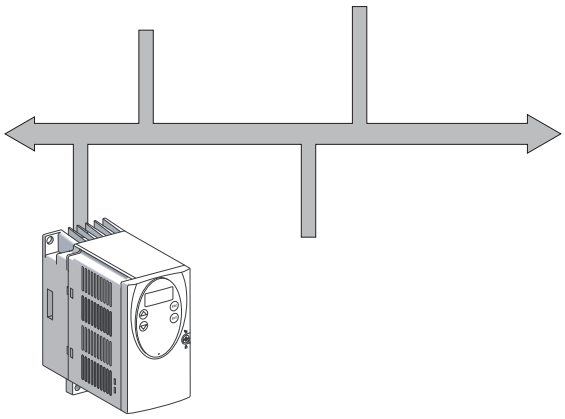
Telemecanique

Technical Documentation



Fieldbus manual

Protocol for AC servo drive

LXM05 Modbus USA

Document: BLMT00005

Edition: V1.1, 04.2007

a brand of Schneider Electric

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.

See safety section for additional critical instructions.

Not all product variants are available in all countries.
Please consult the current catalogue for information on the availability of product variants.

We reserve the right to make changes during the course of technical developments.

All details provided are technical data and not promised characteristics.

In general, product names must be considered to be trademarks of the respective owners, even if not specifically identified as such.

BLMT00005, V1.1, 04.2007

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 Safety instructions 2-2

3 Installation

4 Commissioning

4.1 Fieldbus settings 4-2

4.2 Starting network operation 4-4

4.3 Running function test 4-4

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 operating modes 5-12
5.4.1 Profile position operating mode 5-12
5.4.2 Operating mode Profile velocity 5-13
5.4.3 Operating mode Homing 5-14

5.5 Examples of manufacturer specific operating modes 5-15
5.5.1 Current control mode 5-15

BLMT00005, V1.1, 04.2007

5.5.2 Speed control operating mode 5-16
5.5.3 Electronic gearbox operating mode 5-17
5.5.4 Jog 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 Glossary

8.1 Terms and Abbreviations 8-1

9 Index

BLMT00005, V1.1, 04.2007

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.

BLMT00005, V1.1, 04.2007

1 Introduction

1.1 Modbus

This manual describes the fieldbus handling for products in the fieldbus 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.

Fieldbus 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 fieldbus handling of products which are responded to by the modbus:

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

1.3 Literature

device
- LXM05A Product manual

Modbus
- Modicon Modbus Protocol Reference Guide, PI-MBUS-300 Rev. J, June 1996, MODICON, Inc., USA
- http://www.modicon.com

BLMT00005, V1.1, 04.2007

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 system must not be commissioned and operated until completion of installation in accordance with the EMC regulations and the specifications in this manual.

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).

BLMT00005, V1.1, 04.2007

2.3 Safety instructions

⚠ DANGER

ELECTRIC SHOCK, FIRE OR EXPLOSION

- Only qualified personnel who are familiar with and understand the contents of this manual 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 at DC bus and check for <45V. (The DC bus LED is not a safe indication for absence of the DC bus voltage).
- The system manufacturer is responsible for compliance with all applicable regulations relevant to grounding the drive system.
- Many components, including printed wiring boards, operate at mains 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.

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

⚠ DANGER

RISK OF INJURY BY COMPLEX SYSTEM

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

- Only start the system if there are no persons in the operating zone of the moving components and the system can be operated safely.

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

BLMT00005, V1.1, 04.2007

⚠ WARNING

LOSS OF CONTROL

- The designer of any control scheme must consider the potential failure modes of control paths and, for certain critical control functions, provide a means to achieve a safe state during and after a path failure. Examples of critical control functions are emergency stop and overtravel stop.
- Separate or redundant control paths must be provided for critical control functions.
- System control paths may include communication links. Consideration must be given to the implications of unanticipated transmission delays or failures of the link. *

Each implementation of LXM05* must be individually and thoroughly tested for proper operation before being placed into service.

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

*** For additional information, refer to NEMA ICS 1.1 (latest edition), Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control and to NEMA ICS 7.1 (latest edition), Safety standards for Construction and Guide for Selection, Indtallation and Operation of Adjustable-Speed Drive Systems.**

⚠ 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 fieldbus and PLC cycles not operating synchronously.

