.

Method	dmControl Bits 0 ... 6 MODE+ACTION	Reference value RefA16	Reference value RefB32
Start sequence	$1D_h$	Data set number	Value 1: Use data set number
Start individual data set	$3D_h$	Data set number	-

Status information The word "driveStat" provides information on the operating mode.

bit	Name	Meaning
13	X_ADD1	1: End of a sequence
14	X_END	0: Operating mode started 1: Operating mode terminated
15	X_ERR	0: No error 1: Error

Terminating the operating mode The operating mode is terminated when the motor is at a standstill and one of the following conditions is met:

- Individual data set terminated
- Individual data set of a sequence terminated (waiting for transition condition to be fulfilled)
- Sequence terminated
- Stop caused by "Halt" or "Quick Stop"
- Stop caused by an error

0198441113808, V1.02, 12.2009

0198441113808, V1.02, 12.2009

7 Diagnostics and troubleshooting

7

7.1 Fieldbus communication error diagnostics

A properly operating fieldbus is essential for evaluating operating and error messages.

Connections for fieldbus mode

If the product cannot be addressed via the fieldbus, first check the connections. The product manual contains the technical data of the device and information on network and device installation. Check the following:

- $24V_{dc}$ power supply
- Power connections to the device
- Fieldbus cable and fieldbus wiring
- Network connection to the device

You can also use the commissioning software for troubleshooting.

Fieldbus function test

If the connections are correct, check the settings for the fieldbus addresses. After correct configuration of the transmission data, test fieldbus mode.

▶ In addition to the master that knows the product via the EDS file and addressing, activate a bus monitor that, as a passive device, displays messages.

▶ Switch the supply voltage off and on.

▶ Observe the network messages that are generated briefly after the supply voltage is switched on. A bus monitor can be used to record the elapsed time between messages and the relevant information in the messages.

Addressing, parameterization

If it is impossible to connect to a device, check the following:

▶ Addressing

Each network device must have a unique address.

▶ Parameterization

"Vendor ID" and "Product Code" must match the values stored in the EDS file.

0198441113808, V1.02, 12.2009

7.2 Error indication

The last cause of error and the last 10 error messages are stored. You can display the last 10 error messages using the commissioning software and the fieldbus.

See the product manual for a list of the error numbers.

Asynchronous errors

Asynchronous errors are triggered by internal monitoring (for example, temperature) or by external monitoring (for example, limit switch). An error response is initiated if an asynchronous error occurs.

Asynchronous errors are indicated in the following way:

- Transition to operating state **7** Quick Stop Active or to operating state **9** Fault.
- Information in the word "driveStat"
- Error number is written to parameter `_LastError`

Synchronous errors

Synchronous errors occur as direct errors in response to a fieldbus command. They comprise, for instance:

- Error during execution of an action command or control command
- Parameter value outside the permissible value range
- Invalid action command or control command during processing
- Access to unknown parameter

For a detailed description of the synchronous errors, see chapter 7.2.1 "Synchronous errors".

0198441113808, V1.02, 12.2009

7.2.1 Synchronous errors

Explicit Error Response

If an Explicit Request message cannot be processed by the slave, the master receives an error message in the associated Explicit Response. This response message contains 2 bytes:

- General Error Code
- Additional Error Code

Errors can be read with object 100.1.1. If the general error code has the value =1F$_h$, the field "additional error code" contains vendor-specific error numbers in coded form.

Response during I/O connection

If an I/O command cannot be processed by the slave, bit 6 (ME) is set in the word "mfStat". This does not interrupt the current process (factory setting). The error response can be adjusted via the parameter `ErrorResp_bit_ME`.

To determine the cause of the error, the master can read the error number with the object 100.1.1 by means of an explicit access.

The error indication is reset when the next valid data frame is transmitted.

Table of general error codes

The error codes that can be contained in the "General Error Code" field are listed in the following table:

Error code	Name [1)]	Meaning
00$_h$	Success	The service was successfully executed by the specified object.
01$_h$	Connection failure	A connection-specific service has not been successful along the connection path.
02$_h$	Resource unavailable	Resources that the object required to execute the requested service were not available.
03$_h$	Invalid parameter value	See status code 20$_h$, which is the preferred value for this condition.
04$_h$	Path segment error	The path segment identifier or segment syntax could not be interpreted by the processing node. Path processing is terminated when a path segment error is detected.
05$_h$	Path destination unknown	The path refers to an object class, an instance or a structure element that is unknown or not contained in the processing node. Path processing is terminated if an error is detected that is due to an unknown path destination.
06$_h$	Partial transfer	Only part of the expected data was transmitted.
07$_h$	Connection lost	The connection for message transmission was interrupted.
08$_h$	Service not supported	The requested service was not implemented or was not defined for this object class or this instance.
09$_h$	Invalid attribute value	Invalid attribute data was detected
0A$_h$	Attribute list error	An attribute in the response "Get_Attribute_List" or "Set_Attribute_List" has a status that is not zero.
0B$_h$	Already in requested mode/ state	The object is already in the mode/state that was requested by the service
0C$_h$	Object state conflict	The object cannot execute the requested service in its current mode/state
0D$_h$	Object already exists	The requested instance of the object to be created already exists.
0E$_h$	Attribute not settable	A request to change an attribute that cannot be set was received.
0F$_h$	Privilege violation	Checking of an authorization/privilege has not been successful
10$_h$	Device state conflict	The requested service cannot be executed in the current mode/state of the device.

