ebmpapst

Modbus parameter specifications
for
"ebm-papst series 84 / 112 / 150 / 200"

V5.00

Document: ext000757333.doc

Excerpt

October 1, 2012

Table of contents

Preface

This document details the specifications for the Modbus parameters of the "series 84 / 112 / 150 / 200" unit from ebm-papst.

This excerpt differs from the complete document in that the parameters for internal use by ebm-papst are not described here. These parameters are marked by the word "reserved".

Knowledge of general Modbus specifications is required:
- MODBUS over Serial Line Specification & Implementation guide V1.0
- MODBUS Application Protocol Specification V1.1

These documents are available on the Internet at modbus.org

The general Modbus specifications form the basis for this document and are valid in full with the exception of the restrictions described in this document.

History of changes/modifications

Version	Changes
1.00	Document annex
1.01	
1.02	
2.00	Change in encoding of the following register from 8 bit to 16 bit (MSB was previously always 0): a) Holding register D001 Default set value D114 / D115 Set value (EEPROM) D12A - D12D Potentiometer characteristic b) Input register D010 Actual speed D01A Current set value D01B Actual sensor value Change in encoding of I factor control parameter (resolution 1 to 1/65536 instead of 256 to 1/256) Change in controller equalisation (kp no longer influences I portion) Definition for ramping-up/down curves changed as set value now assumes 65536 steps instead of 256 steps Default set value no longer stored in parameter "Set value (stored)" on write access, but rather in "Set value (EEPROM)"; parameter "Set value (stored) no longer used Identification 0x0002 redefined for Modbus parameter specifications V2.00
2.01	Added: the following holding registers: D135 Max. power D136 - D138 Parameters for derating max. power D13B Max. winding current D13C - D13F Potentiometer characteristic "Night" D140 - D143 0 to 10V output characteristic Redefinition of holding register D12F: "Control limitation" The following holding registers have been deleted: D11D Max. DC-link current Added: input register D021 Current power Added: new bits in input register D012 Warning: P_Limit and I_Limit
2.02	
3.00	Added: the following holding registers: D009 Operating hours counter D00A Operating minutes counter D145 Limit speed for running monitor D15C Bypass function on/off

Version	Changes
	D15D Bypass function set value D15E Bypass function time lag Function of the following holding registers changed D180 Operating hours counter: is now merely backup D181 Operating minutes counter: is now merely backup Added: new bits in input register D012 Warning: n_Low, phase
3.01	
3.02	Added: the following holding registers: D148 Running direction source D147 Actual sensor value source Changed: the following holding registers: D104 Parameter set source (new value: 2) D12E Source of control function (new value: 2) D130 Output function 0 to 10V (new value: 2) D184ff Error history: Bit 15 "Brake" deleted Bit 14 "UeHigh" added D1A2 -D1A3 Fan serial number D1A4 Fan production date Added: the following input registers: D023 Actual sensor value 1 D024 Actual sensor value 2 Changed: the following input registers: D011 Motor status: Bit 15 "Brake" deleted Bit 14 "UeHigh" added Bit 13 "UeLow" added D012 Warning: Bit 7 "Brake" added Renamed: the following holding registers: D102 Preferred running direction (instead of running direction) D104 Parameter set source (instead of day/night switch external/internal) D105 Internal parameter set (instead of day/night internal) D12E Source of control function (instead of Control function switch external/internal) D15C Emergency operation function on/off (instead of Bypass function on/off) D15D Emergency operation set value (instead of Bypass function set value) D15E Emergency operation time lag (instead of Emergency operation function time lag) All "day" suffixes renamed to "parameter set 1" All "night" suffixes renamed to "parameter set 2" Renamed: the following input registers: D01D Current parameter set (instead of day/night parameter set)
4.00	New command: 0x46 Write Single Register Addressed by Serial No. Allow change to interface settings

Version	Changes
	Emergency running function extended to analogue input (cable break detection) Added: the following holding registers: D149 Transmission rate D14A Parity configuration D15F Potentiometer characteristic, limit value for cable break Added: the following input registers: D012 Warning: Bits 10 "Cable br."; 1 "L_high" added Write authorisation changed for holding register D145 Limit speed for running monitor (customer)
5.00	New commands: 0x43 Read Holding Register Addressed By Serial No. 0x44 Read Input Register Addressed By Serial No. 0x50 Write Multiple Register Addressed By Serial No. Broadcast address setup for all commands "... Addressed By Serial No." Maximum telegram length increased to 23 bytes Changed: the following holding registers: D130 0 to 10V output function / speed monitoring: new value 3 (actual speed -> pulses / revolution); MSB: pulses / revolution added D149 Transmission rate (new values: 6, 7 -> 57600, 115200 bit/sec) D182 - D19F Error history: Bits 14 "UeHigh", 11 "UzHigh" and 9 "TFEI" deleted Added: the following holding registers: D14D Motor temperature power derating start D14E Motor temperature power derating end D150 Sheding function D151 Max. starting modulation level D152 Number of startup attempts D153 Relay drop-out delay D155 Max. power D15B Emergency operation running direction Renamed: the following holding registers: D135 Max. **permitted** power D136 Max. power for ~~temperature~~ derating end D137 **Module** temperature power derating start D138 **Module** temperature power derating end Changed: the following input registers: D011 Motor status: Bits 14 "UeHigh", 13 "UeLow, 11 "UzHigh" and 9 "TFEI" deleted D012 Warning: Bit 15 "Shf", 14 "UeHigh", 12 "UzHigh" and 11 "Heating" added D016 Motor temperature (format: signed integer; enable negative values)

1 Frame protocol

Data are transferred using the Modbus protocol defined in these specifications exclusively in an environment defined as a master/slave system. The orderly progression of data is defined by the master. A slave is required to respond to its command prompt. For this reason, it is important to ensure that no slave address is assigned more than once when constructing a system.

A twisted pair wire with RS485 standard should preferably be used.

Only RTU transmission mode is supported (see MODBUS over Serial Line Specification & Implementation guide V1.0, chapter 2.5.1)

ASCII transmission mode is not supported!

1.1 Structure of a byte

According to the MODBUS over Serial Line Specification & Implementation guide V1.0. a byte has the following structure:

Start (low)	Bit 0 (LSB)	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7 (MSB)	Parity	Stop (high)

The definition of the parity bit ("Even", "Odd", "None") can be set with the parameter "Parity configuration" (see 2.38.2).

The transmission rate is variable and can be set with the parameter "transmission rate" (see 2.38.1).

1.2 Communication process

The MODBUS over Serial Line Specification & Implementation guide V1.0 defines the following framework for the transmission protocol:

Command from master:	Start > 3.5 char	Address 8 bits	Cmd 8 bits	Data N * 8 bits	CRC L 8 bits	CRC H 8 bits
Response from fan:	Start > 3.5 char	Address 8 bits	Cmd 8 bits	Data N * 8 bits	CRC L 8 bits	CRC H 8 bits

In contrast to the general specifications, the maximum telegram length is 23 bytes!

1.2.1 Command from master

A master device is, for example, a PC or a control device.

Initial synchronisation:

A transmission pause of at least 3.5 bytes is used for initial synchronisation.
The following byte is then interpreted as the first byte of a frame (i.e. address).
The pause between the individual bytes of a frame may be a maximum of 1.5 bytes.

Address:

The address field has a size of 8 bits.
Address values from 1 to 247 are permitted
The address 0 is reserved for broadcast commands (i.e. commands to all fans in the network).

Command:

The following commands from the "MODBUS Application Protocol Specification V1.1" general specifications are supported:

Code	Command
0x03	Read holding register
0x04	Read input register
0x06	Write single register
0x08	Diagnostics
0x10	Write multiple register

Other commands are not supported.

The following additional commands are defined by ebm-papst:

Code	Command
0x43	Read Holding Register Addressed By Serial No.
0x44	Read Input Register Addressed By Serial No.
0x46	Write Single Register Addressed By Serial No.
0x50	Write Multiple Register Addressed By Serial No.

Data:

Depending on the command concerned, the number of data bytes and their meaning may differ.
Please refer to 1.3.Commands

CRC L / CRC H

A CRC checksum is defined via the complete telegram.
The polynomial for defining the checksum is $1 + x^2 + x^{15} + x^{16}$ (i.e. XOR link to 0xA001).
The initial value is 0xFFFF.

The low byte of checksum is transmitted first, then the high byte.

More detailed information about calculating the checksum can be found in the "MODBUS over Serial Line Specification & Implementation guide V1.0".

1.2.2 Response from fan

A fan will only respond if
- It receives a message through its own address.
  No response is sent at the broadcast address.
- The telegram length is at most 23 bytes.
- The correct number of data bytes have been sent so that the telegram can be interpreted.
- The checksum has been correctly recognised.

Initial synchronisation:

After the command from the master has been completed, the fan will wait for *at least* one transmission pause of 3.5 bytes. Depending on the command and on the processing time, the pause may be much longer (until the fan has received all the data it has requested)

Address:

The address is repeated by the command from the master (i.e. its own fan address)

Command:

If the command can be processed, the command code will be repeated.

If the command cannot be processed, the fan will respond with an exception.
Here, the MSB is set to command.
The command byte is then, for example, 0x83 for the command "Read holding register (0x03)".

Data:

Depending on the command concerned, the number of data bytes and their meaning may differ.
Please refer to 1.3 Commands.

CRC L / CRC H

A CRC checksum is defined via the complete telegram.
The way the checksum is defined is no different from the procedure described above for the command from PC.

ebm-papst Mulfingen GmbH & Co. KG | Bachmühle 2 · 74673 Mulfingen, Germany · Phone: +49-7938-81-0 · Fax: +49-7938-81-110 · www.ebmpapst.com · info1@de.ebmpapst.com

1.3 Commands

1.3.1 Read holding register

Command code: 0x03

This command is used to read out the content of a number of holding registers.
Holding registers are parameters that can be both read and write-accessed

Command from master:

4 data bytes are transmitted:
- 1st holding register MSB address
- 1st holding register LSB address
- Number of MSB addresses to be read
- Number of LSB addresses to be read

The description of the holding registers can be found at a later point.

Response from fan:

The following data bytes are transmitted:
- Byte count (number of addresses to be read * 2)
- Data in 1st holding register MSB
- Data in 1st holding register LSB

Optional:
- Data from the following holding registers (0 to n)

Exception codes:

In case of error, only one data byte (the exception code) will be transmitted

Exception codes:

0x02: Permissible range of the holding registers 0xD000 to 0xD37F exceeded
0x03: Maximum telegram length for answer (23 bytes) exceeded
i.e. either more than 9 holding registers or 0 holding registers were requested.
0x04: A holding register cannot be read due to a defect in the electronics

1.3.2 Read input register

Command code: 0x04

This command is used to read out the content from multiple input registers.
Input registers are parameters that can be read-accessed.

Command from master:

4 data bytes are transmitted:
- 1st input register MSB address
- 1st input register LSB address
- Number of MSB addresses to be read
- Number of LSB addresses to be read

The description of the input registers can be found at a later point.

Response from fan:

The following data bytes are transmitted:
- Byte count (number of addresses to be read * 2)
- Data in 1st holding register MSB
- Data in 1st holding register LSB

Optional:
- Data from the following input registers (0 to n)

Exception codes:

In case of error, only one data byte (the exception code) will be transmitted

Exception codes:

0x02: Permissible range of the input registers 0xD000 to 0xD026 exceeded
0x03: Maximum telegram length for answer (23 bytes) exceeded
i.e. either more than 9 input registers or 0 input registers were requested.

1.3.3 Write single register

Command code: 0x06

This command is used to describe the content of *one* holding register.

Command from master:

4 data bytes are transmitted:
- Holding register MSB address
- Holding register LSB address
- MSB data to be written
- LSB data to be written

The description of the holding registers can be found at a later point.

Response from fan:

4 data bytes are transmitted:
- Holding register MSB address
- Holding register LSB address
- MSB data to be written
- LSB data to be written

Exception codes:

In case of error, only one data byte (the exception code) will be transmitted

Exception codes:

0x02: Permissible range of the holding registers 0xD000 to 0xD37F exceeded
0x04: The holding register cannot be written due to a defect in the electronics
- There is no write permission in this authorisation level (password).

1.3.4 Diagnostics

Command code: 0x08

This command is used to check the Modbus function

Command from master:

The following data bytes are transmitted:
- MSB subfunction code
- LSB subfunction code
- 1 - 17 data bytes

Only subfunction code 0000 is supported!

Response from fan:

The following data bytes are transmitted:
- MSB subfunction code
- LSB subfunction code
- 1 - 17 data bytes

Exception codes:

In case of error, only one data byte (the exception code) will be transmitted

Exception codes:

0x01: Subfunction code not supported (≠ 0000)

1.3.5 Write multiple registers

Command code: 0x10

This command is used to write the content of *several* holding registers.

Command from master:

The following data bytes are transmitted:
- Holding register MSB address
- Holding register LSB address
- Number of MSB addresses to be written
- Number of LSB addresses to be written
- Byte count (number of addresses to be written * 2)
- Data to be written in 1st MSB holding register
- Data to be written in 1st LSB holding register

Optional:
- Data to be written to the following holding registers (0 to n)

The description of the holding registers can be found at a later point.

Response from fan:

4 data bytes are transmitted:
- Holding register MSB address
- Holding register LSB address
- Number of MSB addresses to be written
- Number of LSB addresses to be written

Exception codes:

In case of error, only one data byte (the exception code) will be transmitted

Exception codes:

0x02: Permissible range of the holding registers 0xD000 to 0xD37F exceeded
0x03: - The number of maximum possible registers has been exceeded
i.e. either more than 123 holding register data or 0 holding register data were defined.
- Byte count ≠ 2 * number of registers
- Number of data bytes ≠ byte count
0x04: The holding register cannot be written due to a defect in the electronics
- There is no write permission in this authorisation level (password).

1.3.6 Commands with addressing via serial number

These commands also use the fan serial numbers for addressing:
An identifier for the fan serial number (6 bytes) is transmitted in the data section of the commands.
Only the fan that has the given Modbus address and serial number will respond to and answer the command.

The command with addressing using the serial number is made up of the standard command, to which a serial number identifier is added after the command byte.

Serial number format:

For each fan, ebm-papst assigns an individual serial number. This serial number has 10 digits.

