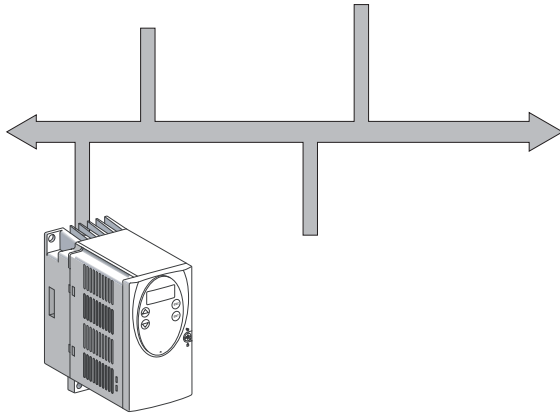


Technical Documentation



Field bus

CPD Modbus

Order no.: ACC1MDAMB00EN

Edition: V1.01, 01.2005

Berger Lahr GmbH & Co. KG
Breslauer Str. 7
D-77933 Lahr

Important information

The drive systems described here are products for general use that conform to the state of the art in technology and are designed to prevent any dangers. However, drives and drive controllers that are not specifically designed for safety functions are not approved for applications where the functioning of the drive could endanger persons. The possibility of unexpected or unbraked movements can never be totally excluded without additional safety equipment. For this reason personnel must never be in the danger zone of the drives unless additional suitable safety equipment prevents any personal danger. This applies to operation of the machine during production and also to all service and maintenance work on drives and the machine. The machine design must ensure personal safety. Suitable measures for prevention of property damage are also required.

For more information see the chapter on safety.

Not all product types are available in all countries. Please see the current catalogue for the availability of products.

We reserve the right to make technical changes.

All information refers to specifications and not to assured properties.

Most product designations are registered trademarks of their proprietors, even when not specifically noted.

Table of Contents

Important information	-2
Table of Contents	-3
Writing conventions and symbols	-5
1 Introduction	
1.1 Modbus	1-1
1.2 Directives and standards	1-1
1.3 Literature	1-1
2 Safety	
2.1 Qualification of personnel	2-1
2.2 Intended use	2-1
2.3 Hazard categories	2-2
2.4 Safety instructions	2-3
3 Installation	
4 Commissioning	
4.1 Field bus settings	4-2
4.2 Starting network operation	4-3
4.3 Running function test	4-3
4.4 Replacing units	4-4
5 Operation	
5.1 Basics	5-1
5.1.1 Modbus network	5-1
5.1.2 Modbus transmission technology	5-1
5.1.3 Modbus RTU protocol	5-3
5.2 Function codes	5-5
5.2.1 FC 3 (Read Multiple Registers)	5-5
5.2.2 FC 8 (Diagnostics)	5-6
5.2.3 FC 16 (Write Multiple Registers)	5-7
5.2.4 FC 23 (ReadWrite Multiple Registers)	5-8
5.2.5 FC 43 (Read Device Identification)	5-9
5.3 Examples of function codes FC	5-10
5.4 Examples for standardised operation modes	5-12
5.4.1 Operation mode point - to - point	5-12
5.4.2 Operation mode Profile velocity	5-13
5.4.3 Operation mode Homing	5-14

5.5	Examples of manufacturer specific operation modes .	5-15
5.5.1	Current regulation mode.....	5-15
5.5.2	Speed regulation operating mode	5-16
5.5.3	Electronic gearbox operation mode	5-17
5.5.4	Manual displacement operation mode	5-18
5.6	Connection monitoring	5-19
6	Diagnostics and troubleshooting	
6.1	Communication errors	6-1
6.2	Protocol errors	6-1
6.3	Troubleshooting	6-2
6.3.1	Synchronous errors	6-2
6.3.2	Asynchronous errors	6-2
7	Service, maintenance and disposal	
8	Glossaries	
8.1	Terms and Abbreviations	8-1
9	Index	

Writing conventions and symbols

Work steps If work steps must be carried out in sequence, they are shown as follows:

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

If a response to a work step is specified, this will inform you that the step has been carried out correctly.

Unless otherwise stated, the individual instruction steps must be carried in the given sequence.

Lists Lists can be sorted alphanumerically or by priority. Lists are structured as follows:

- Point 1
- Point 2
 - Subpoint to 2
 - Subpoint to 2
- Point 3

Making work easier Information on making work easier can be found at this symbol:



*This offers supplementary information on making work easier.
See the chapter on safety for an explanation of the safety instructions.*

1 Introduction

1.1 Modbus

This manual describes the field bus handling for products in the field bus network, which are addressed to by the modbus RTU.

In order to be able to use a PC as a master in a modbus network, it will need to be equipped with a RS485 interface card. A device for level adaptation must be connected between the network connection and the PC interface for a PC with an RS232 port.

Field bus units by other manufacturers can be operated in the same RS485 network as long as they support the modbus protocol.

1.2 Directives and standards

The following guidelines and standards apply to the field bus handling of products which are responded to by the modbus:

- RS485 standard,
- EIA RS485.2-4 serial interface

1.3 Literature

- | | |
|---------------|--|
| <i>Unit</i> | <ul style="list-style-type: none">• CPD Operating Instructions |
| <i>Modbus</i> | <ul style="list-style-type: none">• Modicon Modbus Protocol Reference Guide, PI-MBUS-300 Rev. J, June 1996, MODICON, Inc., USA• http://www.modicon.com |

2 Safety

2.1 Qualification of personnel

Only technicians who are familiar with and understand the contents of this manual and the other relevant manuals are authorised to work on and with this drive system. The technicians must be able to detect potential dangers that may be caused by setting parameters, changing parameter values and generally by the mechanical, electrical and electronic equipment.

The technicians must have sufficient technical training, knowledge and experience to recognise and avoid dangers.

The technicians must be familiar with the relevant standards, regulations and safety regulations that must be observed when working on the drive system.

2.2 Intended use

The drive systems described here are products for general use that conform to the state of the art in technology and are designed to prevent any dangers. However, drives and drive controllers that are not specifically designed for safety functions are not approved for applications where the functioning of the drive could endanger persons. The possibility of unexpected or unbraked movements can never be totally excluded without additional safety equipment. For this reason personnel must never be in the danger zone of the drives unless additional suitable safety equipment prevents any personal danger. This applies to operation of the machine during production and also to all service and maintenance work on drives and the machine. The machine design must ensure personal safety. Suitable measures for prevention of property damage are also required.

In the system configuration described the drive systems must be used in industrial applications only and must have a fixed connection only.