0198441113808, V1.02, 12.2009

Error code	Name [1)]	Meaning
11_h	Reply data too large	The data volume for transmission in the response buffer is larger than the allocated response buffer
12_h	Fragmentation of a primitive value	The service has specified an operation that results in fragmentation of the original data value, i.e. half a REAL data type.
13_h	Not enough data	The service did not deliver enough data to execute the specified operation.
14_h	Attribute not supported	The attribute specified in the request is not supported
15_h	Too much data	The service delivered more data than expected
16_h	Object does not exist	The specified object does not exist in the device.
17_h	Service fragmentation sequence not in progress	The fragmentation sequence for this service is not currently activated for this data.
18_h	No stored attribute data	The attribute data of this object were not saved before the requested service.
19_h	Store operation failure	The attribute data of this object was not saved, because an error occurred during the attempt.
$1A_h$	Routing failure, request packet too large	The request packet of the service was too large to be transmitted to the destination on a network along the path. The routing device had to cancel the service.
$1B_h$	Routing failure, response packet too large	The response packet of the service was too large to be transmitted from the destination on a network along the path. The routing device had to cancel the service.
$1C_h$	Missing attribute list entry data	The service did not provide an attribute in an attribute list that was required to allow the service to carry out the requested behavior.
$1D_h$	Invalid attribute value list	The service returns the list of the attributes provided with the status information for the invalid attributes.
$1E_h$	Embedded service error	An embedded service resulted in an error.
$1F_h$	Vendor specific error	An error with a vendor-specific error number has been detected. The "Additional Code" field (additional error code) of the response message contains the error number. If "Additional Code" contains the value "FE", the vendor-specific error number can be read with the object 100.1.1. If the "Additional Code" contains a value other than "FE", the error number can be directly generated by prepending "$A3_h$". Example: Error Code = $1F_h$ Additional Code = 08_h Error code = $A308_h$ (operating state **9** Fault).
20_h	Invalid parameter	A parameter assigned to the request was invalid. This code is used if a parameter does not meet the requirements of this specification and/or the requirements defined in an application object specification.
21_h	Write-once value or medium already written	An attempt was made either to write data again to a write-once medium (e. g. WORM drive, PROM) that already contained data, or to change a specified value that cannot be changed.
22_h	Invalid Reply Received	An invalid reply was received (for example, the service reply code is not the same as the request code or the reply is shorter than the expected minimum length of the reply). This status code may also be used for other causes of invalid responses.
23_h - 24_h		Reserved by CIP for future extensions
25_h	Key Failure in path	The key segment that was inserted in the path as the first segment does not match the destination module. The object-specific status indicates the part of the key test that has not been successful.
26_h	Path Size Invalid	The size of the path that was sent with the service request is either not large enough to allow for routing of the request, or it contained too much routing data.
27_h	Unexpected attribute in list	An attempt was made to set an attribute that cannot be set at this time.

0198441113808, V1.02, 12.2009

Error code	Name [1)]	Meaning
28_h	Invalid Member ID	The member ID specified in the request does not exist in the specified class/ instance/attribute
29_h	Member not settable	A request to change a member that cannot be set was received.
$2A_h$	Group 2 only server general failure	Service or attribute is not supported (attribute cannot be set)
$2B_h$ - CF_h	-	Reserved by CIP for future extensions
$D0_h$ -FF_h	Reserved for Object Class and service errors	This error code range is used to indicate errors related to object classes. Use this range only if none of the error codes listed in this table exactly represents the detected error.

1) See: The CIP Networks Library, Volume 1, Common Industrial Protocol, Appendix B

0198441113808, V1.02, 12.2009

0198441113808, V1.02, 12.2009

8 Object dictionary

8

This chapter describes the communication parameters supported by the product.

The following classes are supported:

- Identity Object (class 1)
- DeviceNet Object (class 3)
- Assembly Object (class 4)
- Connection Object (class 5)
- Position Controller Supervisor Object (class 36)
- Position Controller Object (class 37)
- Acknowledge Handler Object (class 43)
- Vendor-specific objects (classes 100 ... 199)

Acronyms

NV: Persistent (Non-Volatile)
V: Not persistent (Volatile)
RO: Read Only
RW: Read Write

Vendor-specific objects

The vendor-specific objects (parameters) are described in the product manual.