Format: YYWW00XXXX

YY : Year of production
WW : Calendar week of production
00 : Fixed value 00
XXXX : Sequential number

The first 4 digits contain the production date (year/calendar week).
With the beginning of each production week, the number XXXX starts counting up from zero and is increased by 1 for each fan. Each character can represent values from 0-9 and from A-Z. The maximum number of characters that can be encoded for each digit is thus 36, i.e. 36^4 = 1 679 616 devices / week

Example:
ebm-papst serial number: 09230012GY
Year of production: 09 (2009)
Calendar week: 23 (23)
Sequential No.: 12GY (49525)

Year of production and calendar week are each encoded as hex values in the command.
For the series number, each character is encoded as an ASCII value.

In the commands with addressing via the serial number, the serial number must be given in the following manner:

Byte No.	Meaning	Example	Corresponds to holding register
Byte 0:	[Address]		
Byte 1:	[Command code]		
Byte 2:	Year of production	0x09 (-> 2009)	D1A4 MSB
Byte 3:	Calendar week	0x17 (-> 23)	D1A4 LSB
Byte 4:	Series number 1st character	0x31 (-> 1)	D1A3 MSB
Byte 5:	Series number 2nd character	0x32 (-> 2)	D1A3 LSB
Byte 6:	Series number 3rd character	0x47 (-> G)	D1A2 MSB
Byte 7:	Series number 4th character	0x59 (-> Y)	D1A2 LSB

The order depends on the sequence in which the serial number is stored in the holding registers D1A2 - D1A4 (see 2.52.1)!

Broadcast address

If the serial number identifier value 0x00 is transmitted in one or more bytes, the corresponding part of the serial number will not be checked by the fan. All fans with the remaining part of the serial number identifier will then answer.
If all bytes are transmitted with the value 0x00, all fans will answer.

In the answer, the fans' own addresses and serial numbers will be given in place of the broadcast address / serial number 0x00.

Options for identifying the serial number:

- Printed on the fan type plate
- Read the serial number from the holding register D1A2 - D1A4 (see 2.52.1)
- Broadcast address: The fan responds with its serial number

1.3.6.1 Read Holding Register Addressed By Serial No.

Command code: 0x43

This command is used to read the content of a number of holding registers.
Holding registers are parameters that can be both read and write-accessed

In contrast to "MODBUS over Serial Line Specification & Implementation Guide V1.0", the fan also answers to a broadcast command (Modbus address = 0).

Command from master:

10 data bytes are transmitted:
- Serial No. byte 1
- Serial No. byte 2
- Serial No. byte 3
- Serial No. byte 4
- Serial No. byte 5
- Serial No. byte 6
- 1st holding register MSB address
- 1st holding register LSB address
- Number of MSB addresses to be read
- Number of LSB addresses to be read

The description of the holding registers can be found at a later point.

Response from fan:

The following data bytes are transmitted:
- Serial No. byte 1
- Serial No. byte 2
- Serial No. byte 3
- Serial No. byte 4
- Serial No. byte 5
- Serial No. byte 6
- Byte count (number of addresses to be read * 2)
- Data in 1st holding register MSB
- Data in 1st holding register LSB

Optional:
- Data from the following holding registers (0 to n)

Exception codes:

In case of error, only one data byte (the exception code) will be transmitted

Exception codes:

0x02: Permissible range of the holding registers 0xD000 to 0xD37F exceeded
0x03: Maximum telegram length for response (23 bytes) exceeded
i.e. either more than 6 holding registers or 0 holding registers were requested.
0x04: A holding register cannot be read due to a defect in the electronics

1.3.6.2 Read Input Register Addressed By Serial No.

Command code: 0x44

This command is used to read out the content from multiple input registers.
Input registers are parameters that can be read-accessed.

In contrast to "MODBUS over Serial Line Specification & Implementation Guide V1.0", the fan also answers to a broadcast command (Modbus address = 0).

Command from master:

10 data bytes are transmitted:
- Serial No. byte 1
- Serial No. byte 2
- Serial No. byte 3
- Serial No. byte 4
- Serial No. byte 5
- Serial No. byte 6
- 1st input register MSB address
- 1st input register LSB address
- Number of MSB addresses to be read
- Number of LSB addresses to be read

The description of the input registers can be found at a later point.

Response from fan:

The following data bytes are transmitted:
- Serial No. byte 1
- Serial No. byte 2
- Serial No. byte 3
- Serial No. byte 4
- Serial No. byte 5
- Serial No. byte 6
- Byte count (number of addresses to be read * 2)
- Data in 1st holding register MSB
- Data in 1st holding register LSB

Optional:
- Data from the following input registers (0 to n)

Exception codes:

In case of error, only one data byte (the exception code) will be transmitted

Exception codes:

0x02: Permissible range of the input registers 0xD000 to 0xD026 exceeded
0x03: Maximum telegram length for answer (23 bytes) exceeded
i.e. either more than 6 input registers or 0 input registers were requested.

1.3.6.3 Write Single Register Addressed By Serial No.

Command code: 0x46

This command is used to describe the content of *one* holding register.

In contrast to "MODBUS over Serial Line Specification & Implementation Guide V1.0", the fan also responds to a broadcast command (Modbus address = 0), provided that no broadcast identifier (0) is used in the serial number. Because no two different fans can have the same serial number, it is not possible for more than one fan to answer.

Command from master:

10 data bytes are transmitted:
- Serial No. byte 1
- Serial No. byte 2
- Serial No. byte 3
- Serial No. byte 4
- Serial No. byte 5
- Serial No. byte 6
- Holding register MSB address
- Holding register LSB address
- MSB data to be written
- LSB data to be written

The description of the holding registers can be found at a later point.

Response from fan:

10 data bytes are transmitted:
- Serial No. byte 1
- Serial No. byte 2
- Serial No. byte 3
- Serial No. byte 4
- Serial No. byte 5
- Serial No. byte 6
- Holding register MSB address
- Holding register LSB address
- MSB data to be written
- LSB data to be written

The command from the master is repeated in the answer

Exception codes:

In case of error, only one data byte (the exception code) will be transmitted

Exception codes:

0x02: Permissible range of the holding registers 0xD000 to 0xD37F exceeded
0x04: There is no write permission in this authorisation level (password).

1.3.6.4 Write Multiple Registers Addressed By Serial No.

Command code: 0x50

This command is used to write the content of *several* holding registers.

Command from master:

The following data bytes are transmitted:
- Serial No. byte 1
- Serial No. byte 2
- Serial No. byte 3
- Serial No. byte 4
- Serial No. byte 5
- Serial No. byte 6
- Holding register MSB address
- Holding register LSB address
- Number of MSB addresses to be written
- Number of LSB addresses to be written
- Byte count (number of addresses to be written * 2)
- Data to be written in 1st MSB holding register
- Data to be written in 1st LSB holding register

Optional:
- Data to be written to the following holding registers (0 to n)

The description of the holding registers can be found at a later point.

Response from fan:

10 data bytes are transmitted:
- Serial No. byte 1
- Serial No. byte 2
- Serial No. byte 3
- Serial No. byte 4
- Serial No. byte 5
- Serial No. byte 6
- Holding register MSB address
- Holding register LSB address
- Number of MSB addresses to be written
- Number of LSB addresses to be written

Exception codes:

In case of error, only one data byte (the exception code) will be transmitted

Exception codes:
0x02: Permissible range of the holding registers 0xD000 to 0xD37F exceeded
0x03: - The number of maximum possible registers has been exceeded
i.e. either more than 123 holding register data or 0 holding register data were defined.
- Byte count ≠ 2 * number of registers
- Number of data bytes ≠ byte count
0x04: The holding register cannot be written due to a defect in the electronics
- There is no write permission in this authorisation level (password).

1.3.6.5 Applications

Initialising an installation

In factory condition, all fans have the same address 1. To communicate with each fan individually, when a system composed of multiple fans is set up, this address must be changed so that each fan has a different address.

Conventional method (command 0x06 Write Single Register):
1. Switch on the first fan (all others remain switched off)
2. Change the address via command 0x06 Write Single Register
3. Switch on the next fan
4. Repeat steps 2 and 3 until all fans have an individual address.

Procedure for known serial numbers (e.g. printed on type plate)
1. Switch on all fans
2. Change the addresses using command 0x46 Write Single Register Addressed By Serial No.

Procedure with unknown serial numbers:
1. Switch on all fans
2. Identification of the serial numbers using command 0x43 Read Holding Register Addressed By Serial No. or 0x44 Read Input Register Addressed By Serial No. in broadcast address with collision detection
3. If a serial number has been identified:
   Change the Modbus address via command 0x46 Write Single Register Addressed By Serial No.

The advantage over the conventional procedure is that all the fans can be switched on at once when initialising an installation. Despite the same address 1, each fan can be addressed individually. Each fan can be assigned an individual address by the master device (e.g. PC) without switching actions being needed on the installation.

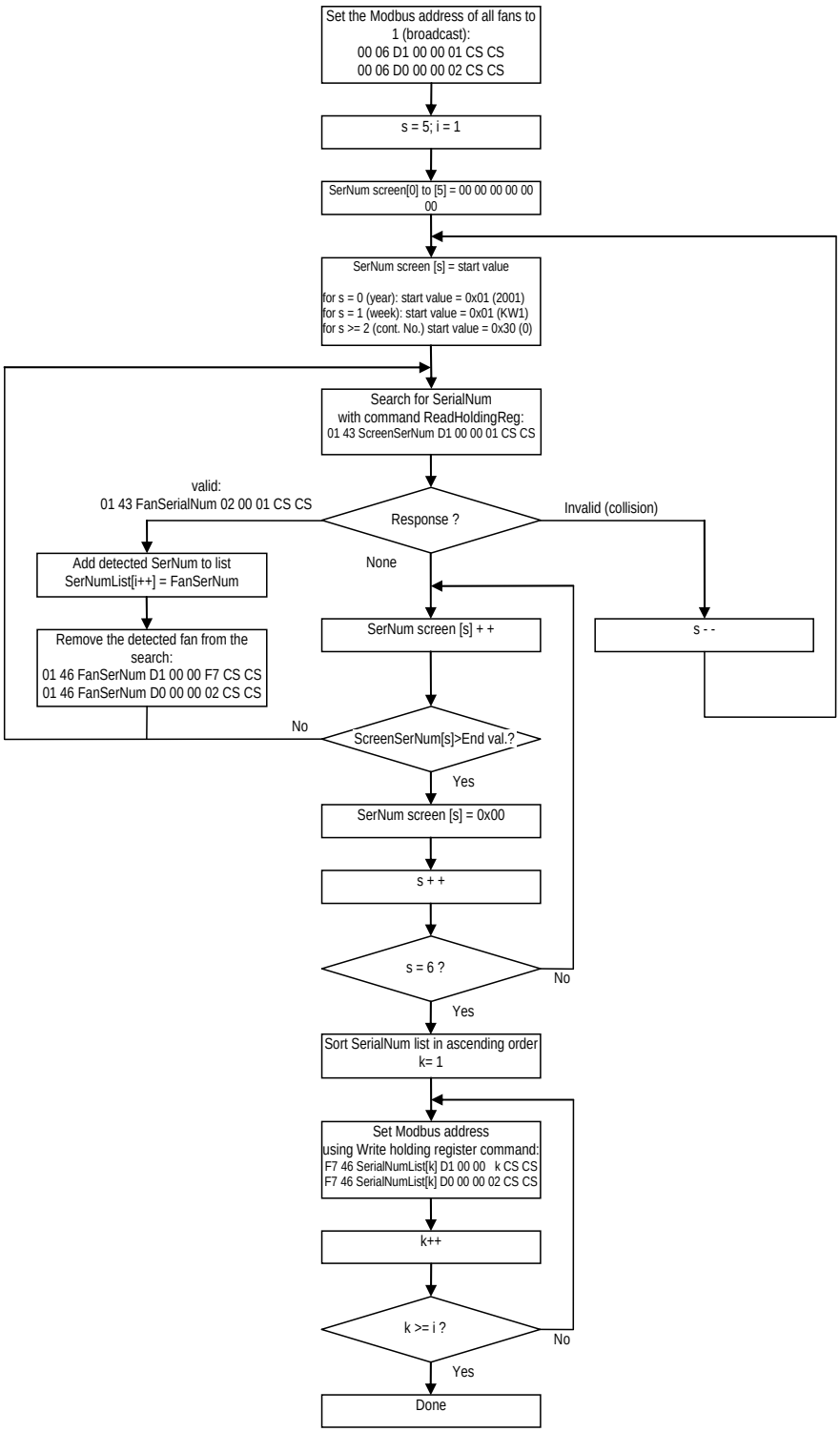
Implementation in the master device (e.g. PC):

For known serial numbers, we recommend the following procedure for implementation in a master device:

1. The master device is furnished with all serial numbers via manual entry

2. Each fan is assigned a Modbus address via the serial number.
   For this purpose, 2 broadcast commands are required per fan:
   a. Set the holding register fan address (D001):
      00 46 SNr SNr SNr SNr SNr SNr D1 00 00 Adr CS CS
      (CS = CRC checksum)
   b. Set the holding register reset (D000) to 0x02 Adopt parameters:
      00 46 SNr SNr SNr SNr SNr SNr D0 00 00 02 CS CS
      SNr = determined serial number of the fan; Adr = assigned Modbus address

For unknown serial numbers, we recommend the following procedure for implementation in a master device:



Function:

When the system is being set up, the fans have to be arranged in ascending order by serial number. This is the prerequisite for ensuring that the automatically assigned Modbus addresses can easily be assigned to the fans of the system.