In all cases the applicable safety regulations and the specified operating conditions, such as environmental conditions and specified technical data, must be observed.

The drive systems may be commissioned and operated only after installation in accordance with EMC requirements and the product-specific specifications.

To prevent personal injury and damage to property damaged drive systems must not be installed or operated.

Changes and modifications of the drive systems are not permitted and if made all no warranty and liability will be accepted.

The drive system must be operated only with the specified wiring and approved accessories. In general, use only original accessories and spare parts.

The drive systems must not be operated in an environment subject to explosion hazard (ex area).

2.3 Hazard categories

Safety notes and general information are indicated by hazard messages in the manual. In addition there are symbols and instructions affixed to the product that warn of possible hazards and help to operate the product safely.

Depending on the seriousness of the hazard, the messages are divided into three hazard categories.



DANGER!

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



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.

2.4 Safety instructions



DANGER!

Electric shock, fire or explosion

- Only qualified personnel who are familiar with and understand the contents of this manual and the other relevant manuals are authorised to work on and with this drive system.
- Before working on the drive system:
 - Switch off power to all terminals.
 - Place a sign "DO NOT SWITCH ON" on the switch and lock to prevent switching on.
 - **Wait 6 minutes** (for discharge of DC bus capacitors).
 - Measure voltage between DC+ and DC- and check for <48V. (The DC bus LED is not a safe indication for absence of the DC bus voltage).
- Do not short-circuit DC bus or touch unshielded components or screws of the terminals with voltage present.
- Install all covers and close the housing doors before applying power.
- The motor generates voltage when the shaft is rotated. Lock the shaft of the motor to prevent rotation before starting work on the drive system.
- The system manufacturer is responsible for compliance with all applicable regulations relevant to earthing the drive system.
- Do not reach into the drive system (e.g. no pointed objects).



WARNING!

Danger of injury and damage to system components by loss of control!

- Observe the accident prevention regulations.
- The system manufacturer must consider the possible errors that could occur with the signals and in particular the critical functions to ensure a safe status during and after errors. Examples for these are: emergency stop, final position limit, power failure and restart.
- Consideration of possible errors must also include unexpected delay and failure of signals or functions.
- Separate redundant controller paths must be provided for dangerous functions.
- Verify the effectiveness of the measures.

**DANGER!****Danger of injury by complex system!**

When starting field bus operation the attached controllers are generally out of view of the operator and cannot be directly monitored.

- Start the system only if there are no persons within the actuation zone of the moving system components and the system can be operated safely.

**CAUTION!**

Danger of injury and damage to system components by evaluation of faulty control commands!

If a PLC is used as the master unit, the exchange of data can lead to inconsistent transmission data as a result of field bus and PLC cycles not operating synchronously.

- Please observe the notes concerning the operation using PLC.

3 Installation

**WARNING!**

Danger of injury and damage to system components by loss of control!

- Observe the accident prevention regulations.
- The system manufacturer must consider the possible errors that could occur with the signals and in particular the critical functions to ensure a safe status during and after errors. Examples for these are: emergency stop, final position limit, power failure and restart.
- Consideration of possible errors must also include unexpected delay and failure of signals or functions.
- Separate redundant controller paths must be provided for dangerous functions.
- Verify the effectiveness of the measures.

**WARNING!**

Interference with signals and devices may cause injury

Distorted signals can cause unexpected device responses.

- Install the wiring in accordance with the EMC requirements.
- Check compliance with the EMC requirements, particularly in an environment subject to strong interference.

The installation should be carried out in accordance with the details given in the unit manual under the chapter on installation, with special emphasis on the EMC requirements.

4 Commissioning



DANGER!

Danger of injury by complex system!

When starting field bus operation the attached controllers are generally out of view of the operator and cannot be directly monitored.

- Start the system only if there are no persons within the actuation zone of the moving system components and the system can be operated safely.



WARNING!

Danger of personal injury and damage to system parts by uncontrolled system operation!

- Do not write to reserved parameters.
- Do not write to parameters before you have understood the function. For more information see the controller manuals.
- Run the first tests without coupled loads.
- Make sure that the system is free and ready for the movement before changing parameters.
- Check the use of the bits during field bus communication: Bit 0 is far right (least significant). Bit 15 is far left (most significant).
- Check the use of the word sequence during field bus communication:
- Do not establish a field bus connection before you have understood the communications principles.

4.1 Field bus settings

Transmission format The data transmission format is set in the factory to:

- Modbus RTU
- 19200 baud
- 8 data bits (LSB is transmitted first)
- Even parity
- 1 stop bit

The communication between master and slave takes place in semi-duplex mode

Node address Up to 31 field bus participants can be connected to the bus. Each field bus participant must have its own node address, which can be assigned only once in the network. The node address is set in the factory to 1. The node address 0 is a broadcast address, which all field bus participants in the network receive data from, but do not answer.

Parameter Name Code HMI menu, Code	Meaning	Unit Minimum value Default value Maximum value	Data type R/W per Expert	Parameter address via field bus
MBadr	Modbus address ()		UINT16	
MBAD	valid addresses : 1 to 247	1	R/W	
COM- <i>nAdr</i>		1 247	per. -	

Baud rate The baud rate must be set to the same value for all field bus participants.

Parameter Name Code HMI menu, Code	Meaning	Unit Minimum value Default value Maximum value	Data type R/W per Expert	Parameter address via field bus
MBbaud	Modbus baud rate()	Baud	UINT16	
MBBD	Allowed baud rates:	0	R/W	
COM- <i>nbbd</i>	9600 19200 38400	19200 38400	per. -	

Data bits;stop bits and parity The following combinations of data bits, stop bits and parity can be set.

Parameter Name Code HMI menu, Code	Meaning	Unit Minimum value Default value Maximum value	Data type R/W per Expert	Parameter address via field bus
MBformat	Modbus data format()		UINT16	
MBFO	1 / 8Bit NoParity 1Stop / 8n1: 8 bit, no parity bit, 1 stop bit	1	R/W	
COM- <i>nbfO</i>	2 / 8Bit EvenParity 1Stop / 8e1: 8 bit, even parity bit, 1 stop bit (default)	2	per.	
	3 / 8Bit OddParity 1Stop / 8o1: 8 bit, odd parity bit, 1 stop bit	4	-	
	4 / 8Bit NoParity 2Stop / 8n2: 8 bit, no parity bit, 2 stop bits			

Node guarding A monitoring time can be set for the node guarding.