Structure of the address of an object:
Class.Instance.Attribute

0198441113808, V1.02, 12.2009

8.1 Identity Object (class 1)

The object contains the identification data of the product.

8.1.1 Class attributes

ID	Access	Name	Data type	Description	Value
1	Get (NV-RO)	Revision	UINT	Revision	1
2	Get (NV-RO)	Max instance	UINT	Greatest currently existing instance number of an object derived from this class	1

8.1.2 Instance attributes

ID	Access	Name	Data type	Description	Value
1	Get (NV-RO)	Vendor ID [1]	UINT	Unique vendor number	243 ($F3_h$)
2	Get (NV-RO)	Device Type [2]	UINT	Device family	16 (10_h) Position Controller
3	Get (NV-RO)	Product Code	UINT	Unique device type	2565 (0A 05_h)
4	Get (NV-RO)	Revision Major Revison Minor Revision	STRUCT of USINT USINT	Revision of device	xx.xx
5	Get (V-RO)	Status [3]	WORD	Summarized device status	
6	Get (NV-RO)	Serial Number	UDINT	CIP serial number [4]	
7	Get (NV-RO)	Product Name [5]	SHORT_ STRING	Device name in text form	"Lexium 32"
8	Get (V-RO)	State	USINT	Current device state in state diagram	
10	Get/Set (NV-RW)	Heartbeat Interval	USINT	Interval between two heartbeat messages (in sec)	Default value is 0 (no Heartbeat message)

1) Vendor number assigned by
2) Corresponds to the ODVA device profile
3) Current device status; bits 8...11 contain the operating state
4) Corresponds to the product serial number
5) 32 characters maximum

0198441113808, V1.02, 12.2009

8.2 DeviceNet Object (class 3)

The object contains the communication parameters for the DeviceNet interface. There is one instance with ID 1.

8.2.1 Class attributes

ID	Access	Name	Data type	Description	Value
1	Get (NV-RO)	Revision	UINT	Revision	2
2	Get (NV-RO)	Max instance	UINT	Greatest currently existing instance number of an object derived from this class	1

8.2.2 Instance attributes

ID	Access	Name	Data type	Description	Value
1	Get/Set (NV-RW)	MAC ID	USINT	Device address	0...63
2	Get/Set (V-RW)	Baud Rate [1)]	USINT	Baud rate setting	0 = 125 kBaud 1 = 250 kBaud 2 = 500 kBaud
3	Get/Set (V-RW)	BOI	BOOL	Reaktion auf Bus Off Interrupt	0: CAN Controller remains in Bus Off (Default) 1: Controller reset and restart
4	Get/Set (V-RW)	Bus-Off Counter	USINT	Counter indicating the number of times the CAN Controller was in Bus Off Status Write access clears the counter	0...255
5	Get (V-RO)	Allocation Information	STRUCT of		
		Allocation Choice Byte [2)]	BYTE		
		Master's MAC ID	USINT	Detected master MAC ID	0...63
100	Get/Set (NV-RW)	Baud Rate [1)]	USINT	Baud rate setting	0 = 125 kBaud 1 = 250 kBaud 2 = 500 kBaud 3 = Autobaud (default)

1) The baud rate is automatically detected with Autobaud
2) See: The CIP Networks Library, Volume 3, DeviceNet Adaption of CIP, chapter 5-3

0198441113808, V1.02, 12.2009

8.3 Assembly Object (class 4)

This object is a container that contains one or more attributes of other objects. This way, multiple attributes can be transmitted from or to a slave simultaneously with a single connection.

In this context, the terms input data and output data are used:

- Outputs are commands from the network to the device
- Inputs are status messages from the device to the network

The following instances of the Assembly Object are implemented in the device:

Instance ID	Type	Name	Number of bytes
101	Output Assembly	Vendor-specific standard profile	8
111	Input Assembly	Vendor-specific standard profile	8
102	Output Assembly	Vendor-dependent extended profile	12
112	Input Assembly	Vendor-dependent extended profile	14

8.3.1 Class attributes

ID	Access	Name	Data type	Description	Values
1	Get (NV-RO)	Revision	UINT	Revision	2
2	Get (NV-RO)	Max instance	UINT	Greatest currently existing instance number of an object derived from this class	5

8.3.2 Common instance attributes

The object is static. The mapping can be read with attributes 1 and 2 (read only).

Attribute 3 is the standard data attribute.