- Please observe the notes concerning the operation using PLC.

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

BLMT00005, V1.1, 04.2007

BLMT00005, V1.1, 04.2007

3 Installation

⚠ WARNING

LOSS OF CONTROL

- The designer of any control scheme must consider the potential failure modes of control paths and, for certain critical control functions, provide a means to achieve a safe state during and after a path failure. Examples of critical control functions are emergency stop and overtravel stop.
- Separate or redundant control paths must be provided for critical control functions.
- System control paths may include communication links. Consideration must be given to the implications of unanticipated transmission delays or failures of the link. *
- Each implementation of LXM05* must be individually and thoroughly tested for proper operation before being placed into service.

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

* **For additional information, refer to NEMA ICS 1.1 (latest edition), Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control and to NEMA ICS 7.1 (latest edition), Safety standards for Construction and Guide for Selection, Indtallation and Operation of Adjustable-Speed Drive Systems.**

⚠ 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.

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

For information on device installation and connecting the device to the fieldbus see the product manual.

BLMT00005, V1.1, 04.2007

BLMT00005, V1.1, 04.2007

4 Commissioning

⚠ DANGER

RISK OF INJURY BY COMPLEX SYSTEM

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

- Only start the system if there are no persons in the operating zone of the moving components and the system can be operated safely.

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

⚠ WARNING

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 product manual.
- 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 fieldbus communication: Bit 0 is far right (least significant). Bit 15 is far left (most significant).
- Check the use of the word sequence during fieldbus communication:
- Do not establish a fieldbus connection before you have understood the communications principles.

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

BLMT00005, V1.1, 04.2007

4.1 Fieldbus 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 fieldbus participants can be connected to the bus. Each fieldbus 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 fieldbus participants in the network receive data from, but do not answer.

Parameter Name **Code** **HMI menu, Code**	**Description**	**Unit** **Minimum value** **Default value** **Maximum value**	**Data type** **R/W** **persistent** **Expert**	**Parameter address via fieldbus**
MBadr MBAD COM-nbAd	Modbus address() valid addresses : 1 to 247	- 1 1 247	UINT16 R/W per. -	CANopen 3016:4$_h$ Modbus 5640

Baud rate

The baud rate must be set to the same value for all fieldbus participants.

Parameter Name **Code** **HMI menu, Code**	**Description**	**Unit** **Minimum value** **Default value** **Maximum value**	**Data type** **R/W** **persistent** **Expert**	**Parameter address via fieldbus**
MBbaud MBBD COM-nbbd	Modbus baud rate() Allowed baud rates: 9600 19200 38400 NOTE: A change of the setting is not activated until the unit is switched on again.	- 9600 19200 38400	UINT16 R/W per. -	CANopen 3016:3$_h$ Modbus 5638

Data bits;stop bits and parity

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

BLMT00005, V1.1, 04.2007

Parameter Name Code HMI menu, Code	Description	Unit Minimum value Default value Maximum value	Data type R/W persistent Expert	Parameter address via fieldbus
MBformat MBFO COM-MbFo	Modbus data format() **1 / 8Bit NoParity 1Stop / 8n1**: 8 bit, no parity bit, 1 stop bit **2 / 8Bit EvenParity 1Stop / 8e1**: 8 bit, even parity bit, 1 stop bit (default) **3 / 8Bit OddParity 1Stop / 8o1**: 8 bit, odd parity bit, 1 stop bit **4 / 8Bit NoParity 2Stop / 8n2**: 8 bit, no parity bit, 2 stop bits NOTE: A change of the setting is not activated until the unit is switched on again.	- 1 2 4	UINT16 R/W per. -	CANopen 3016:5$_h$ Modbus 5642

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

Parameter Name Code HMI menu, Code	Description	Unit Minimum value Default value Maximum value	Data type R/W persistent Expert	Parameter address via fieldbus
MBnode_guard -	Modbus Node Guard() Connection monitoring 0 : inactive (default) >0 : Monitoring time	ms 0 0 10000	UINT16 R/W - -	CANopen 3016:6$_h$ Modbus 5644

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	Description	Unit Minimum value Default value Maximum value	Data type R/W persistent Expert	Parameter address via fieldbus
MBdword_order MBWO COM-MbLo	Modbus word sequence for double words (32 bit values)() Send High Word first or Low Word first **0 / HighLow / HiLo**: HighWord-LowWord, High Word first -> Modicon Quantum (default) **1 / LowHigh / LoHi** : LowWord-HighWord Low Word first -> Premium, HMI (Telemecanique)	- 0 0 1	UINT16 R/W per. -	CANopen 3016:7$_h$ Modbus 5646

Detailed description A detailed description of the settings is given in the "Commissioning" chapter of the product manual.