Parameter Name Code HMI menu, Code	Meaning	Unit Minimum value Default value Maximum value	Data type R/W per Expert	Parameter address via field bus
MBnode_guard	Modbus Node Guard() Connection monitoring 0 : inactive (default) >0 : Monitoring time	ms 0 0 10000	UINT16 R/W - -	

Word sequence This setting specifies how the parameter data (2 words) are sent.

Example: parameter value = 1234 5678_h

- HighWord-LowWord = 1234_h , 5678_h
- LowWord-HighWord = 5678_h , 1234_h

Parameter Name Code HMI menu, Code	Meaning	Unit Minimum value Default value Maximum value	Data type R/W per Expert	Parameter address via field bus
MBdword_order	Modbus word sequence for double words (32 bit values)	0	UINT16 R/W	
MBWO	(Register Swap)	0	per.	
COM- HighLow	0 / HighLow / HiLo : HighWord-LowWord, big endian -> Modicon Quantum (default) 1 / LowHigh / LoHi : LowWord-HighWord little endian -> Premium, HMI (Telemecanique)	1	-	

Detailed description A detailed description of the settings is given in the "Installation" chapter of the unit documentation.

4.2 Starting network operation

Network operation can be started from a master unit. This can be a PLC or a PC which uses the corresponding user software to send field bus commands and to read reception data.

4.3 Running function test

If the slave does not send a response, check the following settings:

- Voltage supply switched on and master for network operation started?
- Cable connections mechanically sound?
- Correct address set?
- Same settings for baud rate and interface parameters (data bits, parity, stop bits)?

Information concerning fault finding and trouble shooting can be found in the chapter 6 "Diagnostics and troubleshooting" or in the unit manual.

4.4 Replacing units

The unit behaviour should remain the same after a slave unit has been replaced. This means that the same parameters must be set on the new unit as were set on the old unit.

The field bus parameters should be set via the HMI or by using the installation software, otherwise communication with the unit in the field bus network will not be possible.

If the new unit has already been configured for field bus operation, then it will recognise the values of the field bus parameters when starting up.

If other factory set parameters are to be modified, these values can be stored in the master control system. They will be transferred after each start-up of the unit e.g. in the "ReadyToSwitchOn" (*rdy*) state.

5 Operation

5.1 Basics

5.1.1 Modbus network

A modbus network consists of one superordinate master and at least one slave unit.

Master Masters are active field bus participants which control the data traffic within a network. Examples of masters are:

- automation devices, e.g. PLCs
- PCs

Slave Slaves are passive field bus participants. They receive control commands and supply data to the master. Examples of slaves are programmable drive controls, such as, for example, the unit here in consideration.

A typical application for Modbus is communications between devices in automated manufacturing.

Slave address In order for the master to be able to communicate with a slave in the field bus, it must address the slave. The node address of the slave is contained within the data frame. Notes on setting the address in the units can be found on page 4-2.

Parameter addresses The modbus parameter addresses (WR or RD par.) are the basis for the communication between the units and the field bus master. A collection of the valid addresses for the unit together with the corresponding unit parameters can be taken from the chapter "Parameters" in the unit documentation.

5.1.2 Modbus transmission technology

Data transmission in the modbus system takes place via a serial interface (RS485)

Data exchange between the field bus participants takes place in the master-slave process. Only the master can send commands (queries). The master device can address every slave device individually. The reaction (response) of a slave device is to send the requested data or to confirm execution of the requested operating function, depending on the command.

During data transmission request and response alternate continuously.

The master sends commands to the slave device. The slave device only sends data when requested by the master device.

The exchange of data follows a fixed routine. The process is always viewed from the point of view of the master device.

The commands in the form of function codes are embedded in the transmitted data frame.

The request contains a function code which represents a command to be executed for the slave unit. The information required to execute the command is included in the transmitted data bytes.

The error-checking bytes enable the slave device to check the integrity of the received data.

The response from the slave unit contains the function code of the request as an "echo". The data bytes of the response are dependent upon the function code being used and are provided by the slave unit. The error-checking bytes enable the master to check the validity of the received data.

The structure of the data sent is determined by the modbus protocol.

Modbus protocols

In general, there are 3 variants of the modbus protocol:

- Modbus RTU Master-slave-communication, binary coded
- Modbus ASCII: Master-slave-communication, ASCII coded
- Modbus PLUS: Peer-to-peer communication

The unit only supports the Modbus RTU protocol.

5.1.3 Modbus RTU protocol

5.1.3.1 Modbus RTU message

A modbus message is also designated as a data frame or telegram. If the message is addressed to a slave unit then it is called sending data frame or request. In reply to this request, this unit sends a response, the reception data frame.

A Modbus RTU data frame consists of the following fields:

<SlaveAddr> <FC> <Data> <CRC>

Field name	Meaning	Number of bytes
<SlaveAddr>	Slave address	1
<FC>	Function code	1
<Data>	Data	n (High-Byte, Low-Byte)
<CRC>	Checksum	2 (Low-Byte, High-Byte)

Table 5.1 Fields of a Modbus RTU message

The beginning and the end of a data frame are detected by a time condition. An interruption of 3.5 characters means that the data frame is completed and that the next character is to be interpreted as a slave address. As a result, a data frame must be sent as a continuous data stream. In the event of an interruption of more than 1.5 characters the data will be rejected by the receiver.

5.1.3.2 Request and response in the Modbus RTU

Request and response have an analogue structure.

If an error occurs on receipt of the request or the slave device cannot execute the action, the slave device sends an error message as the response.

5.1.3.3 Fields of a Modbus RTU data frame

- <SlaveAddr> The device address identifies the target device. It is the same in the query and the response.
- <FC> The function code determines which modbus service the slave unit should carry out. The function code is the same in both the request and in the response.
- <Data> Whether a data field is contained in the data frame and its length, is dependent upon the function code being used. The relevant control and action commands for the specific function code are in the data field of a request. The data field of the response contains the data which was requested by the master, depending upon the function code. It can also include an error message.
- <CRC> To allow error checking under Modbus RTU the "Cyclic Redundance Checksum" (CRC) is formed from the transmitted fields <SlaveAddr>, <FC> and <Data>. It comprises a CRC16 with a generator polynome $A001_h$, which is calculated from the algorithm shown in the following illustration.