At the beginning, all fans are set to Modbus address 0x01.
For this purpose, 2 broadcast commands are required:

- Set the holding register fan address (D001) to 0x01: 00 06 D1 00 00 01 CS CS
  (CS = CRC checksum)
- Set the holding register reset (D000) to 0x02 Adopt parameters: 00 06 D0 00 00 02 CS CS

All of the serial numbers of the system are determined in a loop:
The screen for the serial number address is set initially to broadcast address
(00 00 00 00 00 00). The last byte is set to the start value of 0x30 (0).
The SerNr screen is thus 00 00 00 00 00 30.
This screen searches for fans that have the value 0x30 (0) in the last position of the serial number.
You can use any command for reading a holding register or input register, e.g. Read holding register fan address: 01 43 00 00 00 00 00 30 D1 00 00 01 CS CS

There are multiple options for the response:

- There is a unique response:
  The fan answers in the address field with its serial number (SerNrVent).
  This serial number is stored in a list.
  Then, the fan is disabled for further queries by setting the Modbus address to 0xF7 (247).
  The same serial number then has to be queried again, as the possibility cannot be eliminated that another fan was also addressed, but did not answer because it already detected the beginning of the other fan's response due to run time differences.
- There is an invalid response due to overlapping responses from multiple fans:
  In this case, the screen for the serial number must be limited further by now also setting the second-last byte to the start value 0x30 (0). The next query is now answered only by fans that have the value 0x30 (0) in the last two digits of the serial number.
- There is no response:
  In this case, all serial numbers can be eliminated that have the value 0x30 in the last position. The last byte of the screen is increased to the value 0x31 (1). The next query is now answered by all fans that have the value 0x31 (1) in the last position of the serial number.

The loop is now continued until all serial numbers have been queried:

- If a valid response is given, the serial number of the fan is stored in the list.
- In case of an invalid response, the serial number range is restricted further by masking an additional byte, beginning with the start value. For the last 4 bytes, the start value is 0x30 (0). The first two bytes have the start value 0x01, as the year (2001) and calendar week (KW1) are encoded here.
- If there is no response, the respective position is increased by 1 until the end value is reached.
  For the last 4 bytes, the end value is 0x5A (Z). For the first byte (year), the end value is 0x63 (2099); for the second byte (KW), the end value is 0x35 (KW53).
  If the end value is reached, the respective position is again assigned an address via broadcast (0x00) and the next position is increased by 1. If the address setup for the last byte reaches the end value 0x5A (Z), all serial numbers have been checked. The query of serial numbers can thus be concluded.

Afterwards, all serial numbers found in the master are sorted in ascending order.
Each fan is assigned a Modbus address via the serial number.
For this purpose, 2 commands are required per fan:

- Set the holding register fan address (D001): F7 46 SNr SNr SNr SNr SNr SNr D1 00 00 Adr CS CS

ebm-papst Mulfingen GmbH & Co. KG | Bachmühle 2 · 74673 Mulfingen, Germany · Phone: +49-7938-81-0 · Fax: +49-7938-81-110 · www.ebmpapst.com · info1@de.ebmpapst.com

DocNo.: 634505DocNo.: 446144DocNo.:358982DocNo.:322523DocNo.:309753DocNo.:303997DocNo.:276241DocNo.:256078DocNo.:196392 · Template: 2 dated 2003-10-06 · File: ext000757333.doc · Last printed 29.10.2012 11:40:00 · Page 24 of 84

- Set the holding register reset (D000) to 0x02 Adopt parameters:
  F7 46 SNr SNr SNr SNr SNr SNr D0 00 00 02 CS CS
  SNr = determined serial number of the fan; Adr = assigned Modbus address

Extension of the address space

Up to 247 fans can be addressed with standard Modbus commands.
Commands with addressing via serial number offer the possibility of addressing an unlimited number of ebm-papst fans using the Modbus interface. In this case, the address of the fan is not just made up on one byte, but rather of 7 bytes (Modbus address + 6 byte serial number). This address is defined in the factory and cannot be changed.

For the purpose of addressing, the serial number must be known in the master device (e.g. PC) as a scan of all serial numbers is not possible due to the large numbers involved. To identify the serial number, see above (Initialising an installation).

Only commands with addressing via the serial number can be used with an extended address space. If using standard commands, conflicts are inevitable as several fans will have the same Modbus address.

Identifying an unknown Modbus address

- Only one fan may be connected to the bus
- Command 0x43 Read Holding Register Addressed By Serial No. or 0x44 Read Input Register Addressed By Serial No. in broadcast address:
  The Modbus address and serial number identification are specified using 0x00
- The connected fan responds with its Modbus address and serial number

1.3.7 Other commands

All other commands are not supported.
A command is always answered with exception code 0x01.

2 Holding Register

2.1 Overview

The holding registers are stored in the RAM and in the EEPROM of the fan.
Depending on the range concerned, access times (and thus also response times) may differ

The following ranges are defined:

Address	Area	Typ. read access time	Typ. write access time
D000 to D0FF	RAM	1µs / byte	1µs / byte
D100 to D17F	EEPROM internal	2µs / byte	4ms / byte
D180 to D37F	EEPROM external	500µs / byte	6.5ms / byte

The following list gives an overview of all parameters.
Apart from the Modbus address and the designation, it shows which authorisation level is required to write a parameter, and the address of the memory space for default setting and customer setting (if applicable).

The function of the parameters is described in the following chapters

Modbus Address	Designation	Write ebm-papst	Write Customer	Write End customer	Default Address	Cust. setting Address
D000	Reset	X	X	X	-	-
D001	Default set value	X	X	X	-	-
D002 D003 D004	Password	X	X	X	-	-
D005	Control default setting	X	X *)	-	-	-
D006	Control customer setting	X	X	X *)	-	-
D007	Reserved	X	-	-	-	-
D008	Reserved	X	-	-	-	-
D009	Operating hours counter	X	-	-	-	-
D00A	Operating minutes counter	X	-	-	-	-
D00B	Reserved	-	-	-	-	-
D00C - D0FF	Vacant	-	-	-	-	-
D100	Fan address	X	X	X	D280	D200
D101	Source set value	X	X	X	D281	D201
D102	Preferred running direction	X	X	X	D282	D202
D103	Store set value	X	X	X	D283	D203
D104	Parameter set source	X	X	X	D284	D204
D105	Internal parameter set	X	X	X	D285	D205
D106	Operation mode (parameter set 1)	X	X	X	D286	D206
D107	Operation mode (parameter set 2)	X	X	X	D287	D207
D108	Control function (parameter set 1)	X	X	X	D288	D208
D109	Control function (parameter set 2)	X	X	X	D289	D209
D10A	P factor (parameter set 1)	X	X	X	D28A	D20A
D10B	P factor (parameter set 2)	X	X	X	D28B	D20B

Modbus Address	Designation	Write ebm-papst	Write Customer	Write End customer	Default Address	Cust. setting Address
D10C	I factor (parameter set 1)	X	X	X	D28C	D20C
D10D	I factor (parameter set 2)	X	X	X	D28D	D20D
D10E	Max. modulation level (parameter set 1)	X	X	-	D28E	D20E
D10F	Max. modulation level (parameter set 2)	X	X	-	D28F	D20F
D110	Min. modulation level (parameter set 1)	X	X	X	D290	D210
D111	Min. modulation level (parameter set 2)	X	X	X	D291	D211
D112	Enable motor stop (parameter set 1)	X	X	X	D292	D212
D113	Enable motor stop (parameter set 2)	X	X	X	D293	D213
D114	Set value (parameter set 1)	X	X	X	D294	D214
D115	Set value (parameter set 2)	X	X	X	D295	D215
D116	Starting modulation level	X	-	-	D296	-
D117	Max. permissible modulation level	X	-	-	D297	-
D118	Min. permissible modulation level	X	-	-	D298	-
D119	Max. speed	X	X	-	D299	D219
D11A	Max. permissible speed	X	-	-	D29A	-
D11B	Reserved	X	-	-	D29B	-
D11C	Reserved	X	-	-	D29C	-
D11D	Vacant	X	X	X	D29D	D21D
D11E	Reserved	X	-	-	D29E	-
D11F	Ramp-up curve	X	X	X	D29F	D21F
D120	Ramp-down curve	X	X	X	D2A0	D220
D121	Reserved	X	-	-	D2A1	-
D122	Reserved	X	-	-	D2A2	-
D123	Reserved	X	-	-	D2A3	-
D124	Reserved	X	-	-	D2A4	-
D125	Reserved	X	-	-	D2A5	-
D126	Reserved	X	-	-	D2A6	-
D127	Reserved	X	-	-	D2A7	-
D128	Limit speed	X	-	-	D2A8	-
D129	Vacant	X	X	X	D2A9	D229
D12A	Potentiometer characteristic point 1 X co-ordinate (par. 1)	X	X	X	D2AA	D22A
D12B	Potentiometer characteristic point 1 Y co-ordinate (par. 1)	X	X	X	D2AB	D22B
D12C	Potentiometer characteristic point 2 X co-ordinate (par. 1)	X	X	X	D2AC	D22C
D12D	Potentiometer characteristic point 2 Y co-ordinate (par. 1)	X	X	X	D2AD	D22D
D12E	Source for controller function	X	X	X	D2AE	D22E
D12F	Control limitation	X	-	-	D2AF	-
D130	Output function 0 to 10V / speed monitoring	X	X	-	D2B0	D230
D131	Reserved	X	-	-	D2B1	-
D132	Reserved	X	-	-	D2B2	-
D133	Reserved	X	-	-	D2B3	-
D134	Reserved	X	-	-	D2B4	-
D135	Max. permitted power	X	-	-	D2B5	-
D136	Max. power for derating end	X	X	-	D2B6	D236
D137	Module temperature power derating start	X	X	-	D2B7	D237
D138	Module temperature power derating end	X	X	-	D2B8	D238
D139	Reserved	X	-	-	D2B9	-
D13A	Reserved	X	-	-	D2BA	-

Modbus Address	Designation	Write ebm-papst	Write Customer	Write End customer	Default Address	Cust. setting Address
D13B	Max. coil current	X	-	-	D2BB	-
D13C	Potentiometer characteristic point 1 X co-ordinate (par. 2)	X	X	X	D2BC	D23C
D13D	Potentiometer characteristic point 1 Y co-ordinate (par. 2)	X	X	X	D2BD	D23D
D13E	Potentiometer characteristic point 2 X co-ordinate (par. 2)	X	X	X	D2BE	D23E
D13F	Potentiometer characteristic point 2 Y co-ordinate (par. 2)	X	X	X	D2BF	D23F
D140	0 to 10V output characteristic - point 1 X	X	X	X	D2C0	D240
D141	0 to 10V output characteristic - point 1 Y	X	X	X	D2C1	D241
D142	0 to 10V output characteristic - point 2 X	X	X	X	D2C2	D242
D143	0 to 10V output characteristic - point 2 Y	X	X	X	D2C3	D243
D144	Reserved	X	X	-	D2C4	D244
D145	Limit speed for running monitor	X	X	-	D2C5	D245
D146	Reserved	X	-	-	D2C6	-
D147	Actual sensor value source	X	X	X	D2C7	D247
D148	Running direction source	X	X	X	D2C8	D248
D149	Transmission rate	X	X	-	D2C9	D249
D14A	Parity configuration	X	X	-	D2CA	D24A
D14B	Reserved	X	-	-	D2CB	-
D14C	Reserved	X	-	-	D2CC	-
D14D	Motor temperature power derating start	X	X	-	D2CD	D24D
D14E	Motor temperature power derating end	X	X	-	D2CE	D24E
D14F	Reserved	X	-	-	D2CF	-
D150	Sheding function	X	X	-	D2D0	D250
D151	Max. starting modulation level	X	-	-	D2D1	-
D152	Number of startup attempts	X	X	-	D2D2	D252
D153	Relay drop-out delay	X	X	X	D2D3	D253
D154	Reserved	X	-	-	D2D4	-
D155	Max. power	X	X	-	D2D5	D255
D156 - D15A	Vacant	X	X	X	D2D6 - D2DA	D256 - D25A
D15B	Emergency operation running direction	X	X	-	D2DB	D25B
D15C	Emergency operation function on/off	X	X	-	D2DC	D25C
D15D	Emergency operation set value	X	X	-	D2DD	D25D
D15E	Emergency operation time lag	X	X	-	D2DE	D25E
D15F	Potentiometer characteristic, limit value for cable break	X	X	-	D2DF	D25F
D160 D161	Min. sensor value	X	X	X	D2E0 D2E1	D260 D261
D162 D163	Max. sensor value	X	X	X	D2E2 D2E3	D262 D263
D164 - D169	Sensor unit	X	X	X	D2E4 - D2E9	D264 - D269
D16A - D16F	Vacant	X	X	X	D2EA - D2EF	D26A - D26F
D170 - D17F	Customer data	X	X	-	D2F0 - D2FF	D270 - D27F
D180	Operating hours counter (backup)	X	-	-	-	-

Modbus Address	Designation	Write ebm-papst	Write Customer	Write End customer	Default Address	Cust. setting Address
D181	Reserved	X	-	-	-	-
D182	Error indicator	X	-	-	-	-
D183	Vacant	X	-	-	-	-
D184	1st error	X	-	-	-	-
D185	1st error timing	X	-	-	-	-
D186 - D19F	Error history Error history timing	X	-	-	-	-
D1A0	Reference value of DC-link voltage	X	-	-	-	-
D1A1	Reference value of DC-link current	X	-	-	-	-
D1A2 D1A3	Fan serial number	X	-	-	-	-
D1A4	Fan production date	X	-	-	-	-
D1A5 - D1AA	Fan type	X	-	-	-	-
D1AB - D1AC	Reserved	X	-	-	-	-
D1AD - D1AE	Reserved	X	-	-	-	-
D1AF	Reserved	X	-	-	-	-
D1B0	Reserved	X	-	-	-	-
D1B1	Reserved	X	-	-	-	-
D1B2	Reserved	X	-	-	-	-
D1B3	Reserved	X	-	-	-	-
D1B4 - D1B5	Reserved	X	-	-	-	-
D1B6	Reserved	X	-	-	-	-
D1B7	Reserved	X	-	-	-	-
D1B8 - D1BF	Reserved	X	-	-	-	-
D1C0 - D1C1	Reserved	X	-	-	-	-
D1C2 - D1C9	Reserved	X	-	-	-	-
D1CA - D1D1	Reserved	X	-	-	-	-
D1D2 - D1D9	Reserved	X	-	-	-	-
D1DA - D1E1	Reserved	X	-	-	-	-
D1E2	Reserved	X	-	-	-	-
D1E3 - D1E4	Reserved	X	-	-	-	-
D1E5 - D1E6	Reserved	X	-	-	-	-
D1E7	Reserved	X	-	-	-	-
D1E8	Reserved	X	-	-	-	-
D1E9	Reserved	X	-	-	-	-
D1EA - D1F8	Vacant	X	-	-	-	-
D1F9 -	Reserved	X	-	-	-	-

Modbus Address	Designation	Write ebm-papst	Write Customer	Write End customer	Default Address	Cust. setting Address
D1FA						
D1FB - D1FC	Reserved	X	-	-	-	-
D1FD	Reserved	X	-	-	-	-
D1FE	Reserved	X	-	-	-	-
D1FF	Reserved	X	-	-	-	-
D300 - D30F	Reserved	X	-	-	-	-
D310 - D31F	Reserved	X	-	-	-	-
D320 - D32F	Reserved	X	-	-	-	-
D330 - D33F	Reserved	X	-	-	-	-
D340	Reserved	X	-	-	-	-
D341 - D35F	Vacant	X	-	-	-	-
D360 - D37F	Reserved	X	-	-	-	-

*) only in part

Encoding of parameters:

Unless specified otherwise, parameters are encoded in "big endian" format, i.e. the byte with the highest-value bits comes first. This is particularly true for parameters that include multiple holding registers.