4.2 Starting network operation

Network operation is started via a master. This can be a PLC or a PC which uses the corresponding user software to send fieldbus 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 product 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 fieldbus parameters should be set via the HMI or by using the installation software, otherwise communication with the unit in the fieldbus network will not be possible.

If the new unit has already been configured for fieldbus operation, then it will recognise the values of the fieldbus 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.

BLMT00005, V1.1, 04.2007

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 fieldbus participants which control the data traffic within a network. Examples of masters are:

- automation devices, e.g. PLCs
- PCs

Slave Slaves are passive fieldbus 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 fieldbus, it must address the slave. The node address of the slave is contained within the data frame. Notes on setting the address of the slave can be found on page 4-2.

Parameter addresses Communications between fieldbus master and Slave is based on parameters to which a unique parameter address is assigned. There are write and read parameters. A description of all parameters available for this product (=slave) with the corresponding Modbus addresses can be found in the "Parameters" chapter of the product manual.

The Modbus address is decimal. It must be converted to hexadecimal for input. Values input in hexadecimal are identified in this manual by an attached "$_h$". The examples with the Premium are also identified with "16#" before the number.

5.1.2 Modbus transmission technology

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

Data exchange between the fieldbus participants takes place in the master-slave process. Only the master can send commands (queries). The master can address every slave 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. 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.

BLMT00005, V1.1, 04.2007

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 by the slave device. 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 device 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 device. 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.

BLMT00005, V1.1, 04.2007

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 device then it is called sending data frame or request. In reply to this request, this device sends a response, the reception data frame.

A Modbus RTU data frame consists of the following fields:

<SlaveAdr> <FC> <Data> <CRC>

Field name	Description	Number of bytes
<SlaveAdr>	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 in the response.

<FC> The function code determines which Modbus service the slave device 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> For error-checking with Modbus RTU the "**C**yclic **R**edundancy **C**hecksum" (CRC) is formed from the transferred fields <SlaveAdr>, <FC> and <Data>. It comprises a CRC16 with a generator polynome $A001_h$, which is calculated from the algorithm shown in the following illustration.