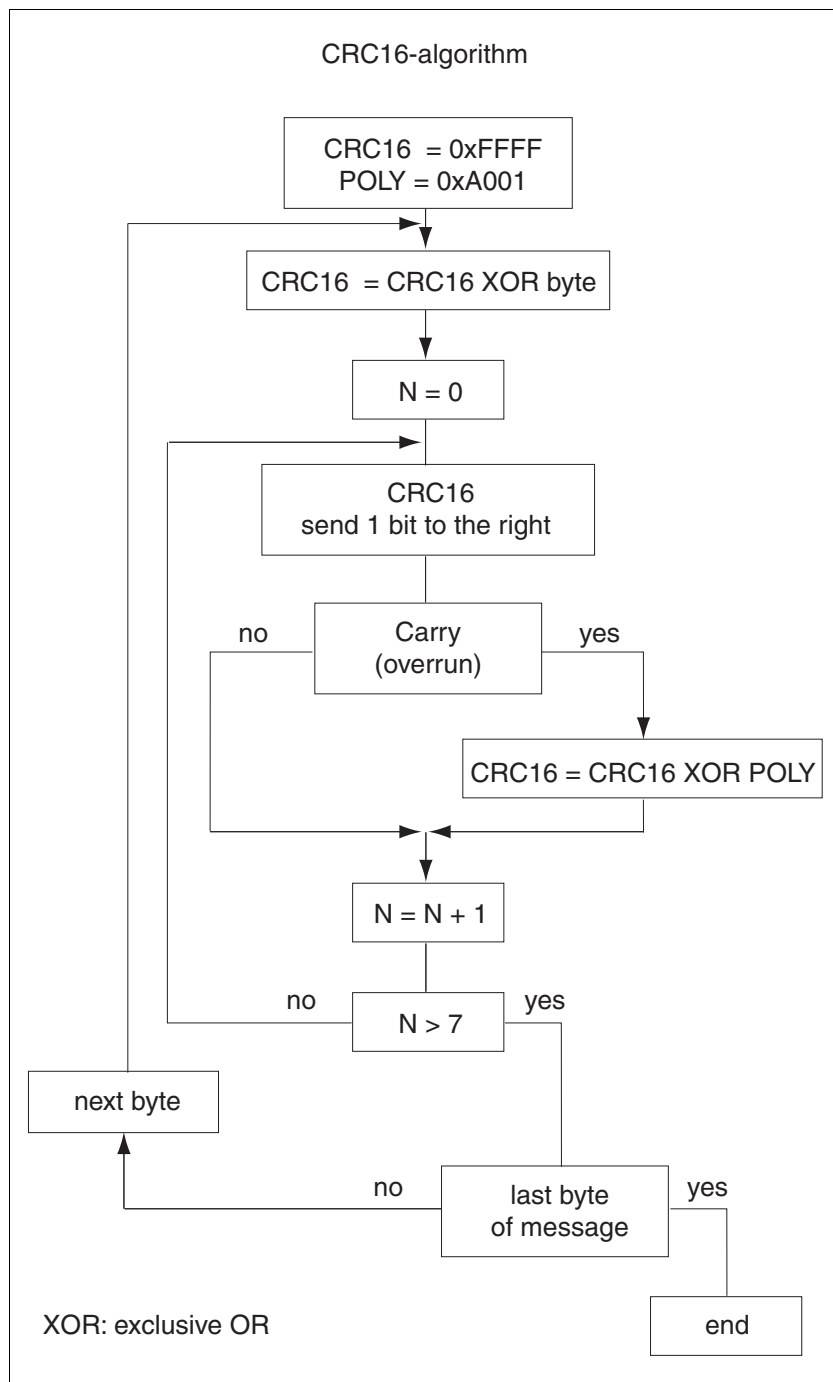


Figure 5.1 CRC16 algorithm

5.2 Function codes

The function codes (function codes, FC) permit the initiation of various communication mechanisms (services) which are provided by the modbus protocol. The following table provides an overview of the function codes implemented in the unit.

FC	Modbus meaning	Unit meaning
3	Read Multiple Registers	Read n Parameter
8	Diagnostics	Diagnostics
16	Write Multiple Registers	Write n Parameter
23	Read/Write Multiple Registers	Read Write n Parameter
43 Subcode14	Read Schneider Identification	–

5.2.1 FC 3 (Read Multiple Registers)

With this function code, it is possible to read n sequential parameters from any address from the unit.

Structure of request: <FC> <1st ReadAddr> <NumRegister>

Field	Bytes	Value	Meaning
FC	1	3 = 03 _h	Request code
1st ReadingAddress	2	(various)	Address of the first parameter to be read
NumRegister	2	2 * n	Number of 16-bit registers to be read.

Structure of the positive response <FC> <NumBytes> <Data>

Field	Bytes	Value	Meaning
FC	1	3 = 03 _h	Response code
NumBytes	1	4 * n	Number of data bytes
Data	4n	(various)	n read parameter values

Structure of the negative response see chapter 6.2 "Protocol errors"

Example for Premium in PL7 ► Read the actual position of the unit.

The modbus register address for the actual position (PACT) is 16#1F22. The following command calls the service "Read Multiple Registers" (FC 3). The current position is saved in %MW20, the result of the execution of the function from %MW30.

```
// Request 2 words from the register address 16#1F22
from the unit with the address (ADR)READ_VAR (ADR,
'%MW', 16#1F22, 2, %MW20:2, %MW30:4);
```



All parameters are sent as 4 byte wide values (32Bit). A list of all the parameters can be found in the unit manual under the chapter "Parameters".

5.2.2 FC 8 (Diagnostics)

This function code allows to read diagnostics data from a unit.

Structure of request: <FC> <Subfunction> <Data>

Field	Bytes	Value	Meaning
FC	1	8 = 08 _h	Request code
Subfunction	2	(various)	Subfunction (see Table 5.2)
Data	2	(various)	Data (depending on subfunction)

Structure of the positive response: <FC> <Subfunction> <Data>

Field	Bytes	Value	Meaning
FC	1	8 = 08 _h	Response code
Subfunction	2	(various)	Subfunction (see)Table 5.2)
Data	2	(various)	Requested diagnostics data

Structure of the negative response see chapter 6.2 "Protocol errors"

Subfunctions The following subfunctions are provided by the modbus protocol:

Code	Subfunction	Unit-specific function
00	Return Query Data	Return request as response
01	Restart Communication Option	Re-initialise the communication port
02	Return Diagnostic Register	Provide the error number in the event of synchronous errors
03	(reserved)	–
04	Force Listen Only Mode	Switch the slave unit to mute
05..09	(reserved)	–
10	Clear Counters and Diagnostic Register	Reset all statistical counters
11	Return Bus Message Count	Show the number of received messages
12	Return Bus Communication Error Count	Show number of detected LRC errors
13	Return Bus Exception Error Count	Show number of detected exception errors
14	(reserved)	–
15	(reserved)	–
16	Return Slave NAK Count	Show the number of detected "not-acknowledged" errors
17	Return Slave Busy Count	Show the number of "Slave Busy" errors
18	Return Bus Char Overrun Count	Show number of detected character overrun errors
> 18	(reserved)	–

Table 5.2 Modbus subfunctions for FC 8

5.2.3 FC 16 (Write Multiple Registers)

This function code allows m sequential parameters from any address to be written on the unit.