The instance attributes supported are described in the following table:

ID	Access	Name	Data type	Description	Value
1	Get (NV-RO)	Number Of Members In List	UINT		See below
2	Get (NV-RO)	Member List [1)]	ARRAY of STRUCT	List of DeviceNet paths	See below
		Member Data Description	UINT	Size in bits	See below
		Member Path Size	UINT	Path size in bytes	See below
		Member Path	EPATH		See below
3	Get/Set (V-RW)	Data [2)]	ARRAY of BYTE	Data from/to device	
4	Get (NV-RO)	Size	UINT	Number of bytes in Attribute 3	See below

1) List of the members with data type and DeviceNet path to the attributes contained
2) Attribute 3 contains the Input Assemblies or the Output Assemblies. "Set" access is only possible with the output assemblies

0198441113808, V1.02, 12.2009

8.4 Connection Object (class 5)

The object manages access channels to or from the devices.

The product supports the "Predefined Master/Slave Connection Set".

8.4.1 Class attributes

ID	Access	Name	Data type	Description	Value
1	Get (NV-RO)	Revision	UINT	Revision	1
2	Get (NV-RO)	Max instance	UINT	Number of instances	5

8.4.2 Explicit message connection object

Instance ID 1 = predefined Explicit Connection
Instance ID 5 and 6 = dynamic Explicit Connection

This instance represents a point-to-point connection between two devices. In an explicit message, a **single** attribute of an object is transmitted to a device or read by the terminal device.

ID	Access	Name	Data type	Description	Value
1	Get (NV-RO)	Status	USINT	Status of object	0 = Non-existent 3 = Assembled 5 = Deleted
2	Get (NV-RO)	Instance type	USINT	I/O or Explicit Message	0 = Explicit Message
3	Get (NV-RO)	Transport Class Trigger	BYTE	Behavior of the connection	83_h = Class 3 Server
4	Get (NV-RO)	Produced Connection ID	UINT		11000xxxxxx xxxxxx = Node address
5	Get (NV-RO)	Consumed Connection ID	UINT		11100xxxxxx xxxxxx = Node address
6	Get (NV-RO)	Initial Comm Characteristics	BYTE		33_h Producer: Group 3 Consumer: Group 3
7	Get (NV-RO)	Produced Connection Size	UINT	Maximum number of bytes that are transmitted via this connection	44
8	Get (NV-RO)	Consumed Connection Size	UINT	Maximum number of bytes that are transmitted via this connection	44
9	Get/Set (NV-RW)	Expected Packet Rate	UINT	Time response of connection (ms)	2500
12	Get/Set (NV-RW)	Watchdog Timeout Action	USINT	Response after timeout	1 = Auto Delete 3 = Deferred Delete (Default = 1)
13	Get (NV-RO)	Produced Connection Path length	UINT	Length of attribute 14	0
14	Get/Set (NV-RW)	Produced Connection Path	EPATH		Zero
15	Get (NV-RO)	Consumed Connection Path length	UINT	Length of attribute 16	0
16	Get/Set (NV-RW)	Consumed Connection Path	EPATH		Zero

0198441113808, V1.02, 12.2009

ID	Access	Name	Data type	Description	Value
18	Get/Set (NV-RW)	Connection Time-out Multiplier	USINT	For Watchdog Timer to monitor the Expected Packet Rate [1)]	0

1) See: The CIP Networks Library, Volume 1, Common Industrial Protocol, chapter 3-4.4.18

Explicit messages must be confirmed. An error is confirmed with an error message.

8.4.3 Polled I/O Message Object

Instance ID 2 = Predefined Poll Connection
Instance ID 7 = Dynamic I/O Connection

A poll command message and a poll response message transmit multiple I/O data between a master and one or more slaves.

In a Poll I/O Connection, a DeviceNet master acts as a client and a DeviceNet slave as a server. The client sends commands to the server in a Poll Command message, the server returns status data to the client in a Poll Response message.[1]

ID	Access	Name	Data type	Description	Value
1	Get (NV-RO)	Status	USINT	Status of object	0 = Non-existent 3 = Assembled 5 = Deleted
2	Get (NV-RO)	Instance type	USINT	I/O or Explicit Message	1 = I/O Message
3	Get (NV-RO)	Transport Class Trigger	BYTE	Behavior of the connection [1)]	83_h = Class 3 Server
4	Get (NV-RO)	Produced Connection ID	UINT		01111xxxxxx xxxxxx = Node address
5	Get (NV-RO)	Consumed Connection ID	UINT		10xxxxxx101 xxxxxx = Node address
6	Get (NV-RO)	Initial Comm Characteristics	BYTE		01_h Producer: Group 1 Consumer: Group 2
7	Get (NV-RO)	Produced Connection Size	UINT	Maximum number of bytes that can be transmitted via this connection without fragmentation	8
8	Get (NV-RO)	Consumed Connection Size	UINT	Maximum number of bytes that can be transmitted via this connection without fragmentation	8
9	Get/Set (NV-RW)	Expected Packet Rate	UINT	Time response of the connection (ms) [2)]	1000
12	Get/Set (NV-RW)	Watchdog Timeout Action	USINT		0 = Transition to Timeout 1 = Auto Delete 2 = Auto Reset (Default = 0) [3)]
13	Get (NV-RO)	Produced Connection Path length	UINT	Length of attribute 14	6