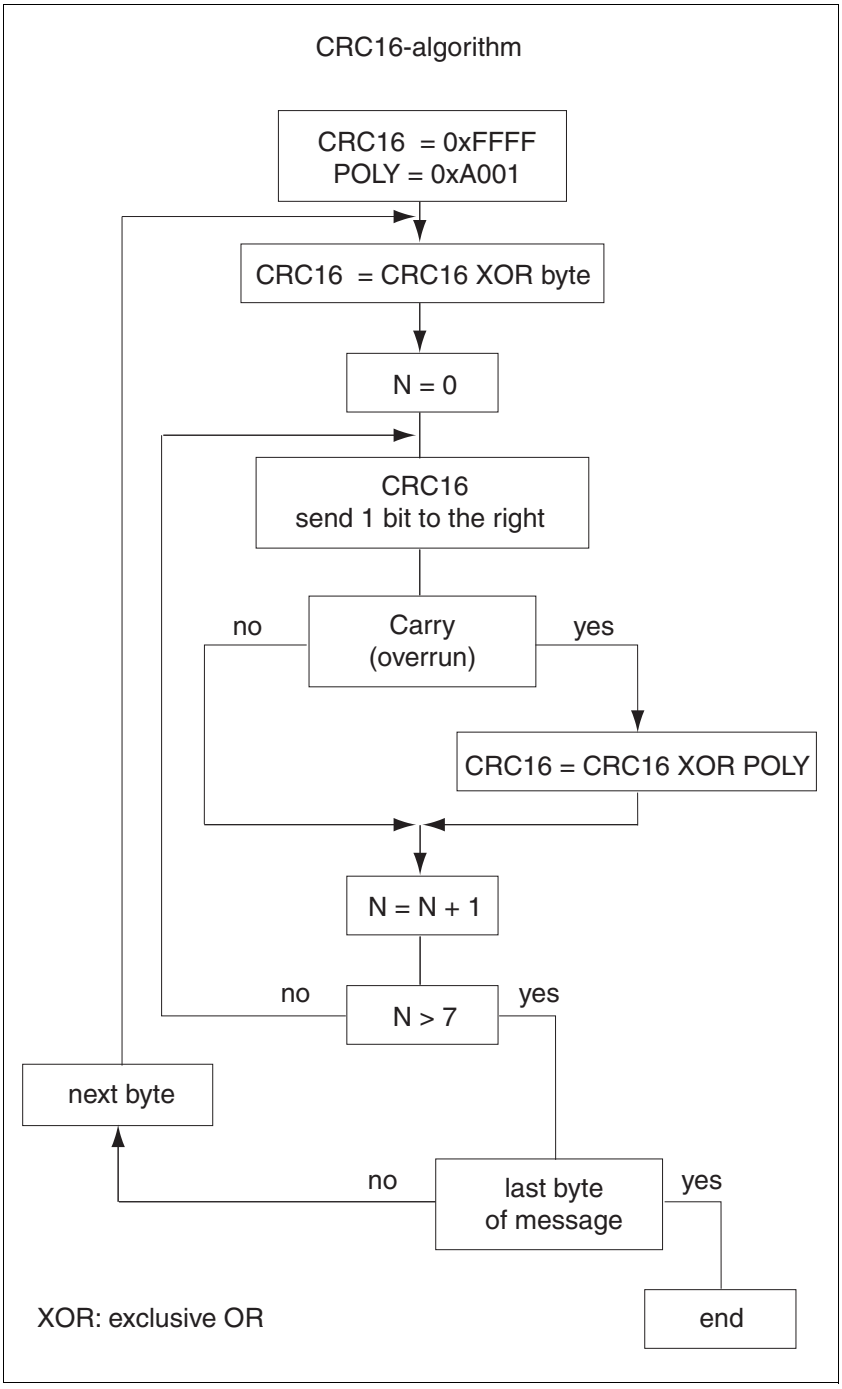


Figure 5.1 CRC16 algorithm

BLMT00005, V1.1, 04.2007

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 device.

Structure of request: <FC> <1. ReadingAddress> <NumberParameter>

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

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

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

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

Example for Premium in PL7

► Read out the current motor position:

The Modbus parameter address for the current position (`_p_act`) is 7700 (16#1E14). 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 parameter address 16#1E14
from the device with the address (ADR)
READ_VAR (ADR, '%MW', 16#1E14, 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 product manual under the chapter "Parameters".

5.2.2 FC 8 (Diagnostics)

This function code allows diagnostics data to be read by the slave.

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

Field	Bytes	Value	Description
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	Description
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	device-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	Mute slave
05..09	(reserved)	–
10	Clear Counters and Diag-nostic 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 device.

Structure of request: <FC> <1. WriteAddress> <NumberParameter> <NumberBytes> <Data>

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

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

Field	Bytes	Value	Description
FC	1	16 = 10_h	Response code
ParamAddress	2	(various)	inverted from request
NumberParameter	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 (profile position mode):

The Modbus parameter address for the target position `PPp_targetusr` is 6940 (16#1B1C). 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 parameter address 16#1B1C of
the device with the address (ADR)
WRITE_VAR (ADR, '%MW', 16#1B1C, 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 product 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 device in reading and writing modes.

Structure of request: <FC> <1. Read Address> <NumberParameter> <1. WriteAddress> <NumberParameter> <NumberBytes> <Data>

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

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

Field	Bytes	Value	Description
FC	1	23 = 17_h	Response code
NumBytes	1	2 * n	Number of data bytes
Data	2 * n	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 product manual under the chapter "Parameters".