Structure of request: <FC> <1st WriteAddress> <NumRegister> <NumBytes> <Data>

Field	Bytes	Value	Meaning
FC	1	16 = 10 _h	Request code
1st WriteAddress	2	(various)	Address of the first parameter to be written
NumRegister	2	2 * m	Number of registers to be written
NumBytes	1	4m	Number of data bytes
Data	2m	(various)	m parameter values to be written

Structure of the positive response: <FC> <ParamAddress> <NumRegister>

Field	Bytes	Value	Meaning
FC	1	16 = 10 _h	Response code
ParamAddress	2	(various)	inverted from request
NumRegister	2	2	inverted from request

Structure of the negative response see chapter 6.2 "Protocol errors"

Example for Premium in PL7 ► Write a target position into the unit (point - to - point operation).

The modbus register address for the target position `TargetPosition` is 6942dec = 16#1B1E. The following command calls the modbus service "Write Multiple Registers" (FC 16). The target position is in %MW25, the result of the execution of the function is saved from %MW35.

```
// Write 2 words into the register address 16#1B1E of
the unit with the address (ADR)WRITE_VAR (ADR, '%MW',
16#1B1E, 2, %MW25:2, %MW35:4);
```



All parameters are sent as 4 byte wide values (32Bit). A list of all the parameters can be found in the unit manual under the chapter "Parameters".

5.2.4 FC 23 (ReadWrite Multiple Registers)

This function code allows data to be exchanged between master and slave unit in reading and writing modes.

Structure of request: <FC> <1st ReadAddress> <NumRegister> <1st WriteAddress> <NumRegister> <NumBytes> <Data>

Field	Bytes	Value	Meaning
FC	1	23 = 17 _h	Function code
1st ReadingAddress	2	(various)	Address of the first parameter to be read
NumRegister	2	2n	Number of 16-bit registers to be read.
1st WriteAddress	2	(various)	Address of the first parameter to be written
NumRegister	2	2m	Number of registers to be written
NumBytes	1	4m	Number of data bytes
Data	4m	(various)	m parameter values to be written

Structure of the positive response: <FC> <NumBytes> <Data>

Field	Bytes	Value	Meaning
FC	1	23 = 17 _h	Response code
NumBytes	1	2n	Number of data bytes
Data	2n	n sequential	n read parameter values

Structure of the negative response see chapter 6.2 "Protocol errors"



All parameters are sent as 4 byte wide values (32Bit). A list of all the parameters can be found in the unit manual under the chapter "Parameters".

5.2.5 FC 43 (Read Device Identification)

This function code allows data for unit identification to be read out.

Structure of request: <FC> <MEI> <ReadDevID> <ObjID>

Field	Bytes	Value	Meaning
FC	1	43 = 2B _h	Function code
MEI	1	14 = 0E _h	Modbus Encapsulated Interface Type(subfunction)
ReadDevID	1	01	Read Device ID Codeall objects
ObjID	1	0x00	Object ID vendor name, product code, revision

Structure of the positive response: <FC> <MEI> <ReadDevID> <ConfLev><MoreFoll><NextObjID><NumbObj><Data>

Field	Bytes	Value	Meaning
FC	1	43 = 2B _h	Function code
MEI	1	14 = 0E _h	Modbus Encapsulated Interface Type(subfunction)
ReadDevID	1	01	Read Device ID Codeall objects
ConfLev	1	02	Conformity Level fixed value
MoreFoll	1	00	More Follows fixed value, since telegram length < 255
NextObjID	1	00	Next Object ID fixed value because of MoreFoll = 00
NumbObj	1	03	Number of objects
Data	1	(various)	Object ID (1 byte, see following table) Object Length (1 byte) Object data (various)

The following identification data can be read:

Object ID	Object Name	Value
0x00	vendor name	e.g. "Telemecanique"
0x01	product code	„XXXXxxXxxXx“ (e.g.: „DCX1702F10S1“)
0x03	revision	"Vxx.yyy" (e.g.: "V02.001")

Structure of the negative response see chapter 6.2 "Protocol errors"

5.3 Examples of function codes FC

In general: parameters are always read or written individually.

Exception: If modbus addresses (or modbus register addresses) are located sequentially, (modbus address, modbus address+2), it is sufficient to have one request for the transmission of the values.

Example 1 Read an error register entry -> ErrNum (15362) / ErrClass (15364) / ErrTime (15366) / ErrQual (15368). Since all the error information have modbus addresses in ascending sequence, one reading request as follows is sufficient:

Field	Bytes	Value	Meaning
FC (Request Code)	1	3	Request code (Multiple Register READ)
ParamAddress	2	15362dec (3C02hex)	First register address to be read
NumRegister	2	4 * 2 = 8	Number of 16-bit registers to be read = 8, i. e. read 16 byte data

Table 5.3 Example 1, FC3 Request

Field	Bytes	Value	Meaning
FC (Request Code)	1	3	Request code Multiple Register READ
NumBytes	1	16	Number of bytes: 8 Byte Data
Data	16	32Bit value 32Bit value 32Bit value 32Bit value	ErrNumc, 15362 (error number) ErrClass, 15364 (error class) ErrTime, 15366 (error time) ErrQual, 15368 (error designation)

Table 5.4 Example 1, FC3 Positive Response

Example 2 Writing of the software limit switches -> SoftwareLimPos (1544) / SoftwareLimNeg(1546).

Since these parameters also are located one after the other (modbus address, modbus address+2), a write command can be used:

Field	Bytes	Value	Meaning
FC (Request Code)	1	16	Request code (Multiple Register WRITE)
ParamAddress	2	1544dec (608hex)	First register address to be written
NumRegister	2	2 * 2 = 4	Number of registers = 4 (8 bytes data)
NumBytes	1	8	Number of bytes: 8 bytes data
Data	8	32Bit value 32Bit value	SoftwareLimPos, 1544 SoftwareLimNeg, 1546

Table 5.5 Example 2, FC16 Request

Field	Bytes	Value	Meaning
FC (Request Code)	1	16	Response code (Multiple Register WRITE)
ParamAddress	2	1544dec (608hex)	Modbus register address
NumRegister	2	2 * 2 = 4	Number of registers = 4 (8 bytes data)

Table 5.6 Example 2, FC16 Positive Response

5.4 Examples for standardised operation modes

5.4.1 Operation mode point - to - point

Example Node address 1.