2.2 Reset

Address : D000
Write authorisation : ebm-papst, customer, end customer

Encoding:

MSB	0	0	0	0	0	0	0	0
LSB	0	0	0	0	Reset	Error	Parameter	AWS Rst

A bit will trigger the following action in the fan when it is set:

Reset : Software reset (includes "Reset error" and "Adopt parameters")
The software is launched at the start of booting
Error : Errors are reset
Parameter : All parameters are copied from the EEPROM into the RAM
This bit must be set to validate revised parameters
Reset AWS : User software reset
(includes "Reset error" and "Adopt parameters")
The software is launched at the start of user software

After the action has been executed, the bit will automatically be reset by the fan.

When reset (bit 3), the program will be relaunched at the start of booting. Commands from the master will then be neither answered nor executed for a few seconds.
Reset (bit 3) is only recommended if new user software is to be loaded using a bootloader (because communication is interrupted).
If the software should only be relaunched, user software reset (bit 0) is recommended. Communication is then not interrupted.

2.3 Default set value

Address : D001
Write authorisation : ebm-papst, customer, end customer

The parameter "Default set value" is used in Modbus to specify a set value for each control mode.
The condition for this is that the "Source set value" RS485 (1) is specified (see 2.10 Source set value).
Otherwise, the parameter will have no function.

If the "Store" function is activated in the parameter "Store set value", the value in the set value parameter (EEPROM) will be stored every time the default set value is write-accessed (see 2.23 Set value (EEPROM))

The external input "parameter set 1/2" and the parameter "Internal parameter set" are used to select whether the value in "Set value (parameter set 1)" or in "Set value (parameter set 2)" is stored (see 2.15 Internal parameter set).

Following a reset, the motor will again run with this value, provided the parameter set selection has not been changed. (see 2.13 Store set value)

Important!
If the "Store set value" function is activated and the parameter set is changed (from 1 to 2 or vice versa), the default set value will automatically be changed to the corresponding parameter "Set value (parameter set 1)" or "Set value (parameter set 2)"!

Encoding:

Note: The 4 LSBits are of no relevance for the set value and will always be assumed to be 0.

a) in closed loop speed control

The default set value denotes a speed:

$$\text{Default set value [rpm]} = \frac{\text{Data bytes}}{64000} \cdot \text{nMax [rpm]}$$

nMax [rpm] to maximum speed in revolutions per minute (see 2.27 Maximal speed)

The value zero means motor standstill

b) in open loop PWM control

The default set value denotes a modulation level:

$$\text{Default set value [\%]} = \frac{\text{Data bytes}}{65536} \cdot 100\%$$

The value zero means motor standstill

c) in closed loop sensor control

The default set value denotes a sensor variable:

A sensor that converts the control variable into a voltage of 0 to 10V or a current of 4 to 20mA must be connected to the fan.

A set value can be entered for the output voltage or the output current of the sensor used.
The default set value for the control variable is then made up of the set value parameter and the Rg(U/I) characteristic of the sensor used.

Rg (U) = Control variable, voltage-dependent

$$\text{Default set value [V]} = \frac{\text{Data bytes}}{65536} \cdot 10V$$

$$\text{Default set value [unit (Rg)]} = Rg\,(\text{set value [V]})$$

$$= Rg\left(\frac{\text{Data bytes}}{65536} \cdot 10V\right)$$

or

Rg (U) = Control variable, current-dependent

$$\text{Default set value [mA]} = \frac{\text{Data bytes}}{65536} \cdot 16mA + 4mA$$

$$\text{Default set value [unit (Rg)]} = Rg\,(\text{set value [V]})$$

$$= Rg\left(\frac{\text{Data bytes}}{65536} \cdot 16mA + 4mA\right)$$

2.4 Password

Address : D002 - D004
Write authorisation : ebm-papst, customer, end customer

Encoding:

$$\text{Password} = \text{Data bytes}$$

In order to prevent unauthorised writing of certain parameters, these are only written if the correct password for the necessary authorisation is entered at this point.
If a user forgets to reset the password, this will automatically be reset to 0x000000000000 after 4 minutes of inactivity.

When the "Password" parameter is read, the value 0 is always output to prevent a user with a lower authorisation level gaining access to a password to a higher level.

2.5 Control default setting

Address : D005
Write authorisation : ebm-papst, customer (in part)

Encoding:

MSB	0	0	0	0	0	0	0	0
LSB	0	0	0	0	0	Error	D -> W	W -> D

Setting the bit D -> W causes all parameters in the data range (D100 to D17F) to be copied to the default setting (range D280 to D2FF).
The authorisation level "ebm-papst" is needed to set this bit!

Setting the bit W -> D causes all parameters in the default setting (D280 to D2FF) to be copied to the data range (range D100 to D17F).
The authorisation level "customer" is sufficient to set this bit.

Once the copying process is complete, the bit will automatically be reset by the fan.

If an error is detected during the copying process, the "Error" bit will be set and the copy process cancelled

2.6 Control customer setting

Address : D006
Write authorisation : ebm-papst, customer, end customer (in part)

Encoding:

MSB	0	0	0	0	0	0	0	0
LSB	0	0	0	0	0	Error	D -> K	K -> D

Setting the bit D -> K causes all parameters in the data range (D100 to D17F) for which the customer has write authorisation to be copied to the customer setting (range D200 to D27F). The minimum authorisation level needed to set this bit is "customer"!

Setting the bit K -> D causes all parameters in the customer setting (range D200 to D27F) for which the customer has write authorisation to be copied to the data range (D100 to D17F).
The authorisation level "end customer" is sufficient to set this bit.

Once the copying process is complete, the bit will automatically be reset by the fan.

If an error is detected during the copying process, the "Error" bit will be set and the copy process cancelled

2.7 Operating hours counter

Address : D009
Write authorisation : ebm-papst

Encoding:

Operating time [h] = Data bytes

After every hour the fan runs, the operating hours counter increases by 1.
The holding register D180 "operating hours counter backup" is updated simultaneously whenever a change is made (see 2.48).

With 16 bits, a maximum of 65,535 hours (approx. 7.5 years) can be counted.
If the operating hours counter overshoots this figure, it will no longer be written, i.e. it will stay at 65,535.

2.8 Operating minutes counter

Address : D00A
Write authorisation : ebm-papst

Encoding:

Operating minutes [h] = Data bytes

After every full minute the fan runs, the operating minutes counter increases by 1.

2.9 Fan address

Address : D100
Write authorisation : ebm-papst, customer, end customer

Encoding:

Fan address = Data bytes (LSB)

The MSB is of no relevance!

Permitted value range: 1 to 247

2.10 Source set value

Address : D101
Write authorisation : ebm-papst, customer, end customer

Encoding:

This parameter specifies the source from which the set value is taken:

Value	Source set value
0	Analogue input 0 to 10V
1	RS485 (default set value parameter D001)

The MSB is of no relevance!

Permitted value range: 0 to 1

2.11 Running direction source

Address : D148
Write authorisation : ebm-papst, customer, end customer

Encoding:

This parameter specifies which sources specify the running direction.

Value	Running direction source
0	Digital input Din2 and parameter "Preferred running direction"
1	Only parameter "Preferred running direction"
2	Digital input Din3 and parameter "Preferred running direction"

The MSB is of no relevance!

Permitted value range: 0 to 2

If only the parameter "Preferred running direction (1)" is selected as the running direction source, the actual running direction corresponds to that specified in the parameter "Preferred running direction".

If a digital input is selected as the running direction source, the actual running direction corresponds to that specified in the parameter "Preferred running direction" if the corresponding digital input is open.
If the digital input is bridged to 0V, the actual running direction is opposite to that specified in the Preferred running direction parameter.

The following table provides an overview of the actual running direction:

Running direction source	0	0	0	0	1	1	2	2	2	2
Preferred running direction	0	0	1	1	0	1	0	0	1	1
Digital input Din2	0V	open	0V	open	X	X	X	X	X	X
Digital input Din3	X	X	X	X	X	X	0V	open	0V	open
Running direction	clockwise	counter-clockwise	counter-clockwise	clockwise	counter-clockwise	clockwise	clockwise	counter-clockwise	counter-clockwise	clockwise

If the emergency operation function is activated and a break in the Modbus connection is detected, the direction of rotation is set to the value of "Emergency operation running direction" (see 2.44). The parameter Running direction source is not relevant then.

2.12 Preferred running direction

Address : D102
Write authorisation : ebm-papst, customer, end customer

Encoding:

This parameter specifies the preferred running direction.

Value	Preferred running direction
0	counter-clockwise
1	clockwise

The MSB is of no relevance!

Permitted value range: 0 to 1

The new value immediately becomes applicable when this parameter is written (no reset necessary).

In addition to this parameter, "Running direction source" and digital inputs Din2 and Din3 affect the actual running direction.
For additional information about the actual running direction, see 2.11Running direction source

If the emergency operation function is activated and a break in the Modbus connection is detected, the direction of rotation is set to the value of "Emergency operation running direction" (see 2.44). The parameter Preferred running direction is not relevant then.

2.13 Store set value

Address : D103
Write authorisation : ebm-papst, customer, end customer

Encoding:

This parameter specifies whether or not an incoming default set value (D001) is also stored in the EEPROM under set value (EEPROM) (D114 / D115).

Value	Function
0	Set value is not stored The fan is stationary after a reset
1	Set value is stored in EEPROM After reset, the fan will run with the stored set value

The MSB is of no relevance!

Permitted value range: 0 to 1

2.14 Parameter set source

Address : D104
Write authorisation : ebm-papst, customer, end customer

Encoding:

This parameter specifies whether the external input or the internal parameter is to be used to switch between parameter set 1 and parameter set 2.

Value	Parameter set switch by...
0	Digital input Din2
1	Parameter "Internal parameter set" (see 2.15 Internal parameter set)
2	Digital input Din3

The MSB is of no relevance!

Permitted value range: 0 to 2

The following table provides an overview of the selected parameter set:

Parameter set source	0	0	1	1	2	2
Internal parameter set	X	X	0	1	X	X
Digital input Din2	0V	open	X	X	X	X
Digital input Din3	X	X	X	X	0V	open
Selected parameter set	2	1	1	2	2	1

2.15 Internal parameter set

Address : D105
Write authorisation : ebm-papst, customer, end customer

Encoding:

This parameter specifies whether parameter set 1 or parameter set 2 is used.
The specification in this parameter is applicable only if "Parameter set source" has the value "internal" (1) (see 2.14 Parameter set source).

Value	Parameter set
0	Parameter set 1
1	Parameter set 2

The MSB is of no relevance!

Permitted value range: 0 to 1

The new value immediately becomes applicable when this parameter is written (no reset necessary).

2.16 Control mode

Address of parameter set 1 : D106
Address of parameter set 2 : D107
Write authorisation : ebm-papst, customer, end customer

The external input "parameter set" and the parameter "Internal parameter set" are used to select whether the value in "Operation mode (parameter set 1)" or the value in "Operation mode (parameter set 2)" is applicable (see 2.15 Internal parameter set).

Encoding:

Value	Control mode
0	Closed-loop speed control
1	Sensor control
2	Control system

The MSB is of no relevance!

Permitted value range: 0 to 2

2.17 Source of control function

Address : D12E
Write authorisation : ebm-papst, customer, end customer

Encoding:

This parameter specifies whether the control function is determined by the external input or by the internal parameter.

Value	Control function determined by ...
0	Digital input Din3
1	"Control function" parameter (see 2.18 Control function)
2	Digital input Din2

The MSB is of no relevance!

Permitted value range: 0 to 2

The following table provides an overview of the selected control function:

Source for controller function	0	0	1	1	2	2
Control function internal	X	X	0	1	X	X
Digital input Din2	0V	open	X	X	X	X
Digital input Din3	X	X	X	X	0V	open
Selected control function	Negative	Positive	Positive	Negative	Negative	Positive

2.18 Control function

Address of parameter set 1 : D108
Address of parameter set 2 : D109
Write authorisation : ebm-papst, customer, end customer

The external input "parameter set" and the parameter "Internal parameter set" are used to select whether the value in "Control function (parameter set 1)" or the value in "Control function (parameter set 2)" is applicable (see 2.15 Internal parameter set).

The specification in these parameters is applicable only if the parameter "Source of control function" has the value "internal" (1) (see 2.17 Source of control function). Otherwise, both parameters will have no function.

The control function specifies how the control variable is defined from the set value and the actual value

Encoding:

Value	Control function
0	Positive: Control variable = Actual value - Set value
1	Negative: Control variable = Set value - Actual value

A positive control function causes the modulation level of the fan to decrease as the actual value is increased.

A negative control function causes the modulation level of the fan to increase as the actual value is increased.

Remarks:

For sensor control with temperature sensor, a positive control function equates to "heat" and a negative control function equates to "cool".

In "closed loop speed control" mode, only a positive control function is logical. For this reason, the parameter "Control function" has no function in the "Closed-loop speed control" mode and is always assumed to be positive.

2.19 Control parameters

2 control parameters are provided for closed loop speed control and closed loop sensor control:

- P factor kp
- I factor ki

Address of P factor (parameter set 1)	: D10A
Address of P factor (parameter set 2)	: D10B
Address of I factor (parameter set 1)	: D10C
Address of I factor (parameter set 2)	: D10D
Write authorisation	: ebm-papst, customer, end customer

The external input "parameter set" and the parameter "Internal parameter set" are used to select whether the values in "P factor / I factor (parameter set 1)" or the values in "P factor / I factor (parameter set 2)" are applicable (see 2.15 Internal parameter set).