5.2.5 FC 43 (Read Device Identification)

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

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

Field	Bytes	Value	Description
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 IDvendor name, product code, revision

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

Field	Bytes	Value	Description
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 Levelfixed value
MoreFoll	1	00	More Followsfixed value, because message frame length < 255
NextObjID	1	00	Next Object IDfixed 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
00_h	vendor name	Manufacturer name
01_h	product code	"xxxxxxxxxxxx" (see type code
03_h	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 parameter 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 -> `FLT_err_num (15362)` / `FLT_class (15364)` / `FLT_Time (15366)` / `FLT_Qual (15368)`. Since all the error information has Modbus addresses in ascending sequence, one read request as follows is sufficient:

Field	Bytes	Value	Description
FC (Request Code)	1	3	Request code (Multiple Register READ)
ParamAddress	2	15362dec ($3C02_h$)	First parameter address to be read
NumberParameter	2	4 * 2 = 8	Number of 16-bit parameters to be read = 8, i. e. read 16 byte data

Table 5.3 Example 1, FC3 Request

Field	Bytes	Value	Description
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	FLT_err_num, 15362 (error number) FLT_class, 15364 (error class) FLT_Time, 15366 (time of error) FLT_Qual, 15368 (error identification)

Table 5.4 Example 1, FC3 Positive Response

Example 2 Write the software limit switch -> `SPVswLimPusr (1544)`/ `SPVswLimNusr(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	Description
FC (Request Code)	1	16	Request code (Multiple Register WRITE)
ParamAddress	2	1544dec (608_h)	First parameter address to be written
NumberParameter	2	2 * 2 = 4	Number of parameters = 4 (8 bytes data)
NumBytes	1	8	Number of bytes: 8 bytes data
Data	8	32Bit value 32Bit value	SPVswLimPusr, 1544 SPVswLimNusr, 1546

Table 5.5 Example 2, FC16 Request

Field	Bytes	Value	Description
FC (Request Code)	1	16	Response code (Multiple Register WRITE)
ParamAddress	2	1544dec (608_h)	Modbus parameter address
NumberParameter	2	2 * 2 = 4	Number of parameters = 4 (8 bytes data)

BLMT00005, V1.1, 04.2007

Table 5.6 Example 2, FC16 Positive Response

5.4 Examples for standardised operating modes

5.4.1 Profile position operating mode

Example Node address 1.

Description Fieldbus command / parameter name (address)	Value
▶ Acceleration ramp 2000 rpm*s FC16 / RAMPacc (1556)	0000 07D0$_h$
▶ Delay ramp 4000 rpm*s FC16 / RAMPdecel (1558)	0000 0FA0$_h$
▶ Limit set speed 6000 rpm FC16 / RAMPn_max (1554)	0000 1770$_h$
▶ Set speed 4000 rpm FC16 / PPn_target (6942)	0000 0FA0$_h$
▶ Disable Voltage FC16 / DCOMcontrol (6914)	0000 0000$_h$
▶ Shut Down FC16 / DCOMcontrol (6914)	0000 0006$_h$
▶ Operation Enable FC16 / DCOMcontrol (6914)	0000 000F$_h$
▶ Checking operating status [1)] FC 3 / DCOMstatus (6916)	
◁ Operation state active	0000 0007$_h$
▶ Start operating mode FC16 / DCOMopmode (6918)	0000 0001$_h$
▶ Checking operating mode [1)] FC 3 / _DCOMopmd_act (6920)	
◁ Operating mode active	0000 0001$_h$
▶ Store new set position FC16 / PPp_targetusr (6940)	0000 0030$_h$
▶ Start absolute positioning FC16 / DCOMcontrol (6914)	0000 005F$_h$
▶ Checking target position [1)] FC 3 / DCOMstatus (6916)	
◁ Target position reached (Bit 10 = 1)	xxxx x4xx$_h$
▶ Reset start bit FC16 / DCOMcontrol (6914)	0000 000Fh

1) It must be checked cyclically.

BLMT00005, V1.1, 04.2007

5.4.2 Operating mode Profile velocity

Example Node address 1.

Description Fieldbus command / parameter name (address)	Value
▶ Acceleration ramp 2000 rpm*s FC16 / RAMPacc (1556)	0000 07D0h
▶ 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 000Fh
▶ Checking operating status [1)] FC 3 / DCOMstatus (6916)	
◁ Operation state active	0000 0007_h
▶ Start operating mode FC16 / DCOMopmode (6918)	0000 0003_h
▶ Checking operating mode [1)] FC 3 / _DCOMopmd_act (6920)	
◁ Operating mode active	0000 0003_h
▶ Transmission set speed 1000 rpm FC16 / PVn_target (6938)	0000 03E8h
▶ Checking target speed [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
▶ Checking target speed [1)] FC 3 / DCOMstatus (6916)	
◁ Target position reached (Bit 10 = 1)	xxxx $x4xx_h$

1) It must be checked cyclically.