Description Field bus command / parameter name (address)	Value
► Acceleration ramp 2000 rpm*s FC16 / RAMPacc (1556)	0000 07D0h
► Delay ramp 4000 rpm*s FC16 / RAMPdecel (1558)	0000 0FA0h
► Limit set speed 6000 rpm FC16 / RAMPn_max (1554)	0000 1770h
► Set speed 4000 rpm FC16 / PPn_target (6942)	0000 0FA0h
► Disable Voltage FC16 / DCOMcontrol (6914)	0000 0000h
► Shut Down FC16 / DCOMcontrol (6914)	0000 0006h
► Operation Enable FC16 / DCOMcontrol (6914)	0000 000Fh
► Check operation state. ¹⁾ FC 3 / DCOMstatus (6916)	
◁ Operation state active	0000 0007h
► Start operation mode FC16 / DCOMopmode (6918)	0000 0001h
► Check operation mode ¹⁾ FC 3 / _DCOMopmode_act (6920)	
◁ Operating mode active	0000 0001h
► Store new set position FC16 / PPP_targetusr (6940)	0000 0030h
► Start absolute positioning FC16 / DCOMcontrol (6914)	0000 005Fh
► Check target position ¹⁾ FC 3 / DCOMstatus (6916)	
◁ Target position reached (Bit 10 = 1)	xxxx x4xxh
► Reset start bit FC16 / DCOMcontrol (6914)	0000 000Fh

¹⁾ Must be checked cyclically.

5.4.2 Operation mode Profile velocity

Example Node address 1.

Description Field bus command / parameter name (address)	Value
► Acceleration ramp 2000 rpm*s FC16 / RAMPacc (1556)	0000 07D0 _h
► Delay ramp 10000 rpm*s FC16 / RAMPdecel (1558)	0000 2710 _h
► Limit set speed 10000 rpm FC16 / RAMPn_max (1554)	0000 2710 _h
► Disable Voltage FC16 / DCOMcontrol (6914)	0000 0000 _h
► Shut Down FC16 / DCOMcontrol (6914)	0000 0006 _h
► Operation Enable FC16 / DCOMcontrol (6914)	0000 000F _h
► Check operation state. ¹⁾ FC 3 / DCOMstatus (6916)	
◁ Operation state active	0000 0007 _h
► Start operation mode FC16 / DCOMopmode (6918)	0000 0003 _h
► Check operation mode ¹⁾ FC 3 / _DCOMopmode_act (6920)	
◁ Operating mode active	0000 0003 _h
► Transmission set speed 1000 rpm FC16 / PVn_target (6938)	0000 03E8 _h
► Check target position ¹⁾ FC 3 / DCOMstatus (6916)	
◁ Target position reached (Bit 10 = 1)	xxxx x4xx _h
► Transmission set speed 0 rpm FC16 / PVn_target (6938)	0000 0000 _h
► Check target position ¹⁾ FC 3 / DCOMstatus (6916)	
◁ Target position reached (Bit 10 = 1)	xxxx x4xx _h

1) Must be checked cyclically.

5.4.3 Operation mode Homing

Example Node address 1.

Description Field bus command / parameter name (address)	Value
▶ Set speed for displacement to limit switch 100 rpm FC16 / HMn (10248)	0000 0064 _h
▶ Set speed for free displacement 10 rpm FC16 / HMn_out (10250)	0000 000Ah
▶ Disable Voltage FC16 / DCOMcontrol (6914)	0000 0000 _h
▶ Shut Down FC16 / DCOMcontrol (6914)	0000 0006 _h
▶ Operation Enable FC16 / DCOMcontrol (6914)	0000 000F _h
▶ Check operation state. ¹⁾ FC 3 / DCOMstatus (6916)	
◁ Operation state active	0000 0007 _h
▶ Start operation mode FC16 / DCOMopmode (6918)	0000 0006 _h
▶ Check operation mode ¹⁾ FC 3 / _DCOMopmode_act (6920)	
◁ Operating mode active	0000 0006 _h
▶ Select reference displacement method, LimN (17) FC16 / HMmethod (6936)	0000 0011 _h
▶ Start referencing FC16 / DCOMcontrol (6914)	0000 001F _h
▶ Check referencing ¹⁾ FC 3 / DCOMstatus (6916)	
◁ Drive has a valid reference point (Bit 12 = 1)	xxxx 1xxx _h
▶ Reset start bit FC16 / DCOMcontrol (6914)	0000 000F _h

1) Must be checked cyclically.

5.5 Examples of manufacturer specific operation modes

5.5.1 Current regulation mode.

Example Node address 1.

Description	
Field bus command / parameter name (address)	Value
▶ Disable Voltage FC16 / DCOMcontrol (6914)	0000 0000 _h
▶ Shut Down FC16 / DCOMcontrol (6914)	0000 0006 _h
▶ Operation Enable FC16 / DCOMcontrol (6914)	0000 000F _h
▶ Check operation state. ¹⁾ FC 3 / DCOMstatus (6916)	
◁ Operation state active	0000 0007 _h
▶ Start operation mode (-3) FC16 / DCOMopmode (6918)	FFFF FFFD _h
▶ Check operation mode ¹⁾ FC 3 / _DCOMopmode_act (6920)	
◁ Operating mode active	FFFF FFFD _h
▶ Default set value via parameter FC16 / CURreference (6944)	0000 0002 _h
▶ Transfer set current 1000 (10A) FC16 / CUR_I_target (8200)	0000 03E8 _h

1) Must be checked cyclically.

5.5.2 Speed regulation operating mode

Example Node address 1.

Description	
Field bus command / parameter name (address)	Value
▶ Disable Voltage FC16 / DCOMcontrol (6914)	0000 0000 _h
▶ Shut Down FC16 / DCOMcontrol (6914)	0000 0006 _h
▶ Operation Enable FC16 / DCOMcontrol (6914)	0000 000F _h
▶ Check operation state. ¹⁾ FC 3 / DCOMstatus (6916)	
◁ Operation state active	0000 0007 _h
▶ Start operation mode (-4) FC16 / DCOMopmode (6918)	FFFF FFFC _h
▶ Check operation mode ¹⁾ FC 3 / _DCOMopmode_act (6920)	
◁ Operating mode active	FFFF FFFC _h
▶ Default set value via parameter FC16 / SPEEDreference (6946)	0000 0002 _h
▶ Transmission set speed 1000 rpm FC16 / SPEEDn_target (8456)	0000 03E8 _h

1) Must be checked cyclically.