Encoding:

Each control parameter consists of 2 bytes.

a) P factor

$$P\text{-}factor = \frac{Data\ bytes}{256} \cdot 100\%$$

This means that values can be set for the P factor between 0 and 25,600%, in steps of 0.39%

b) I factor

$$I\text{-}factor = \frac{Data\ bytes}{65536} \cdot 100\%$$

This means that values can be set for the I factor between 0 and 100%, in steps of 0.00153%

Remarks concerning control behaviour:

The fan control computes a new modulation level according to the equation below after each scanning cycle time T_a = 50ms:

$$a(n) = \frac{kp}{100\%} \cdot xd(n) + \sum_{i=0}^{n} \frac{ki}{100\%} \cdot xd(i)$$

a(n) ..... Modulation level at time n; standardised to range 0 to 65536, i.e. 65536 -> 100%

xd(n) ..... Standardised control deviation at time n (set value – actual value)

kp ..... P-factor in %

ki ..... I-factor in %

The standardised control variable is calculated from the set value and the actual value.
Set value and actual value are also standardised:
- To maximum speed in closed loop speed control
- To minimum and maximum sensor values in closed loop sensor control

The standardised control equation stated above and the generally applicable control equation

$$A(n) = kr \cdot \left(Xd(n) + \frac{Ta}{Tn} \sum_{i=0}^{n} Xd(i) \right)$$

A(n) ..... Modulation level at time n (0 to 100%)

Xd(n) ..... Absolute control deviation at time n (set value – actual value)

kr ..... Proportional gain

Tn ..... Integral action time

Ta ..... Sampling time

... can be converted to each other with

$$kr = \frac{kp}{\text{Max reference value} - \text{Min reference value}} \quad \text{or} \quad kp = kr \cdot (\text{Max reference value} - \text{Min reference value})$$

$$Tn = \frac{kp}{ki} \cdot Ta \quad \text{or} \quad ki = \frac{Ta}{Tn} \cdot kr \cdot (\text{Max reference value} - \text{Min reference value})$$

with $Ta = 50ms$

In closed loop speed control:

Max reference value = 1.024 · nMax

Min reference value = 0

nMax ..... Maximum speed [rpm]

In closed loop sensor control:

Max reference value = Max sensor value

Min reference value = Min sensor value

Max sensor value ..... Sensor variable for U = 10V

Min sensor value ..... Sensor variable for U = 0V

2.20 Maximum modulation level

Address of parameter set 1 : D10E
Address of parameter set 2 : D10F
Write authorisation : ebm-papst, customer

The external input "parameter set" and the parameter "Internal parameter set" are used to select whether the value in " Maximum modulation level (parameter set 1)" or the value in " Maximum modulation level (parameter set 2)" is applicable (see 2.15 Internal parameter set).

Encoding:

$$\textit{Max. modulation level [\%]} = \frac{\textit{Data byte}}{256} \cdot 100\%$$

The MSB is of no relevance!

Limitation:
8% < Maximum modulation level < Maximum permissible modulation level
(see 2.25 Maximum permissible modulation level)

2.21 Minimum modulation level

Address of parameter set 1 : D110
Address of parameter set 2 : D111
Write authorisation : ebm-papst, customer, end customer

The external input "parameter set" and the parameter "Internal parameter set" are used to select whether the value in " Minimum modulation level (parameter set 1)" or the value in "Minimum modulation level (parameter set 2)" is applicable (see 2.15 Internal parameter set).

Encoding:

$$\textit{Min. modulation level [\%]} = \frac{\textit{Data byte}}{256} \cdot 100\%$$

The MSB is of no relevance!

Limitation:
Minimum modulation level > Minimum permissible modulation level
(see 2.26 Minimum permissible modulation level)

2.22 Enable motor stop

Address of parameter set 1 : D112
Address of parameter set 2 : D113
Write authorisation : ebm-papst, customer, end customer

The external input "parameter set" and the parameter "Internal parameter set" are used to select whether the value in "Enable motor stop (parameter set 1)" or the value in "Enable motor stop (parameter set 2)" is applicable (see 2.15 Internal parameter set).

Encoding:

Value	Motor stop
0	Motor runs continuously (even if set value = 0)
1	Motor stops if set value = 0

The MSB is of no relevance!

2.23 Set value (EEPROM)

Address of parameter set 1 : D114
Address of parameter set 2 : D115
Write authorisation : ebm-papst, customer, end customer

The specification in these parameters is only applicable if the parameter "Source set value" has the value "RS485" (1) (see 2.10 Source set value) and the function "Store set value" is activated (see 2.13 Store set value). Otherwise, both parameters will have no function.

If the parameter "Default set value" is changed, the corresponding parameter "Set value (parameter set 1)" or "Set value (parameter set 2)" is set to the same value provided the function "Store set value" is activated (see 2.3Default set value, 2.13Store set value).
The external input "parameter set" and the parameter "Internal parameter set" are used to select whether the value in "Set value (parameter set 1)" or the value in "Set value (parameter set 2)" is stored (see 2.15Internal parameter set).

As the default set value is stored in the volatile memory, it is necessary to store this value (as necessary) in the non-volatile memory. The parameter "Set value (EEPROM)" is provided to this end.

Encoding:

Note: The 4 LSBits are of no relevance for the set value and will always be assumed to be 0.

<u>a) in closed loop speed control</u>

The set value denotes a speed:

$$Set\ value\ [rpm] = \frac{Data\ bytes}{64000} \cdot nMax\ [rpm]$$

nMax [rpm] to maximum speed in revolutions per minute (see 2.27 Maximal speed)

The value zero means motor standstill

b) in open loop PWM control

The set value denotes a modulation level:

$$Set\ value\ [\%] = \frac{Data\ bytes}{65536} \cdot 100\%$$

The value zero means motor standstill

c) in closed loop sensor control

The set value denotes a sensor variable:

A sensor that converts the control variable into a voltage of 0 to 10V or a current of 4 to 20mA must be connected to the fan.

A set value can be entered for the output voltage or the output current of the sensor used.
The set value for the control variable is then made up of the set value parameter and the Rg(U/I) characteristic of the sensor used.

Rg (U) = Control variable, voltage-dependent

$$Set\ value\ [V] = \frac{Data\ bytes}{65536} \cdot 10V$$

$$Set\ value\ [unit\ (Rg)] = Rg\ (set\ value\ [V])$$

$$= Rg\ \left(\frac{Data\ bytes}{65536} \cdot 10V\right)$$

or

Rg (U) = Control variable, current-dependent

$$Set\ value\ [mA] = \frac{Data\ bytes}{65536} \cdot 16mA + 4mA$$

$$Set\ value\ [unit\ (Rg)] = Rg\ (set\ value\ [V])$$

$$= Rg\ \left(\frac{Data\ bytes}{65536} \cdot 16mA + 4mA\right)$$

2.24 Starting modulation level

Address : D116
Write authorisation : ebm-papst

The starting modulation level specifies the PWM with which the motor is started after a start command.

Encoding:

$$\text{Starting modulation level [\%]} = \frac{\text{Data byte}}{256} \cdot 100\%$$

The MSB is of no relevance!

The motor always starts with the starting modulation level.
A high starting modulation level results in a high current when the motor starts!

2.25 Maximum permissible modulation level

Address : D117
Write authorisation : ebm-papst

This parameter defines the upper limit for the maximum modulation level (see 2.20 Maximum modulation level).

Encoding:

$$\text{Max. perm. modulation level [\%]} = \frac{\text{Data byte}}{256} \cdot 100\%$$

The MSB is of no relevance!

2.26 Minimum permissible modulation level

Address : D118
Write authorisation : ebm-papst

This parameter defines the lower limit for the minimum modulation level (see 2.21 Minimum modulation level).

Encoding:

$$\text{Min. perm. modulation level [\%]} = \frac{\text{Data byte}}{256} \cdot 100\%$$

The MSB is of no relevance!

If too small a value is selected for this parameter (especially values < 8%), the motor may shut down with the error "Locked motor"!

2.27 Maximal speed

Address : D119
Write authorisation : ebm-papst, customer

This parameter has two functions:

- All parameters with speed specifications (set values, actual values) are related to this value.
  The value 64,000 in these speed specifications is equivalent to the maximum speed value specified here.
- In control modes "closed loop sensor control" and "open loop PWM control", the speed is limited to the value specified here. (speed is in any case controlled in closed loop speed control).

Encoding:

Max. speed [rpm] = Data bytes

The maximum speed is made up of 2 bytes.

Limitation:
Maximum speed < Maximum permissible speed (see 2.28 Maximum permissible speed)

2.28 Maximum permissible speed

Address : D11A
Write authorisation : ebm-papst

This parameter defines the upper limit for the maximum speed (see 2.27 Maximal speed).

Encoding:

Max. perm. speed [rpm] = Data bytes

The maximum permissible speed is made up of 2 bytes.

2.29 Ramp-up curve / ramp-down curve

Address of ramp-up curve : D11F
Address of ramp-down curve : D120
Write authorisation : ebm-papst, customer, end customer

These parameters define the ramp time for a set value change of 256 steps
(i.e. the time for a change of the MSB set value by one step).
The parameter "Ramp-up curve" defines the time for a positive change of the set value.
The parameter "Ramp-down curve" defines the time for a negative change of the set value.

With closed loop speed control, 256 steps correspond to a change in speed of
Δn [rpm] = nMax [rpm] / 250

in open loop PWM control, 256 steps correspond to a change of PWM of
ΔPWM [%] = 0.39%

Encoding:

Time for set value change by 256 steps [ms] = Data byte · 10ms

The MSB is of no relevance!

The result of a change from the minimum value 0 to the maximum value 64,000 is a ramp-up time or ramp-down time of

Ramp time [s] = 250 · Time for set value change by 256 steps [ms] = Data byte · 2.5s

If the value is 0, the new set value is immediately effective, without any time lag.

2.30 Limit speed

Address : D128
Write authorisation : ebm-papst

This parameter defines the limit value for the safety function "limit speed". The closed loop speed control in this safety function ensures that the limit speed for the impeller is not exceeded.

Encoding:

Ceiling speed [rpm] = Data bytes

Comment:
The limit speed is not related to the maximum speed!

2.31 Potentiometer characteristic

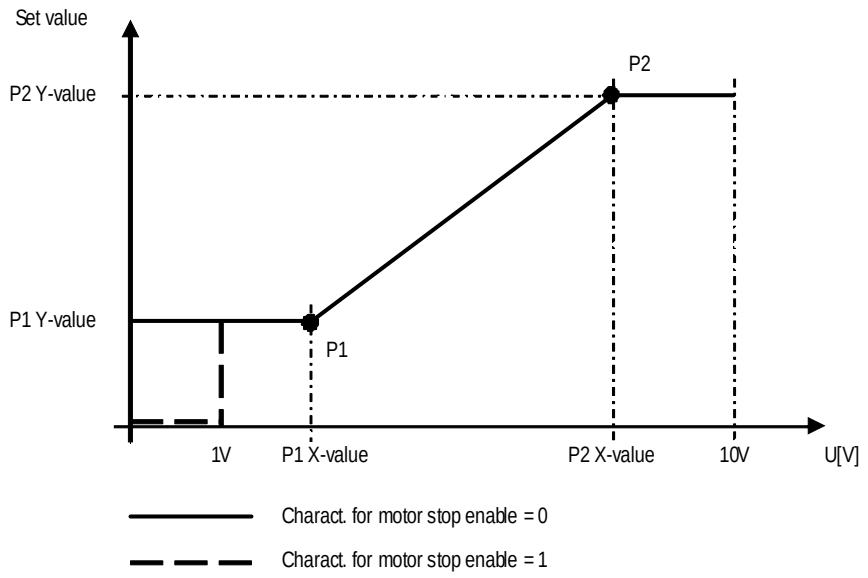
Address of point 1 X co-ordinate, parameter set 1 : D12A
Address of point 1 Y co-ordinate, parameter set 1 : D12B
Address of point 2 X co-ordinate, parameter set 1 : D12C
Address of point 2 Y co-ordinate, parameter set 1 : D12D

Address of point 1 X co-ordinate, parameter set 2 : D13C
Address of point 1 Y co-ordinate, parameter set 2 : D13D
Address of point 2 X co-ordinate, parameter set 2 : D13E
Address of point 2 Y co-ordinate, parameter set 2 : D13F

Write authorisation : ebm-papst, customer, end customer

2 different characteristics can be defined.
The external input "parameter set" and the parameter "Internal parameter set" are used to select whether the values in "parameter set 1" or the values in "parameter set 2" are applicable (see 2.15 Internal parameter set).

These parameters are used to assign a set value to the voltage at the control input.



Points P1 and P2 may be moved at random within the diagram.

Encoding:

The X co-ordinate defines a voltage value between 0 to 10V for the analogue input:

$$U\,[V]\,(Px) = \frac{Data\ bytes}{65536} \cdot 10V$$

The Y co-ordinate defines the associated set value for this point. Depending on the control mode concerned, this may be a speed (closed loop speed control), a sensor variable (closed loop sensor control) or a modulation level (open loop PWM control).

<u>a) closed loop speed control:</u>

$$Set\ value\ [rpm] = \frac{Data\ bytes}{64000} \cdot nMax\ [rpm]$$

nMax [rpm] to maximum speed in revolutions per minute (see 2.27 Maximal speed)

<u>b) in open loop PWM control</u>

$$Set\ value\ [\%] = \frac{Data\ bytes}{65536} \cdot 100\%$$

c) in closed loop sensor control

The set value for the control variable is then made up of the set value parameter and the Rg(U/I) characteristic of the sensor used.

Rg (U) = Control variable, voltage-dependent

$$\text{Set value [V]} = \frac{\text{Data byte}}{65536} \cdot 10V$$

$$\text{Set value [unit (Rg)]} = \text{Rg (set value [V])}$$

$$= \text{Rg} \left(\frac{\text{Data byte}}{65536} \cdot 10V \right)$$

or

Rg (U) = Control variable, current-dependent

$$\text{Set value [mA]} = \frac{\text{Data byte}}{65536} \cdot 16mA + 4mA$$

$$\text{Set value [unit (Rg)]} = \text{Rg (set value [V])}$$

$$= \text{Rg} \left(\frac{\text{Data byte}}{65536} \cdot 16mA + 4mA \right)$$

For voltages at the analogue input that are less than the value defined by the point 1 X co-ordinate, the set value is the value defined in the point 1 Y co-ordinate.
For voltages at the analogue input that are greater than the value defined by the point 2 X co-ordinate, the set value is the value defined in the point 2 Y co-ordinate.
In between, the set value changes linearly between the two values specified in the Y co-ordinates (see chart above).