5.4.3 Operating mode Homing

Example Node address 1.

Description Fieldbus 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 000A$_h$
▶ Disable Voltage FC16 / DCOMcontrol (6914)	0000 0000$_h$
▶ Shut Down FC16 / DCOMcontrol (6914)	0000 0006$_h$
▶ Operation Enable FC16 / DCOMcontrol (6914)	0000 000Fh
▶ Checking operating status [1)] FC 3 / DCOMstatus (6916)	
◁ Operation state active	0000 0007$_h$
▶ Start operating mode FC16 / DCOMopmode (6918)	0000 0006$_h$
▶ Checking operating mode [1)] FC 3 / _DCOMopmd_act (6920)	
◁ Operating mode active	0000 0006$_h$
▶ Select reference displacement method, LIMN (17) FC16 / HMmethod (6936)	0000 0011$_h$
▶ Start homing FC16 / DCOMcontrol (6914)	0000 001F$_h$
▶ Checking homing [1)] FC 3 / DCOMstatus (6916)	
◁ Drive has a valid reference point (Bit 12 = 1)	xxxx 1xxx$_h$
▶ Reset start bit FC16 / DCOMcontrol (6914)	0000 000Fh

1) It must be checked cyclically.

5.5 Examples of manufacturer specific operating modes

5.5.1 Current control mode.

Example Node address 1.

Description Fieldbus 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 000Fh
▶ Checking operating status [1)] FC 3 / `DCOMstatus` (6916)	
◁ Operation state active	0000 0007_h
▶ Start operating mode (-3) FC16 / `DCOMopmode` (6918)	FFFF $FFFD_h$
▶ Checking operating mode [1)] FC 3 / `_DCOMopmd_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 03E8h

1) It must be checked cyclically.

5.5.2 Speed control operating mode

Example Node address 1.

Description Fieldbus 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 000Fh
▶ Checking operating status [1] FC 3 / DCOMstatus (6916)	
◁ Operation state active	0000 0007_h
▶ Start operating mode (-4) FC16 / DCOMopmode (6918)	FFFF $FFFC_h$
▶ Checking operating mode [1] FC 3 / _DCOMopmd_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 03E8h

1) It must be checked cyclically.

BLMT00005, V1.1, 04.2007

5.5.3 Electronic gearbox operating mode

Example Node address 1.

Description Fieldbus 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 000Fh
▶ Checking operating status [1] FC 3 / DCOMstatus (6916)	
◁ Operation state active	0000 0007_h
▶ Start operating mode (-2) FC16 / DCOMopmode (6918)	FFFF $FFFE_h$
▶ Checking operating mode [1] FC 3 / _DCOMopmd_act (6920)	
◁ Operating mode active	FFFF $FFFE_h$
▶ Gearbox activation with immediate synchronisation FC16 / GEARreference (6948)	0000 0001_h
▶ Transfer denominator FC16 / GEARdenom (9734)	0000 0003_h
▶ Transfer numerator FC16 / GEARnum (9736)	0000 0002_h

1) It must be checked cyclically.

5.5.4 Jog mode

Example Node address 1.

Description Fieldbus 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 000Fh
▶ Checking operating status [1] FC 3 / DCOMstatus (6916)	
◁ Operation state active	0000 0007$_h$
▶ Start operating mode (-1) FC16 / DCOMopmode (6918)	FFFF FFFF$_h$
▶ Checking operating mode [1] FC 3 / _DCOMopmd_act (6920)	
◁ Operating mode active	FFFF FFFF$_h$
▶ Jog (clockwise rotation, slow) FC16 / JOGactivate (6930)	0000 0001$_h$
▶ Jog (clockwise rotation, fast) FC16 / JOGactivate (6930)	0000 0005$_h$

1) It must be checked cyclically.

5.6 Connection monitoring

⚠ WARNING

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.

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

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 device will stop the process initiated by the fieldbus and will initiate a "Quick Stop".