5.5.3 Electronic gearbox operation mode

Example Node address 1.

Description	
Field bus command / parameter name (address)	Value
▶ Signal interchange position interface FC16 / IOposInterfac (1284)	0000 0001 _h
▶ Disable Voltage FC16 / DCOMcontrol (6914)	0000 0000 _h
▶ Shut Down FC16 / DCOMcontrol (6914)	0000 0006 _h
▶ Operation Enable FC16 / DCOMcontrol (6914)	0000 000F _h
▶ Check operation state. ¹⁾ FC 3 / DCOMstatus (6916)	
◁ Operation state active	0000 0007 _h
▶ Start operation mode (-2) FC16 / DCOMopmode (6918)	FFFF FFFE _h
▶ Check operation mode ¹⁾ FC 3 / _DCOMopmode_act (6920)	
◁ Operating mode active	FFFF FFFE _h
▶ Gearbox activation with immediate synchronisation FC16 / GEARreference (8948)	0000 0001 _h
▶ Transfer denominator FC16 / GEARdenom (9734)	0000 0003 _h
▶ Transfer numerator FC16 / GEARnum (9736)	0000 0002 _h

1) Must be checked cyclically.

5.5.4 Manual displacement operation mode

Example Node address 1.

Description	
Field bus command / parameter name (address)	Value
▶ Speed slow displacement to 100 rpm FC16 / JOGn_slow (10504)	0000 0064 _h
▶ Speed fast displacement to 250 rpm FC16 / JOGn_fast (10506)	0000 00FA _h
▶ Disable Voltage FC16 / DCOMcontrol (6914)	0000 0000 _h
▶ Shut Down FC16 / DCOMcontrol (6914)	0000 0006 _h
▶ Operation Enable FC16 / DCOMcontrol (6914)	0000 000F _h
▶ Check operation state. ¹⁾ FC 3 / DCOMstatus (6916)	
◁ Operation state active	0000 0007 _h
▶ Start operation mode (-1) FC16 / DCOMopmode (6918)	FFFF FFFF _h
▶ Check operation mode ¹⁾ FC 3 / _DCOMopmode_act (6920)	
◁ Operating mode active	FFFF FFFF _h
▶ Manual displacement (pos. sense of rotation, slow) FC16 / JOGactivate (6930)	0000 0001 _h
▶ Manual displacement (pos. sense of rotation, fast) FC16 / JOGactivate (6930)	0000 0005 _h

1) Must be checked cyclically.

5.6 Connection monitoring



WARNING!

Danger of injury and damage to system components by loss of control!

- Enable the timeout function. Without timeout the system will not detect interruptions in the communication connection.
- The shorter the timeout period the faster the detection of the interruption.

Two types of monitoring are continuously in operation on the master and slave side to monitor error-free data exchange:

- Timeout monitoring
- Received character check

Timeout monitoring

A timeout can be set on the slave side, specifying the time within which the master must re-contact (node guarding). If no contact is received from the master within this time period, the unit will stop the process initiated by the field bus and will initiate a "quick stop".

The time interval for the timeout message can be set with the parameter MbNodeG (5644). Value range:

=0: Node guarding switched off (factory setting) >0: Node guarding in ms (milliseconds)

If the timeout monitoring is not used, then an interruption of the communication does not produce an error message. The control system continues to perform the last transmitted command.

Use the timeout function during normal network operation. The timeout monitoring can be switched off during troubleshooting. The unit manufacturer must then provide alternative control paths for stopping and controlling the motor.

Received character check

The unit checks a field bus command for transmission errors. If the transmission data could not be correctly received, e.g. because of a parity error, an acknowledgement is not generated and the master detects a timeout.

6 Diagnostics and troubleshooting

In troubleshooting, there are two error types to be distinguished:

- Communication errors (serial transmission errors),
- Protocol errors (specific to Modbus),

6.1 Communication errors

Communication errors include:

- character timeout (time exceeded during transmission of characters),
- parity errors,
- framing errors (error in data frame),
- overrun errors (overrun in receive register of the serial module).

If one of these errors arises, the unit no longer responds. The master generates a timeout error.

6.2 Protocol errors

With all protocol errors the response is sent as an exception code. The response has the same function code as the normal response, but, in addition, the "MSB" is set. The function code is followed by a 1-byte wide exception code.

Structure of the negative response For FC3, FC8, FC16, FC23: <FC> <ExcpCode>

For FC43: <FC> <MEI> <ExcpCode>

Field	Bytes	Value	Meaning
FC	1	FC + 128 (80 _h) 03 _h + 80 _h = 83 _h 08 _h + 80 _h = 88 _h 10 _h + 80 _h = 90 _h 17 _h + 80 _h = 97 _h 2B _h + 80 _h = AB _h	Response code for errors FC3 FC8 FC16 FC23 FC43
MEI (only FC43)	1	14	Modbus Encapsulated Interface Type(subfunction)
ExcpCode	1	01 _h .. 04 _h	01 _h = Invalid function 02 _h = Invalid data addresses 03 _h = Invalid data 04 _h = Slave unit error

6.3 Troubleshooting

6.3.1 Synchronous errors

Synchronous errors only occur in response to a command. When a command is sent, it is immediately checked for whether it can be correctly executed. If this is not the case, the unit sends an exception code as a return message to the command. The actual error which has occurred can be read out using the diagnostics function, see page 5-6.

Causes of error

Possible causes of a synchronous error are:

- Unknown command, syntax error or incorrect transmitting data frame
- Parameter value outside the permissible value range
- Non-permissible action or control command during a running process.
- Error while executing an action or control command.

The table with the error numbers is in the manual in the chapter on diagnostics and troubleshooting.

6.3.2 Asynchronous errors

Asynchronous errors

Asynchronous errors are reported by the monitoring devices of the unit as soon as a unit error occurs. An asynchronous error is signalled via bit 3, "Fault" of the parameter `DCOMstatus` (6041_h). For errors that cause a displacement interruption the unit sends an EMCY message.

7 Service, maintenance and disposal

Information concerning service, maintenance, and disposal can be obtained from the corresponding unit manual.