1. See: The CIP Networks Library, Volume 3, DeviceNet Adaption of CIP, chapter 3-11

0198441113808, V1.02, 12.2009

ID	Access	Name	Data type	Description	Value
14	Get/Set (NV-RW)	Produced Connection Path	EPATH	Application objects whose data is produced via this connection. Factory settings: Position Controller	20 24 24 00 30 21_h
15	Get (NV-RO)	Consumed Connection Path length	UINT	Length of attribute 16	6
16	Get/Set (NV-RW)	Consumed Connection Path	EPATH	Application objects whose data is consumed via this connection Factory settings: Position Controller	20 24 24 00 30 20_h
18	Get/Set (NV-RW)	Connection Timeout Multiplier	USINT	For Watchdog Timer to monitor the Expected Packet Rate [4)]	0
100	Get/Set (NV-RW)	Polled I/O Input	USINT	Input Position Controller Profil Input Assembly Instanz	110 111, 112
101	Get/Set (NV-RW)	Polled I/O Output	USINT	Output Position Controller Profil Output Assembly Instanz	100 101, 102

1) See: The CIP Networks Library, Volume 1, Common Industrial Protocol, chapter 3-4.4.3
2) See: The CIP Networks Library, Volume 1, Common Industrial Protocol, chapter 3-4.5
3) See: The CIP Networks Library, Volume 1, Common Industrial Protocol, chapter 3-4.4.12
4) See: The CIP Networks Library, Volume 1, Common Industrial Protocol, chapter 3-4.4.18

0198441113808, V1.02, 12.2009

8.5 Position Controller Supervisor Object (class 36)

Attribute	ro/rw	CIP name	Data type	Description	Remarks
1	ro	Number of Attributes	USINT	Number of supported attributes	
3	ro	Axis Number	USINT	Axis Number	Value is 1
5	ro	General Fault	BOOL	General error that can be reset with a "Fault Reset".	Is set in operating states 2, 3, 7 [1)], 8 and 9.
6	rw	Command Message Type	USINT	Configuration "Command Message Type".	
7	rw	Response Message Type	USINT	Configuration "Response Message Type".	
100	rw	Vendor-specific error information	UINT	Vendor-specific error number of last "I/O Message" with an error	High Word: Message type with error [2)] Low Word: Vendor-specific error number.

1) Not for "Quick Stop via the fieldbus."
2) Value 0 means: Error in byte 0

8.6 Position Controller Object (class 37)

Attribute	ro/rw	CIP name	Data type	Description	Remarks
1	ro	Number of Attributes	USINT	Number of supported attributes	
2	ro	Attribute List	Array of USINT	List of supported attributes	
3	rw	Mode	USINT	Operating mode 0: Profile Position 1: Profile Velocity	You can set other operating modes using attribute 100.
6	rw	Target position	DINT	Target position in operating mode Profile Position Unit: [usr_p]	
7	rw	Target velocity	DINT	Target velocity Unit: [usr_v]	Negative numbers not permitted. See also attribute 23.
8	rw	Acceleration	DINT	Acceleration Unit: [usr_a]	A new value only becomes effective with the next movement.
9	rw	Deceleration	DINT	Deceleration Unit: [usr_a]	A new value only becomes effective with the next movement.
10	rw	Incremental Position Flag	BOOL	Movement 0: Absolute movement 1: Relative movement	Relative movement with reference to the last end position.
11	rw	Load Data/Profile Handshake	BOOL	Used to accept data of an "I/O Message" in the drive and to start a movement.	The bits "Load Data/Profile Handshake" and "Load Data Complete" are used as a handshake for the "I/O Message".
12	ro	On Target Position	BOOL	Target position reached	In operating mode Profile Position