The time interval for the timeout message can be set with the parameter `MBnode_guard (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 device manufacturer must then provide alternative control paths for stopping and controlling the motor.

Received character check

The unit checks a fieldbus 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.

BLMT00005, V1.1, 04.2007

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 responses. 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

BLMT00005, V1.1, 04.2007

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 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 a synchronous 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.

BLMT00005, V1.1, 04.2007

7 Service, maintenance and disposal

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

BLMT00005, V1.1, 04.2007

BLMT00005, V1.1, 04.2007

8 Glossary

8.1 Terms and Abbreviations

Actual position	Current absolute or relative position of moving components in the drive system.
Address	Memory location which can be accessed by its unique number. See also Slave address.
API	Application Program Interface
ASCII	American Standard Code for Information Interchange; Standard for coding text characters
Asynchronous error	Error detected and reported by the internal controller monitoring equipment.
CRC	Cyclical Redundancy Check, error checking
Data frame	Serially transmitted data packet with unique start and end identification. The structure depends on the protocol in use.
Default value	Factory settings.
Direction of rotation	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.
Drive system	The drive system consists of the controller, power amplifier and motor.
Electronic gear	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.
EMC	Electromagnetic compatibility
Error class	Classification of operational faults into groups corresponding to the error responses
Field	Bytes of a message that belong together because of their content.
Fieldbus	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.
Half duplex	Bidirectional data transmission in which only one user can transmit at any time.
HMI	Human Machine Interface, handheld operating unit.
I^2t-monitoring	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/O	Inputs/Outputs
Inc	Increment

Incremental signals	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.
Index pulse	Encoder signal for referencing the rotor position in the motor. The encoder sends one index pulse per revolution.
LED	Light-Emitting Diode
Limit switch	Switch that signals an overrun of the permissible travel range.
LRC	Longitudinal Redundancy Check, error checking
LSB	Least Significant Bit, the least significant bit of a bit sequence, e. g. of a byte
Master	Active bus user that controls the data traffic in the network.
Node Guarding	Monitoring function with slave at an interface for cyclic communication.
Parameter	Device functions and values that can be set and called by the user.
Power amplifier	A device that generates current for controlling the motor in accordance with the positioning signals from the controller.
PC	Personal Computer
PLC	Programmable Logic Controller
Profibus	Standardised open fieldbus compliant with EN 50254-2 over which drives and other devices from different manufacturers communicate with one another.
Protocol	Guideline that specifies the format required for transmitting data.
PWM	Pulse Width Modulation
Quick Stop	Quick stop, function used to provide quick braking of the motor via a command or in the event of a fault.
Register	Memory range of specified size (generally 8, 16 or 32 bits) for temporary storage of data transmitted from one system unit to another.
RS485	Fieldbus interface compliant with EIA-485, which enables serial data transmission with multiple devices.
RTU	Remote Terminal Unit
Slave	Passive bus user that receives control commands and sends data to the master.
Slave address	Direct communication between master and slave devices is only possibly after assignment of addresses.
Synchronous errors	Error reported by the controller if it is unable to execute a command sent by the master.
Timeout	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.

BLMT00005, V1.1, 04.2007

9 Index

A

Abbreviations 8-1
Asynchronous errors 6-2

B

Baud rate 4-2

C

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

D

Data bits 4-2
Data frame 5-3
device address 5-1
Diagnostics 5-6, 6-1
Directives and standards 1-1
Disposal 7-1

E

Electronic gearbox
    example 5-17
Example
    current control 5-15
    Electronic gearbox 5-17
    Function codes FC 5-10
    Homing 5-14
    jog 5-18
    profile position 5-12
    Profile velocity 5-13
    speed control 5-16

F

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

G

Glossary 8-1

H

Homing
    example 5-14

BLMT00005, V1.1, 04.2007

I

Intended use 2-1
Introduction 1-1

J

Jog
example 5-18

M

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

N

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

O

Operation 5-1

P

Parameter addresses, hexadecimal, decimal 5-1
parity 4-2
Profile position
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 control
example 5-16

BLMT00005, V1.1, 04.2007

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

W

Word sequence 4-3
Write Multiple Registers 5-7

BLMT00005, V1.1, 04.2007

BLMT00005, V1.1, 04.2007