Limitation:
X co-ordinate (point 1) ≤ X co-ordinate (point 2)

Comment:
If Y co-ordinate (point 1) > Y co-ordinate (point 2) is selected, the gradient of the characteristic will be negative.
In such cases, the motor stop function will become effective at a voltage > 9V at the analogue input

2.32 Control limitation

Address : D12F
Write authorisation : ebm-papst

This parameter defines which limitation functions are activated.

Encoding:

Reserved	Reserved	0	0	Reserved	Reserved	I	P

The MSB is of no relevance!

I ..... Coil current limitation
P ..... Power limitation

2.33 Output 0 to 10V / speed monitoring

2.33.1 0 to 10V output function / speed monitoring

Address of function : D130
Write authorisation : ebm-papst, customer

This parameter determines
- Which variable is given out at the output
- What form the output signal has
- The number of impulses per revolution given out at the output

Encoding LSB:

Value	Output parameter	Signal form
0	Fan modulation level	0 to 10V
1	Actual speed	0 to 10V
2	Phase control factor fab	0 to 10V
3	Actual speed	Pulses / revolution

The electronics for a fan only supports one signal form (either "0 to 10V" or "pulses / revolution"). It is *mandatory* to select this signal form.

Encoding MSB:

Number of pulses/revolution = Data byte

The value is only relevant if the signal form "pulses / revolution" is selected.

Applications:

a) Fan modulation level -> 0 to 10V (value = 0)

This configuration causes the specification for the modulation level to be output at the 0 to 10V output. This signal can be connected to the set value input for other fans. These will then run with the same modulation level. In the event of a fault, the modulation level at the output is always 0%.

b) Actual speed value -> 0 to 10V (value = 1)

This configuration causes a voltage proportional to the speed to be output at the 0 to 10V output. The value 10V is achieved for a speed n = 1.02 * nMax
The signal can be used to evaluate the actual speed.

c) System modulation level -> 0 to 10V (value = 2)

This configuration causes the specification for the modulation level to be output at the 0 to 10V output. This signal can be connected to the set value input for other fans. These will then run with the same modulation level.
In the event of a fault on the fan, a modulation level continues to be output at the 0 to 10V output, just as if no fault were present. The advantage this provides is that despite a fault on the master fan, the other fans of the system can continue to be controlled.

Additional explanations about the system modulation level:

- In "closed loop speed control" mode, this setting is not logical, as no speed information is available to the other fans at the master fan, and thus control is not possible in the event of fault.
- In "sensor control" mode, in the event of a fault, the sensor variable continues to be used for control and the modulation level is output at the 0 to 10V output level.
- In "control" mode, in the event of fault, the preset modulation level continues to be output at the 0 to 10V output.
- If a fault is no longer present at the master fan, from this time on, the modulation level of the master fan is again output at the 0 to 10V output. Because this is generally increased in a ramp function from 0% on, this curve will also be applied at the output of the 0 to 10V and thus the entire system will be shut down for a short period.

d) Actual speed -> pulses / revolution (value = 3)

This configuration causes a frequency proportional to the speed to be output at the output. The number of pulses per revolution can be selected. The signal can be used to evaluate the actual speed.

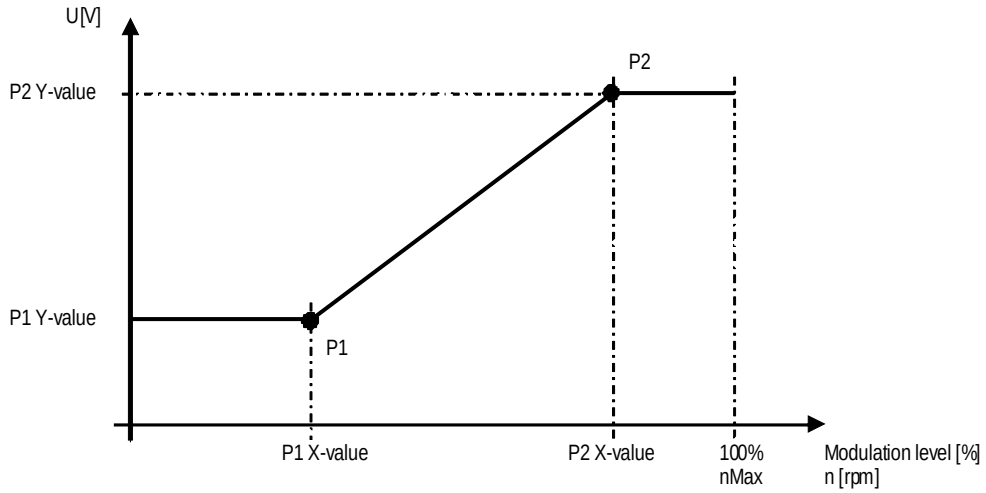
2.33.2 0 to 10V output characteristic

Address of point 1 X co-ordinate : D140
Address of point 1 Y co-ordinate : D141
Address of point 2 X co-ordinate : D142
Address of point 2 Y co-ordinate : D143

Write authorisation : ebm-papst, customer, end customer

These parameters are used to assign a voltage from the 0 to 10V output to a output size defined by the parameter "Output function 0 to 10V / speed monitoring".

The parameters are only relevant if signal form "0 to 10V" is selected (see 2.33.1)



Points P1 and P2 may be moved at random within the diagram.

Encoding:

<u>Output function 0 to 10V = modulation level</u>

The X co-ordinate defines a modulation level

$$Modulation\ level\ [\%]\ (Px) = \frac{Data\ bytes}{65536} \cdot 100\%$$

<u>Output function 0 to 10V = actual speed</u>

The X co-ordinate defines a speed

$$n\ [rpm]\ (Px) = \frac{Data\ bytes}{64000} \cdot nMax$$

nMax [rpm] to maximum speed in revolutions per minute (see 2.27 Maximal speed)

The Y co-ordinate defines the associated voltage at the 0 to 10V output:

$$U\,[V]\,(Px) = \frac{Data\ byte}{65536} \cdot 10V$$

Limitation:
X co-ordinate (point 1) ≤ X co-ordinate (point 2)

Applications:

A characteristic with the following values is recommended
P1X = 0x0000 0% / 0 rpm
P1Y = 0x0000 0V
P2X = 0xFF00 100% / 1.02 * nMax
P2Y = 0xFF00 10V

In this case:

$$U_{0\ to\ 10V}\,[V] = \frac{Modulation\ level\ (specifications)\ [\%]}{100} \cdot 10V$$

or:

$$U_{0\ to\ 10V}\,[V] = \frac{n\ Actual\ [rpm]}{1.02 \cdot n\ Max\ [rpm]} \cdot 10V$$

nMax [rpm] to maximum speed in revolutions per minute (see 2.27 Maximal speed)

A cascaded startup of the motors can be achieved by shifting the characteristic parallel downwards.

2.34 Maximum power

2.34.1 Maximum power

Address of maximum power : D155
Write authorisation : ebm-papst, customer

Encoding:

$$P\ max\ [W] = \frac{Data\ byte}{256} \cdot Reference\ Uz\ [V] \cdot Reference\ Iz\ [A]$$

P max ..... maximum power
Ref. Uz ..... DC-link voltage reference variable (see 2.50 Reference value of DC-link voltage).
Ref. Iz ..... DC-link current reference variable (see 2.51 Reference value of DC-link current).

The MSB is of no relevance!

The motor will limit the power to the value specified here if the function in the parameter "Control limitation" is activated (see 2.32 Control limitation).

Limitation:
Maximum power ≤ maximum permitted power (see 2.34.2 Maximum permitted power)

2.34.2 Maximum permitted power

Address for maximum permitted power : D135
Write authorisation : ebm-papst

Encoding:

$$P\ max,\ perm\ [W] = \frac{Data\ byte}{256} \cdot Reference\ Uz\ [V] \cdot Reference\ Iz\ [A]$$

P max, zul ..... maximum permitted power
Ref. Uz ..... DC-link voltage reference variable (see 2.50 Reference value of DC-link voltage).
Ref. Iz ..... DC-link current reference variable (see 2.51 Reference value of DC-link current).

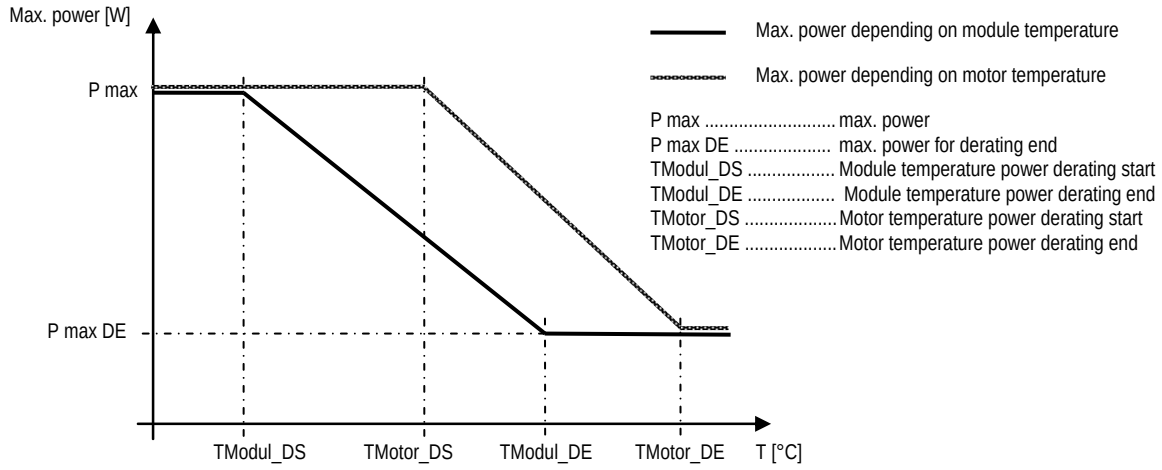
The MSB is of no relevance!

This parameter defines the upper limit for the maximum power (see 2.34.1) and the max power for derating end (see 2.34.3).

2.34.3 Power derating

Address for max. power for derating end	: D136
Address for Module temperature power derating start	: D137
Address for module temperature power derating end	: D138
Address for motor temperature power derating start	: D14D
Address for motor temperature power derating end	: D14E
Write authorisation	: ebm-papst, customer

Depending on the module temperature (see 3.13 Module temperature) and the motor temperature (see 3.14 Motor temperature), the maximum power reduces based on the following curve. Each temperature can have its own defined curve.



From this curve, the follow result

- A max. power value depending on the module temperature (Pmax_TModul)
- A max. power value depending on the motor temperature (Pmax_TMotor)

The power is limited to the smaller of the two Pmax_TModul and Pmax_TMotor values.

Encoding:

$$P\ max\_DE\ [W] = \frac{Data\ byte}{256} \cdot Reference\ Uz\ [V] \cdot Reference\ Iz\ [A]$$

TModul_DS [°C]	=	*Data byte*
TModul_DE [°C]	=	*Data byte*
TMotor_DS [°C]	=	*Data byte*
TMotor_DE [°C]	=	*Data byte*

P max DE..... Maximum power for derating end
Ref. Uz ..... DC-link voltage reference variable (see 2.50 Reference value of DC-link voltage).
Ref. Iz ..... DC-link current reference variable (see 2.51 Reference value of DC-link current)

TModul DS ..... Module temperature power derating start
TModul DE ..... Module temperature power derating end
TMotor DS .....Motor temperature power derating start
TMotor DE .....Motor temperature power derating end

The MSB is of no relevance, respectively!

The values "Max. power" and "Max. power for temperature derating end" are the same for both curves.

For temperatures below the value defined in TModul DS or TMotor DS, maximum power corresponds to the value of P max.
For temperatures above the value defined in TModul DE or TMotor DE, maximum power corresponds to the value of P max_DE.
In between, the maximum power changes linearly between the values of P max and P max_DE.
(see chart above).

Limitation:
If T DS ≥ T DE, T DS is limited to T DE
If P max DE > P_{max} or $P_{max\ zul}$, P max DE is limited to P_{max} or $P_{max\ zul}$

2.35 Maximum coil current

Address : D13B
Write authorisation : ebm-papst

Encoding:

$$I\ max,\ eff\ [A] = \frac{Data\ byte}{170} \cdot Reference\ Iz\ [A]$$

Bezug Iz ..... DC-link current reference variable (see 2.51 Reference value of DC-link current).

The MSB is of no relevance!

If the function in the parameter "Control limitation" is activated, the motor will limit the coil current (effective value) to the value specified here (see 2.32 Control limitation).
The coil current is made up of DC-link current / modulation level

2.36 Limit speed for running monitor

Address : D145
Write authorisation : ebm-papst, customer

Encoding:

The limit speed for the running monitor is made up of the parameter and the maximum speed:

$$\textit{Limit speed for running monitor [rpm]} = \frac{\textit{Data bytes}}{64000} \cdot \textit{nMax [rpm]}$$

nMax [rpm] to maximum speed (in revolutions per minute) - see 2.27Maximal speed

If the actual speed (see 3.8 Actual speed) is less than the limit speed for the running monitor, the error relay will trip. The flag "n_Low" is set in the "Warning" input register (see 3.10Warning)
If "Limit speed for running monitor" = 0, the entire function is deactivated.

2.37 Actual sensor value source

Address : D147
Write authorisation : ebm-papst, customer, end customer

Encoding:

The parameter "Actual sensor value source" defines the input from which the actual sensor value (see 3.19 Actual sensor values) is determined.
The following options are available for selection:

Value	Actual sensor value source
0x00	Input 1 ("Set value input")
0x01	Input 2 ("Sensor input")
0x02	Maximum (input 1; input 2)
0x03	Minimum (input 1; input 2)
0x04	Mean value (input 1; input 2

The MSB is of no relevance!

If input 1 is selected in any form as the actual sensor value source (i.e. value ≠ 0x01), ensure that in the parameter source set value (see 2.10Source set value), the "Analogue input" option is not selected. Otherwise, the same input will be used for the set value and the actual sensor value!
In this case, only the option "RS485" is logical as the source set value.