0198441113808, V1.02, 12.2009

Attribute	ro/rw	CIP name	Data type	Description	Remarks
13	rw	Actual position	DINT	Current position Unit: [usr_p]	Reading the value gets the current position of the motor. Writing executes Position Setting of the Homing operating mode.
14	ro	Actual Velocity	DINT	Current velocity Unit: [usr_v]	Value is positive. See also attribute 23.
17	rw	Enable	BOOL	´0: Disable power stage 1: Enable power stage	0 -> 1: Reset error
20	rw	Smooth Stop	BOOL	Function "Halt"	
21	rw	Hard Stop	BOOL	Function "Quick Stop"	
23	rw	Direction	BOOL	Direction of movement 0: Negative direction 1 : Positive direction	Reading gets the current direction of movement of the motor. Writing sets the direction of movement for the operating mode Profile Velocity.
29	ro	Wrap Around	BOOL	Position overrun 1: Position overrun exists	
45	rw	Max Dynamic Following Error	DINT	Maximum permissible position deviation	Corresponds to parameter `MON_p_dif_load`
47	rw	Following Error	BOOL	Following error 1: Maximum permissible position deviation exceeded	
48	ro	Actual Following Error	DINT	Current position deviation	Corresponds to parameter `_p_dif_load`
56	ro	Positive Limit Triggered	BOOL	Positive limit switch triggered	
57	ro	Negative Limit Triggered	BOOL	Negative limit switch triggered	
58	ro	Load Data Complete	BOOL	Indicates that the data of the "I/O Message" have been accepted.	The bits "Load Data/Profile Handshake" and "Load Data Complete" are used as a handshake for the "I/O Message".
100	rw	ModeExt	USINT	Enhanced operating mode 0: Profile Position 1: Profile Velocity 100: Homing	
101	rw	Homing method	USINT	Homing method	Corresponds to parameter `HMmethod`
102	ro	Drive State	UDINT	Vendor-specific status word	Corresponds to parameter `_xStatus`
103	rw	Fault Reset	BOOL	1: Reset error	

0198441113808, V1.02, 12.2009

8.7 Acknowledge Handler Object (class 43)

The object manages the incoming messages via DeviceNet. It provides information on received confirmations, timeouts, repetitions, etc.

The object defines one instance.

8.7.1 Class attributes

ID	Access	Name	Data type	Description	Value
1	Get (NV-RO)	Revision	UINT	Revision	1
2	Get (NV-RO)	Max instance	UINT	Greatest currently existing instance number of an object derived from this class	1

8.7.2 Instance attributes

ID	Access	Name	Data type	Description	Value
1	Get/Set (V-RW)	Acknowledge Timer	UINT	Time to wait for confirmation before a transmission retry is required	1...65535 ms 0 = Invalid 16 = Default
2	Get/Set (V-RW)	Retry Limit	USINT	Number of Ack Timeouts before the application is informed of a "Retry_Limit_Reached" event	0...255 1=Default
3	Get/Set (V-RW)	COS Producing Connection Instance	UINT	Instance of connection that is informed via Ack handler events	Default: 4 (COS/Cyclic I/O Connection)
4	Get (V-RO)	Ack List Size	BYTE	Max. number of list entries in Ack List	Default: 1
5	Get (V-RO)	Ack List	BYTE, ARRAY of UINT	List of active connections that receive confirmations	Default: $\{01\ 04\ 00\}_h$
6	Get (V-RO)	Data with Ack Path List Size	BYTE	Max. number in attribute 7	Default: 1
7	Get (V-RO)	Data with Ack Path List	BYTE, ARRAY of - UINT - USINT - EPATH		Default: { 01 04 00 06 20 01 24 6D 30 03 $\}_h$

The attributes are stored in the volatile memory. The application can set the values for Acknowledge Timer, Retry Limit and Producing Connection Instance to values different from those specified.

0198441113808, V1.02, 12.2009

8.8 Network management

Device Heartbeat Message

The product supports the heartbeat protocol as per "The CIP Networks Library", Volume 3, DeviceNet Adaption of CIP, chapter 2-12.

The Heartbeat message cyclically sends the device status with the bit "DF". The cycle time is defined with the Identity Object (class 1), attribute 10, "Heartbeat Interval".

The following information is sent by the drive with the heartbeat message:

- Byte "Device Status": Identity Object (class 1) attribute 8
- Bit "SF": Communication error
- Bit "UF": Error caused by user commands
- Bit "DF": Internal error message

Device Shutdown Message

The device supports the shutdown protocol as per "The CIP Networks Library", Volume 3, DeviceNet Adaption of CIP, chapters 2-13. This message is generated if the device switches to offline status.

Two bytes in the shutdown message are reserved for the shutdown code. This code is vendor-specific; it is shown for the device in the following table:

Class ID	Instance ID	Shutdown Code	Description
1	1	4	Remote request: Reset Service for the Identity object (service code 5)
2	1	4	Remote request: Setting of the MAC ID via attribute 1 of the DeviceNet object
3	0	5	The device has been switched off due to CAN interface overflow.

Offline Connection Set

The product supports the Offline Connection Set as per "The CIP Networks Library, Volume 3, DeviceNet Adaption of CIP".

The configuration software uses commands of the Offline Connection Set to find nodes which are in the state "Communication Faulted" as per CIP. A new MAC ID can be assigned to such nodes by means of a command from the "Offline Connection Set".