8 Glossaries

8.1 Terms and Abbreviations

<i>AC</i>	Alternating Current
<i>Actual position</i>	Current absolute or relative position of moving components in the drive system.
<i>Address</i>	Memory location which can be accessed by its unique number. See also Slave address.
<i>API</i>	Application Program Interface
<i>ASCII</i>	American Standard Code for Information Interchange; Standard for coding text characters
<i>Asynchronous error</i>	Error detected and reported by the internal controller monitoring equipment.
<i>CRC</i>	Cyclical Redundancy Check, error checking
<i>Data frame</i>	Serially transmitted data packet with unique start and end identification. The structure depends on the protocol in use.
<i>DC</i>	Direct current
<i>Default values</i>	Factory settings.
<i>Direction of rotation</i>	Rotation of the motor shaft in a positive or negative direction of rotation. A positive direction of rotation is defined as the motor shaft rotating clockwise as the observer faces the end of the protruding shaft.
<i>Drive system</i>	The drive system consists of the controller, power amplifier and motor.
<i>Electronic gear</i>	An input speed is recalculated by the drive system using the values of an adjustable gear factor to derive a new output speed for the motor movement.
<i>EMC</i>	Electromagnetic compatibility
<i>Error class</i>	Classification of possible operating faults of the drive system that result in an error status.
<i>Field</i>	Bytes of a message that belong together because of their content.
<i>Fieldbus</i>	A bus optimised for data transmission between field devices. A Fieldbus is "open", meaning that it is not proprietary (not supported by only one manufacturer). The parameter settings of the drive system can be called and modified via the Fieldbus, inputs can be monitored and outputs controlled and diagnosis and error monitoring functions enabled.
<i>Half duplex</i>	Bidirectional data transmission in which only one user can transmit at any time.
<i>High/open</i>	Signal status of an input or output signal; in the idle state the signal voltage is high, high level.
<i>HMI</i>	Human Machine Interface, handheld operating unit.
<i>Pt-monitoring</i>	Predictive temperature monitoring. The expected temperature rise of unit components is calculated in advance on the basis of the motor current. If a limit value is exceeded, the drive system reduces the motor current.

<i>I/O</i>	Inputs/Outputs
<i>Inc</i>	Increment
<i>Incremental signals</i>	Angular steps of an encoder in the form of square-wave pulse sequences. Relative changes in position are signalled by the number of pulses contained in the pulse sequence.
<i>Index pulse</i>	Encoder signal for referencing the rotor position in the motor. The encoder sends one index pulse per revolution.
<i>LED</i>	Light-Emitting Diode
<i>Limit switch</i>	Switch that signals an overrun of the permissible travel range.
<i>Low/open</i>	Signal status of an input or output signal; in the idle state signal voltage is low (low level).
<i>LRC</i>	Longitudinal Redundancy Check, error checking
<i>LSB</i>	Least Significant Bit, the least significant bit of a bit sequence, e. g. of a byte
<i>M</i>	Motor
<i>Master</i>	Active bus user that controls the data traffic in the network.
<i>Node-Guarding</i>	Monitoring function with slave at an interface for cyclic communication.
<i>Parameter</i>	Device functions and values that can be set and called by the user.
<i>Power amplifier</i>	A device that generates current for controlling the motor in accordance with the positioning signals from the controller.
<i>PC</i>	Personal Computer
<i>PLC</i>	Programmable Logic Controller
<i>Profibus</i>	Standardised open field bus compliant with EN 50254-2 over which drives and other devices from different manufacturers communicate with one another.
<i>Protocol</i>	Guideline that specifies the format required for transmitting data.
<i>PWM</i>	Pulse Width Modulation
<i>Quick Stop</i>	Quick stop, function used to provide quick braking of the motor via a command or in the event of a fault.
<i>Register</i>	Memory range of specified size (generally 8, 16 or 32 bits) for temporary storage of data transmitted from one system unit to another.
<i>RS485</i>	Field bus interface compliant with EIA-485, which enables serial data transmission with multiple devices.
<i>RTU</i>	Remote Terminal Unit
<i>Slave</i>	Passive bus user that receives control commands and sends data to the master.
<i>Slave address</i>	Direct communication between master and slave devices is only possible after assignment of addresses.
<i>Synchronous errors</i>	Error reported by the controller if it is unable to execute a command sent by the master.
<i>Timeout</i>	Error caused by exceeding the maximum allowable time between query and response of devices.

Watchdog Equipment that monitors cyclic basic functions in the drive system. Power amplifier and outputs are switched off in the event of error.

9 Index

A

Abbreviations 8-1
Asynchronous errors 6-2

B

Baud rate 4-2

C

Character check 5-19
Checksum 5-3
Commissioning 4-1
Communication errors 6-1
Connection monitoring 5-19
Current regulation
 example 5-15

D

Danger classes 2-2
Data bits 4-2
Data frame 5-3
Diagnostics 5-6, 6-1
Directives 1-1
Disposal 7-1

E

Electronic gearbox
 example 5-17
Example
 current regulation 5-15
 Electronic gearbox 5-17
 Function codes FC 5-10
 Homing 5-14
 manual displacement 5-18
 point-to-point mode 5-12
 Profile velocity 5-13
 speed regulation 5-16

F

Factory settings 4-2
Fields, data frame 5-3
Function codes 5-5
Function tests 4-3

H

Homing
 example 5-14

I

Intended use 2-1

M

Maintenance 7-1
Manual displacement
 example 5-18
Master 5-1
Messages
 asynchronous errors 6-2
Modbus message 5-3
Modbus network 5-1
Modbus transmission technology 5-1

N

Network operation 4-3
Node address 4-2
Node guarding 4-3

O

Operation 5-1

P

Parameter addresses 5-1
parity 4-2
Point-to-point mode
 example 5-12
Profile velocity
 example 5-13
Protocol errors 6-1
Protocols for the modbus, in general 5-2

Q

Qualifications, personnel 2-1

R

Read Device Identification 5-9
Read Multiple Registers 5-5
Read Write Multiple Registers 5-8
Request 5-2, 5-3
Response 5-2, 5-3

S

Semi-duplex operation 4-2
Service 7-1
Settings 4-2
Slave 5-1
Speed regulation
 example 5-16
Standards 1-1
stop bits 4-2
Synchronous errors 6-2

T

Telegram 5-3

Terms 8-1
Timeout 5-19
Transmission format 4-2
Troubleshooting 6-1

U

Unit address 5-1

W

Word sequence 4-3
Write Multiple Registers 5-7