2.38 Interface settings

2.38.1 Transmission rate

Address : D149
Write authorisation : ebm-papst, customer

Encoding:

Value	Transmission rate
0x00	1200 bit/sec
0x01	2400 bit/sec
0x02	4800 bit/sec
0x03	9600 bit/sec
0x04	19,200 bit/sec
0x05	38,400 bit/sec
0x06	57,600 bit/sec
0x07	115,200 bit/sec

The MSB is of no relevance!

The standard value 19,200 bit/sec (0x04) is recommended for the transmission rate.

Make sure that the same transmission rate is selected for the fan as it is for the master (e.g. EC Control). Otherwise, communication with the fan will not be possible!

If the value set by the customer (holding register D249) or the factory default setting (holding register D2C9) differ from the value selected here, it must be ensured that the transmission rate is also changed to that of the customer setting or factory default setting (see and 2.52.6) after the data transfer. No communication is then possible with the previous setting

2.38.2 Parity configuration

Address : D14A
Write authorisation : ebm-papst, customer

Encoding:

Value	Parity configuration	Number of data bits	Parity	Number of stop bits
0x00	8E1	8	Even	1
0x01	8O1	8	Odd	1
0x02	8N2	8	None	2
0x03	8N1	8	None	1

The MSB is of no relevance!

The standard value 8E1 (0x00) is recommended for the parity configuration.

Note:
If parity configuration 8N1 is selected, the unit will no longer correspond to the Modbus specification "MODBUS over Serial Line Specification & Implementation Guide V1.0" as this specifies a frame of 11 bits.

Make sure that the same parity configuration is selected for the fan as it is for the master (e.g. EC Control). Otherwise, communication with the fan will not be possible!

If the value set by the customer (holding register D24A) or the factory default setting (holding register D2CA) differ from the value selected here, it must be ensured that the parity configuration is also changed to that of the customer setting or factory default setting (see and 2.52.6) after the data transfer. No communication is then possible with the previous setting.

2.38.3 Procedure for changing the interface settings

If the parameters "Transmission rate" and "Parity configuration" do not correspond to the interface setting in the master (e.g. EC Control), no communication will be possible.

If the settings are different, the interface settings will not be checked by the fan, i.e. the interface settings of the fan MUST be known.

To prevent complications caused by interfaces with different settings, it is advisable to leave the interface settings at the preset values:
Transmission rate : 19,200 bit/sec (0x04)
Parity configuration : 8E1 (0x00)

If a change is necessary, take care to ensure that the master and fan have the same settings at all times to guarantee error-free communications.

The new interface settings for the fan will only be adopted after an appropriate command on the parameter "Reset" (Adopt parameter or Reset) - see 2.2 - or when the unit is switched off ("Power off"). Until then, the fan will continue working with the previous settings.

Recommended procedure for changing the interface settings, for example when changing from 19,200 bit/sec, 8E1 to 38,400 bit/sec, 8N2:

- Interface settings at the master must initially be 19,200 bit/sec, 8E1.
- Write password: XX 10 D0 02 00 03 06 PP PP PP PP PP PP CS CS
  (changing the interface settings requires at least "Customer" authorisation)
- Write the transmission speed with the value 38,400 bit/sec: XX 06 D1 49 00 05 CS CS
- Write the parity configuration with the value 8N2: XX 06 D1 4A 00 02 CS CS
- Trigger Adopt parameters: XX 06 D0 00 00 02 CS CS
- After the fan has confirmed the command, it will work with the new setting.
- Change the interface setting in the master: 38,400 bit/sec, 8N2

XX ..... Fan address
PP ..... Password
CS ..... CRC checksum

2.38.4 Interface emergency function:

Two default settings for the interface are stored in the fan.
The fan switches automatically to these default settings for short periods if faulty telegrams are received (for example, as a result of different settings of the interface on the master and slave):

- After 10 failed attempts, the fan will automatically communicate using the setting 19,200 bit/sec, 8E1.
- After an additional 10 failed attempts, the fan communicates automatically at the setting 2400 bit/sec, 8E1
  Exception: For configuration to 115,200 bit/sec, this step is omitted!
- After yet another 10 failed attempts, the fan switches back to the settings for "Transmission speed" and "Parity configuration" stored in the holding registers to prevent unwanted continuous activations of the interface emergency function.
- For additional failed attempts, the interface is no longer switched over automatically, i.e. the interface emergency function can be activated only by switching the device off and on again or carrying out a software reset.

The interface emergency function can only be activated when no valid telegram has been detected from the fan. To make sure that the interface emergency function is enabled, the operating voltage of the unit should first be switched off and back on again.

Application:

If the interface settings of the fan are unknown, they can be identified with the interface emergency function.

Recommended procedure:

- Switch fan on
- Set interface on masters to standard values 19,200 bit/sec, 8E1
- Read interface setting: XX 03 D1 49 00 02 CS CS
- After 10 failed attempts, the fan will automatically communicate using the standard setting 19,200 bit/sec, 8E1. In other words, the command given above has to be transmitted 11 times until a response is transmitted by the fan.
- The interface settings of the fan can be identified from the response transmitted by the fan.
- The interface settings of the master and slave can now be synchronised.

Note:

The fan should not be run continuously in the interface emergency function as the values defined in the parameters "Transmission rate" and "Parity configuration" are activated again after every "Reset" or "Adopt parameters" command (see 2.2).
The interface settings of the master and fan must be synchronised!

2.39 Sheding function

Address of Sheding function : D150
Write authorisation : ebm-papst, customer

Address for max. starting modulation level : D151
Write authorisation : ebm-papst

Address for number of startup attempts : D152
Write authorisation : ebm-papst, customer

Encoding:

a) Sheding function

Value	Sheding function
0	Inactive
1	Active

The MSB is of no relevance!
Permitted value range: 0 to 1

b) Max. starting modulation level

$$\textit{Max. starting modulation level [\%]} = \frac{\textit{Data byte}}{256} \cdot 100\%$$

The MSB is of no relevance!

c) Number of startup attempts

$$\textit{Number of startup attempts} = \textit{Data byte}$$

The MSB is of no relevance!

Function:

The Sheding function is intended to enable a fan with fan blades that have frozen into place to shake the motor free by attempting to start the fan in both directions in alternation; during this process, the modulation level is increased during each attempt.

The first attempt begins with the value specified under starting modulation level (see 2.24 Starting modulation level) and the desired direction of rotation. If this does not succeed in getting the fan rotating, during each additional attempt, the direction of rotation is reversed and the starting modulation level is increased by 5%, to

a value no higher than that specified in "Max. starting modulation level". At the same time, an "Sheding function active" warning is generated (see 3.10 Warning).

This process is continued until the number of attempts specified in the "Number of startup attempts" is reached. If this also does not succeed in getting the fan rotating, the "Locked motor" error message (see 3.9 Motor status) is generated. During additional startup attempts, the modulation level is returned to the value defined in the parameter "Starting modulation level" and not increased any more.

2.40 Relay drop-out delay

Address : D153
Write authorisation : ebm-papst, customer, end customer

In the event of a fault, the relay drops out after a delay equal to the time specified here.

Encoding:

Relay drop-out delay [s] = *Data byte*

The MSB is of no relevance!

2.41 Emergency operation function on/off

Address : D15C
Write authorisation : ebm-papst, customer

Encoding:

This parameter specifies whether the emergency operation function is active or inactive.

Value	Emergency operation function
0	Inactive
1	Active

The MSB is of no relevance!
Permitted value range: 0 to 1

Function:

Emergency operation function active (1)

The emergency operation function monitors the signal wire of the set value and reacts to a failure of the signal.
The function depends on the parameter "Source set value" (see 2.10):

Source set value = 0 (analogue input): Monitoring of the input for set value (potentiometer):
If the voltage at the set value input drops below the value defined in the parameter "Potentiometer curve - limit for cable break" (see 2.45), the fan will automatically switch to the set value for emergency operation (see 2.42).

Source set value = 1 (RS485): Monitoring the RS485 bus interface:
If no command is sent to the fan for the time specified in the parameter "Emergency operation time lag" (see 2.43) the fan will automatically switch to the emergency operation set value (see 2.42).
The direction of rotation is simultaneously set to the value of "Emergency operation running direction" (see 2.44).

Emergency operation function inactive (0)

If the emergency operation function is inactive, the signal wire for the set value is not monitored and is not switched to the set value for emergency operation.

2.42 Emergency operation set value

Address : D15D
Write authorisation : ebm-papst, customer

ebm-papst Mulfingen GmbH & Co. KG | Bachmühle 2 · 74673 Mulfingen, Germany · Phone: +49-7938-81-0 · Fax: +49-7938-81-110 · www.ebmpapst.com · info1@de.ebmpapst.com

If a failure of the set value signal is detected, the fan will automatically switch to the set value for emergency operation given here. For more detailed information about the function, please refer to 2.41.

Encoding:

Remark: The encoding corresponds to the encoding of the default set value (D001)
The 4 LSBits are of no relevance for the set value and will always be assumed to be 0.

a) in closed loop speed control

The emergency operation set value denotes a speed:

$$\textit{Emergency operation set value [rpm]} = \frac{\textit{Data bytes}}{64000} \cdot \textit{nMax [rpm]}$$

nMax [rpm] to maximum speed in revolutions per minute (see 2.27 Maximal speed)

The value zero means motor standstill

b) in open loop PWM control

The emergency operation set value denotes a modulation level:

$$\textit{Emergency operation set value [\%]} = \frac{\textit{Data bytes}}{65536} \cdot 100\%$$

The value zero means motor standstill

c) in closed loop sensor control

The emergency operation set value denotes a sensor variable:

A sensor that converts the control variable into a voltage of 0 to 10V or a current of 4 to 20mA must be connected to the fan.

A set value can be entered for the output voltage or the output current of the sensor used.
The emergency operation set value for the control variable is then made up of the set value parameter and the Rg(U/I) characteristic of the sensor used.

Rg (U) = Control variable, voltage-dependent

$$\textit{Emergency operation set value [V]} = \frac{\textit{Data bytes}}{65536} \cdot 10V$$

$$\begin{aligned} \textit{Emergency operation set value [unit (Rg)]} &= \textit{Rg (set value bypass [V])} \\ &= \textit{Rg} \left(\frac{\textit{Data bytes}}{65536} \cdot 10V \right) \end{aligned}$$

or

Rg (U) = Control variable, current-dependent

$$\text{Emergency operation set value [mA]} = \frac{\text{Data bytes}}{65536} \cdot 16mA + 4mA$$

$$\text{Emergency operation set value [unit (Rg)]} = Rg\ (\text{set value bypass [V]})$$

$$= Rg\ \left(\frac{\text{Data bytes}}{65536} \cdot 16mA + 4mA\right)$$

2.43 Emergency operation time lag

Address : D15E
Write authorisation : ebm-papst, customer

Only for set value source = 0x01 (RS485):
If no command is sent to the fan for the time specified here, the fan will automatically switch to the emergency operation set value (see 2.42Emergency operation set value), provided that the emergency operation function is activated. For more detailed information about the function, please refer to 2.41.

Encoding:

$$\text{Emergency operation time lag [ms]} = \text{Data byte} \cdot 100ms$$

The MSB is of no relevance!

2.44 Emergency operation running direction

Address : D15B
Write authorisation : ebm-papst, customer

If the emergency operation function is activated and a break in the Modbus connection is detected, the direction of rotation is set to the value defined here. The settings for the parameters Preferred running direction (see 2.12) and Running direction source (see 2.11) are not relevant then.

Encoding:

Value	Emergency operation running direction
0	counter-clockwise
1	clockwise
2	none (configured direction of rotation remains intact)

The MSB is of no relevance!

Permitted value range: 0 to 2

2.45 Potentiometer curve, limit value for cable break

Address : D15F
Write authorisation : ebm-papst, customer

Only for set value source = 0x00 (analogue input):
If the voltage at the set value input drops below the value defined here, the fan will automatically switch to the emergency operation set value (see 2.42), provided the emergency operation function is activated. For more detailed information about the function, please refer to 2.41.

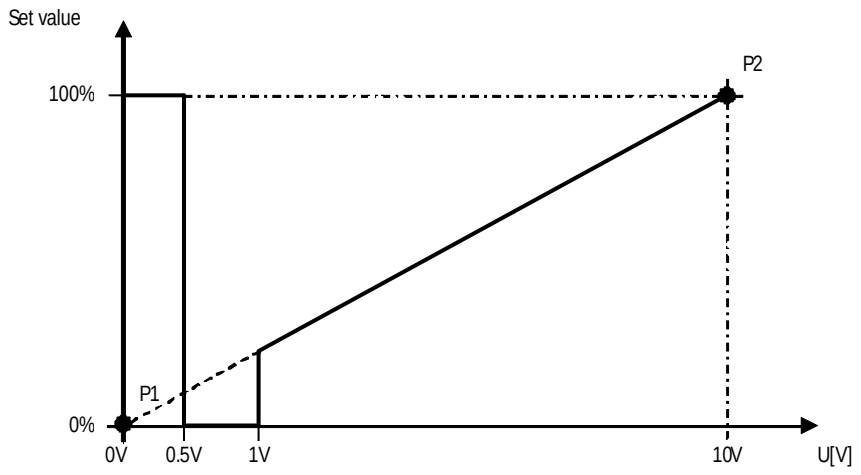
Encoding:

$$U\_cable\ break\ [V] = \frac{Data\ byte}{65536} \cdot 10V$$

Example:

Emergency operation function	= on
Set value for emergency operation	= 100%
Potentiometer characteristic U_cable break	= 0.5V
Potentiometer characteristic P1	= (0V | 0%)
Potentiometer characteristic P2	= (10V | 100%)
Enable motor stop	= 1

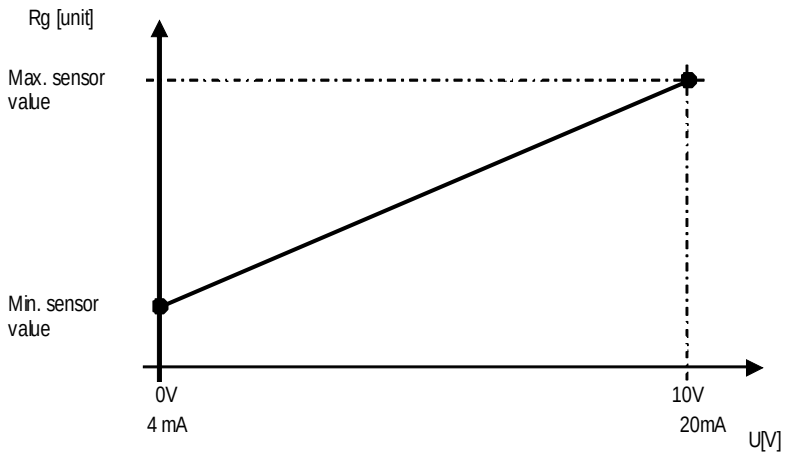
From this, the following curve is derived:



2.46 Sensor

Address for min. sensor value : D160 / D161
Address for max. sensor value : D162 / D163
Address of sensor unit : D164 - D169
Write authorisation : ebm-papst, customer, end customer

These values define the characteristic of the specified sensor as shown in the chart below.