The configuration tool and the nodes generate the following messages in the Offline Connection Set:

Name	Description
Identify Request/ Response Message, Multicast Protocol	This message is used to determine whether any nodes in a network segment are in "Communication Faulted" state. The corresponding nodes respond with a Response Message.
Identify Request/ Response Message, Point-To-Point Protocol	This message is used to identify a single node that is in the state "Communication Faulted". The corresponding node responds with a response message and a flashing NS-LED.
Who Communication Request/ Response Message	This message is used to poll the serial number and the vendor ID of a node that is in the state "Communication Faulted".
Change MAC ID Request/Response Message	This message is used to assign a new MAC ID to a node.

0198441113808, V1.02, 12.2009

0198441113808, V1.02, 12.2009

9 Glossary

9

9.1 Units and conversion tables

The value in the specified unit (left column) is calculated for the desired unit (top row) with the formula (in the field).

Example: conversion of 5 meters [m] to yards [yd]
5 m / 0.9144 = 5.468 yd

9.1.1 Length

	in	ft	yd	m	cm	mm
in	-	/ 12	/ 36	* 0.0254	* 2.54	* 25.4
ft	* 12	-	/ 3	* 0.30479	* 30.479	* 304.79
yd	* 36	* 3	-	* 0.9144	* 91.44	* 914.4
m	/ 0.0254	/ 0.30479	/ 0.9144	-	* 100	* 1000
cm	/ 2.54	/ 30.479	/ 91.44	/ 100	-	* 10
mm	/ 25.4	/ 304.79	/ 914.4	/ 1000	/ 10	-

9.1.2 Mass

	lb	oz	slug	kg	g
lb	-	* 16	* 0.03108095	* 0.4535924	* 453.5924
oz	/ 16	-	* $1.942559*10^{-3}$	* 0.02834952	* 28.34952
slug	/ 0.03108095	/ $1.942559*10^{-3}$	-	* 14.5939	* 14593.9
kg	/ 0.45359237	/ 0.02834952	/ 14.5939	-	* 1000
g	/ 453.59237	/ 28.34952	/ 14593.9	/ 1000	-

9.1.3 Force

	lb	oz	p	dyne	N
lb	-	* 16	* 453.55358	* 444822.2	* 4.448222
oz	/ 16	-	* 28.349524	* 27801	* 0.27801
p	/ 453.55358	/ 28.349524	-	* 980.7	* $9.807*10^{-3}$
dyne	/ 444822.2	/ 27801	/ 980.7	-	/ $100*10^{3}$
N	/ 4.448222	/ 0.27801	/ $9.807*10^{-3}$	* $100*10^{3}$	-

9.1.4 Power

	HP	W
HP	-	* 746
W	/ 746	-

0198441113808, V1.02, 12.2009

9.1.5 Rotation

	min^{-1} (RPM)	**rad/s**	**deg./s**
min^{-1} (RPM)	-	* π / 30	* 6
rad/s	* 30 / π	-	* 57.295
deg./s	/ 6	/ 57.295	-

9.1.6 Torque

	lb·in	**lb·ft**	**oz·in**	**Nm**	**kp·m**	**kp·cm**	**dyne·cm**
lb·in	-	/ 12	* 16	* 0.112985	* 0.011521	* 1.1521	* $1.129*10^6$
lb·ft	* 12	-	* 192	* 1.355822	* 0.138255	* 13.8255	* $13.558*10^6$
oz·in	/ 16	/ 192	-	* $7.0616*10^{-3}$	* $720.07*10^{-6}$	* $72.007*10^{-3}$	* 70615.5
Nm	/ 0.112985	/ 1.355822	/ $7.0616*10^{-3}$	-	* 0.101972	* 10.1972	* $10*10^6$
kp·m	/ 0.011521	/ 0.138255	/ $720.07*10^{-6}$	/ 0.101972	-	* 100	* $98.066*10^6$
kp·cm	/ 1.1521	/ 13.8255	/ $72.007*10^{-3}$	/ 10.1972	/ 100	-	* $0.9806*10^6$
dyne·cm	/ $1.129*10^6$	/ $13.558*10^6$	/ 70615.5	/ $10*10^6$	/ $98.066*10^6$	/ $0.9806*10^6$	-

9.1.7 Moment of inertia

	lb·in^2	**lb·ft^2**	**kg·m^2**	**kg·cm^2**	**kp·cm·s^2**	**oz·in^2**
lb·in^2	-	/ 144	/ 3417.16	/ 0.341716	/ 335.109	* 16
lb·ft^2	* 144	-	* 0.04214	* 421.4	* 0.429711	* 2304
kg·m^2	* 3417.16	/ 0.04214	-	* $10*10^3$	* 10.1972	* 54674
kg·cm^2	* 0.341716	/ 421.4	/ $10*10^3$	-	/ 980.665	* 5.46
kp·cm·s^2	* 335.109	/ 0.429711	/ 10.1972	* 980.665	-	* 5361.74
oz·in^2	/ 16	/ 2304	/ 54674	/ 5.46	/ 5361.74	-

9.1.8 Temperature

	°F	**°C**	**K**
°F	-	(°F - 32) * 5/9	(°F - 32) * 5/9 + 273.15
°C	°C * 9/5 + 32	-	°C + 273.15
K	(K - 273.15) * 9/5 + 32	K - 273.15	-

9.1.9 Conductor cross section

AWG	1	2	3	4	5	6	7	8	9	10	11	12	13
mm^2	42.4	33.6	26.7	21.2	16.8	13.3	10.5	8.4	6.6	5.3	4.2	3.3	2.6

AWG	14	15	16	17	18	19	20	21	22	23	24	25	26
mm^2	2.1	1.7	1.3	1.0	0.82	0.65	0.52	0.41	0.33	0.26	0.20	0.16	0.13

0198441113808, V1.02, 12.2009

9.2 Terms and Abbreviations

See chapter 2.5 "Standards and terminology" for information on the pertinent standards on which many terms are based. Some terms and abbreviations may have specific meanings with regard to the standards.

Assembly Various attributes are combined in one single data packet. Client and server know the structure of the packets. See also Explicit Message.

Attribute A single value of an object (in a network device) that can be read or written over the network. (see Class - Instance - Object - Attribute)

Big Endian format Method of storing data; the highest-value byte of a data word is at the first position (big end first).

CIP **C**ommon **I**ndustrial **P**rotocol, general specification for communication between fieldbus devices.

COS **C**hange **O**f **S**tate: special I/O connection in which data is only transmitted when changes occur.

Class DeviceNet and EtherNet/IP describes the behavior of a network node in so-called object classes. A class defines the behavior of (related) objects and consists of attributes and so-called services to work with these attributes (read/write)
for example: class vehicles, object car, attribute fuel level, service fill
(see Class - Instance - Object - Attribute)

Client First transmitter, then recipient of fieldbus messages in the client-server relationship. Starts transmission with a transmission to the server; the reference point is the server object dictionary.

EDS (**E**lectronic **D**ata **S**heet); contains the specific properties of a product.

Error Discrepancy between a computed, observed or measured value or condition and the specified or theoretically correct value or condition.

Error class Classification of errors into groups. The different error classes allow for specific responses to errors, for example by severity.

Factory setting Factory settings when the product is shipped

Fatal error In the case of fatal error, the product is no longer able to control the motor so that the power stage must be immediately disabled.

Fault Fault is a state that can be caused by an error. Further information can be found in the pertinent standards such as IEC 61800-7, ODVA Common Industrial Protocol (CIP).

Fault reset A function used to restore the drive to an operational state after a detected error is cleared by removing the cause of the error so that the error is no longer active.

Input Direction of data flow from the network perspective: data packet/status message from device, see also Output

Instance An actual object that is derived from a specific class. (see Class - Instance - Object - Attribute)

MAC ID Node address (MAC=Media Access Control); a unique address in the entire network.

Master Active bus device that controls the data traffic on the network.

ODVA **O**pen **D**eviceNet **V**endor **A**ssociation.
User organization for DeviceNet and EtherNet/IP standards.

0198441113808, V1.02, 12.2009

Object	An object is a member of a specific class. The object 'bicycle' is a member of the class 'vehicles'. The object 'car' is a member of the class 'vehicles'. (see Class - Instance - Object - Attribute)
Object dictionary	List of all parameters, values and functions available in the device. Each entry is uniquely referenced via index (16 bit) and subindex (8 bit).
Output	Direction of data flow from the network perspective: data packet/command to a device, see also Input
Parameter	Device data and values that can be read and set (to a certain extent) by the user.
Persistent	Indicates whether the value of the parameter remains in the memory after the device is switched off.
Quick Stop	Function which can be used for fast deceleration of the motor via a command or in the event of an error.
Scanner	Bus device that, as a master unit, controls all data transmission via the bus. Corresponds to the master.
User-defined unit	Unit whose reference to motor movement can be determined by the user via parameters.
Warning	If the term is used outside the context of safety instructions, a warning alerts to a potential problem that was detected by a monitoring function. A warning is not an error and does not cause a transition of the operating state.

0198441113808, V1.02, 12.2009

10 Index

10

0198441113808, V1.02, 12.2009

F

G

H

I

J

L

M

N

O

0198441113808, V1.02, 12.2009

P

Q

S

T

U

W

0198441113808, V1.02, 12.2009

0198441113808, V1.02, 12.2009