The necessary data for min. sensor value and max. sensor value can be found in the data sheet for the sensor.

These parameters are only intended for use in the master. The behaviour of the fan is not influenced by these parameters.

Encoding:

a) Minimum sensor value

Min. sensor value [phys. unit] = Data bytes

The minimum sensor value is stored in "float" format!

b) Maximum sensor value

Max. sensor value [phys. unit] = Data bytes

The maximum sensor value is stored in "IEEE float" format!

c) Sensor unit

Sensor unit [ASCII] = Data bytes

The sensor unit is stored in "ASCII" format!

2.47 Customer data

Address : D170 - D17F
Write authorisation : ebm-papst, customer

For the customer, there are a total of 16 parameters (each with 16 bit) in this range.
Any values required can be stored here.

The behaviour of the fan is not influenced by these parameters.

2.48 Operating hours counter (backup)

Address : D180
Write authorisation : ebm-papst

Encoding:

Operating time [h] = Data bytes

This parameter is a backup copy of the parameter D009 "Operating hours counter" (see 2.7 Operating hours counter). The parameter is continuously updated.

2.49 Error history

Address of error indicator	: D182
Address of 1st error	: D184
Address of 1st error timing	: D185
Address of error history 1 to 13	: D186, D188, D18A, D18C, D18E, D190, D192, D194, D196, D198, D19A, D19C, D19E
Address of error history timing 1 to 13	: D187, D189, D18B, D18D, D18F, D191, D193, D195, D197, D199, D19B, D19D, D19F
Write authorisation	: ebm-papst

A) 1st error

The first error that is detected in the service life of the fan is stored under the parameter "1st error".
At the same time, the reading on the operating hour counter at this time is stored in the parameter "1st error timing". The parameters are written automatically by the fan.

B) Error history

The error history contains the last 13 errors that were detected in the fan.
The record of each error includes the respective operating hours counter reading in the parameter "Error history timing". The parameters are written automatically by the fan.

The error indicator (D182) specifies the address of the last error to be detected in the error history.
The previous error is then at the preceding address.

Example:

Error indicator = D196

Then:

D196	Last error (error n)	D197	Timing of last error (error n)
D194	Error n-1	D195	Timing of error n-1
D192	Error n-2	D193	Timing of error n-2
D190	Error n-3	D191	Timing of error n-3
D18E	Error n-4	D18F	Timing of error n-4
D18C	Error n-5	D18D	Timing of error n-5
D18A	Error n-6	D18B	Timing of error n-6
D188	Error n-7	D189	Timing of error n-7
D186	Error n-8	D18D	Timing of error n-8
D19E	Error n-9	D19F	Timing of error n-9
D19C	Error n-10	D19D	Timing of error n-10
D19A	Error n-11	D19B	Timing of error n-11
D198	Error n-12	D199	Timing of error n-12

Every time a new error is detected, the error indicator increases by 2 and the error and its timing are stored at the address shown on the error indicator. The addresses are written automatically by the fan.
When the error indicator reaches the final value D19E, it will be reset to the initial value D186 when the next error is detected.

Encoding:

Error indicator:

Address of error indicator = Data bytes

Error:

MSB	0	0	0	0 *)	0	0	0	0
LSB	BLK	HLL	TFM	0 *)	SKF	TFE	0	0 *)

If bit = 1, the error described below has been detected:

BLK: Locked motor
HLL: Hall sensor error
TFM: Motor overheated
SKF: Communications error between bus controller and commutation controller
TFE: Power mod overheated

*) The errors "DC-link undervoltage" and "Phase failure" and the general fault detection "Fan bad" are not stored, even if they are set in the motor status (see Motor status)!

Comment:
"Communication error" is saved only if the last error was a different one, since a restart is permitted for this error.

Error timing:

Time of error [h] = Data bytes

2.50 Reference value of DC-link voltage

Address : D1A0
Write authorisation : ebm-papst

To keep the resolution variable, all values for the DC-link voltage are based on this reference value.

Encoding:

Reference Uz [mV] = Data bytes · 20mV

2.51 Reference value of DC-link current

Address : D1A1
Write authorisation : ebm-papst

To keep the resolution variable, all values for the DC-link current are based on this reference value.

Encoding:

Reference Iz [mA] = Data bytes · 2mA

2.52 Production data

Production data means information data to enable the device concerned to be traced. The behaviour of the fan is not influenced by these parameters.

2.52.1 Fan serial number and production date

Address of serial number : D1A2 / D1A3
Address of production date : D1A4
Write authorisation : ebm-papst

For each fan, ebm-papst assigns an individual serial number. This serial number has 10 digits.

Format: YYWW00XXXX

YY : Year of production
WW : Calendar week of production
00 : Fixed value 00
XXXX : Sequential number

The first 4 digits contain the production date (year/calendar week).

With the beginning of each production week, the number XXXX starts counting up from zero and is increased by 1 for each fan. Each character can represent values from 0-9 and from A-Z. The maximum number of characters that can be encoded for each digit is thus 36, i.e. 36^4 = 1 679 616 devices / week

a) Serial number (D1A2 / D1A3)

In the parameter "Serial number", only the part XXXX (sequential number) is stored.
Each digit is encoded as an ASCII value.

Encoding:

Serial number [ASCII] = Data bytes

b) Production date (D1A4)

The YYWW part is stored in the parameter "Production date".
Each digit is encoded as a hex value.

Encoding:

Production year = Data byte (MSB)

Production week = Data bytes (LSB)

Example:

ebm-papst serial number: 09230**12GY**
-> Content D1A2: 0x4759 (GY)
Content D1A3: 0x3132 (12)
Content D1A4: 0x0917 (09/23)

2.52.2 Fan type

Address : D1A5 - D1AA
Write authorisation : ebm-papst

Encoding:
The fan type is stored here in ASCII code
D1A5 contains the first two characters.
D1AA contains the last two characters.

3 Input Register

3.1 Overview

Input registers are stored in the RAM of the fan.
Input registers only have read access

The following list gives an overview of all parameters.
The function of the parameters is described in the following chapters

Modbus Address	Designation
D000	Identification
D001	Max. number of bytes
D002	Software name of bus controller
D003	Software Version of bus controller
D004	Software name of commutation controller
D005	Software version of commutation controller
D010	Actual speed
D011	Motor status
D012	Warning
D013	DC-link voltage
D014	DC-link current
D015	Module temperature
D016	Motor temperature
D017	Electronics temperature
D018	Current direction of rotation
D019	Current modulation level
D01A	Current set value
D01B	Actual sensor value
D01C	Enable input status
D01D	Current parameter set
D01E	Current control function
D01F	Reserved
D020	Reserved
D021	Current power
D022	Reserved
D023	Actual sensor value 1
D024	Actual sensor value 2
D025 - D026	Reserved

3.2 Identification

Address : D000

Identification specifies the type of electronics and protocol concerned.

Encoding:

Value	Device	Specification version
00 01	ebm-papst series 84 / 112 / 150	1.02 *)
00 02	ebm-papst series 84 / 112 / 150	2.01, 3.00 - 3.01 *)
00 03	Customer application	*)
00 04	Customer application	*)
00 05	Customer application	*)
00 06	ebm-papst series 84 / 112 / 150	3.02 *)
00 07	ebm-papst series 84 / 112 / 150	4.00 *)
00 08	ebm-papst series 84 / 112 / 150 / 200	5.00
00 09	Customer application	*)
00 0A	ebm-papst series 84 / 112 / 150 / 200 Lite	5.00 Lite *)
04 00	Customer application	*)
05 00	Customer application	*)
06 00	Customer application	*)

*) Devices with identification ≠ 0x0008 do not correspond to these specifications.
In such cases, the corresponding document should be used.

3.3 Maximum number of bytes

Address : D001

This parameter specifies the maximum number of bytes that a telegram sent via Modbus may contain.

Encoding:

Max. number of bytes = Data bytes

3.4 Software name of bus controller

Address : D002

This parameter specifies the number of the software of the bus controller (without the version).

Encoding:

Software Name = Data bytes

3.5 Software Version of bus controller

Address : D003

This parameter specifies the software version of the bus controller.

Encoding:

Software Version = Data byte LSB

The MSB is always 0.

3.6 Software name of commutation controller

Address : D004

This parameter specifies the number of the software of the commutation controller (without the version).

Encoding:

Software Name = Data bytes

3.7 Software version of commutation controller

Address : D005

This parameter specifies the software version of the commutation controller.

Encoding:

Software Version = Data byte LSB

The MSB is always 0.

3.8 Actual speed

Address : D010

Encoding:

The actual speed is made up of the speed parameter and the maximum speed:

$$Actual\ speed\ [rpm] = \frac{Data\ bytes}{64000} \cdot nMax\ [rpm]$$

nMax [rpm] to maximum speed (in revolutions per minute) - see 2.27 Maximal speed

Comment:
If the actual speed exceeds the value "1.02 * maximum speed", the display will be limited to the value "1.02 * maximum speed" (0xFFF0)

3.9 Motor status

Address : D011

The motor status specifies errors currently detected in the fan.

Encoding:

MSB	0	0	0	UzLow	0	0	0	0
LSB	BLK	HLL	TFM	FB	SKF	TFE	0	PHA

If a bit is set, the error described below has been detected:

UzLow: DC-link undervoltage

BLK: Locked motor
HLL: Hall sensor error
TFM: Motor overheated
FB: Fan bad (general error) *)
SKF: Communication error between master controller and slave controller
TFE: Power mod overheated
PHA: Phase failure (3-phase devices) or mains undervoltage (1-phase devices)

*) "Fan bad" is set for every error

3.10 Warning

Address : D012

A warning is a prestage to an error message, i.e. the limit value for the error message has almost been reached.

Encoding:
A set bit makes the warning active:

MSB	Shf	UeHigh	0	UzHigh	Heating	Cable break	n_Low	Reserved
LSB	Brake	UzLow	TEI_high	TM_high	TE_high	P_Limit	L_high	I_Limit

Shf : Sheding function function active - see 2.39 Sheding function
UeHigh : Supply voltage high
UzHigh : DC-link voltage high
Heating : Heating enabled
The motor should not be started when the heating is enabled!
Cable break : Cable break at set value analogue input
(Voltage at the analogue input < Limit value for cable break - see 2.45)
n_Low : Actual speed is less than limit speed for running monitor
(see 2.36 Limit speed for running monitor)

Brake : Brake mode: set if exterior drive is applied in opposite direction with high speed for prolonged period
UzLow : DC-link voltage low
TEI_high : Electronics interior temperature high
TM_high : Motor temperature high
TE_high : Output stage temperature high
P_Limit : Power limitation engaged
L_high : Line impedance too high (DC-link voltage unstable)
I_Limit : Current limitation engaged

3.11 DC-link voltage

Address : D013

Encoding:

$$Uz\ [V] = \frac{Data\ byte}{256} \cdot Reference\ Uz\ [V]$$

Bezug Uz .....DC-link voltage reference variable (see 2.50 Reference value of DC-link voltage).

3.12 DC-link current

Address : D014

Encoding:

$$Iz\ [A] = \frac{Data\ byte}{256} \cdot Reference\ Iz\ [A]$$

Bezug Iz .....DC-link current reference variable (see 2.51 Reference value of DC-link current)

3.13 Module temperature

Address : D015

Encoding:

$$T\_module\ [°C] = Data\ byte$$

3.14 Motor temperature

Address : D016

Encoding:

$$T\_motor\ [°C] = Data\ bytes$$

Format: signed integer

3.15 Electronics interior temperature

Address : D017

Encoding:

$$T\_EI\ [°C] = Data\ byte$$

3.16 Current direction of rotation

Address : D018

Encoding:

Value	Current direction of rotation
0	counter-clockwise
1	clockwise

3.17 Current modulation level

Address : D019

Encoding:

$$Modulation\ level = \frac{Data\ bytes}{65536} \cdot 100\%$$

3.18 Current set value

Address : D01A

Encoding:

<u>a) in closed loop speed control</u>

The set value denotes a speed:

$$Set\ value\ [rpm] = \frac{Data\ bytes}{64000} \cdot nMax\ [rpm]$$

nMax [rpm] to maximum speed in revolutions per minute (see 2.27 Maximal speed)

The value zero means motor standstill

<u>b) in open loop PWM control</u>

The set value denotes a modulation level:

$$Set\ value\ [\%] = \frac{Data\ bytes}{65536} \cdot 100\%$$

The value zero means motor standstill

c) in closed loop sensor control

The set value denotes a sensor variable:

A sensor that converts the control variable into a voltage of 0 to 10V or a current of 4 to 20mA must be connected to the fan.

A set value can be entered for the output voltage or the output current of the sensor used.
The set value for the control variable is then made up of the set value parameter and the Rg(U/I) characteristic of the sensor used.

Rg (U) = Control variable, voltage-dependent
Rg (U) = Control variable, voltage-dependent

$$\text{Set value [V]} = \frac{\text{Data bytes}}{65536} \cdot 10V$$

$$\text{Set value [unit (Rg)]} = \text{Rg (set value [V])}$$

$$= \text{Rg} \left(\frac{\text{Data bytes}}{65536} \cdot 10V \right)$$

or

Rg (U) = Control variable, current-dependent

$$\text{Set value [mA]} = \frac{\text{Data bytes}}{65536} \cdot 16mA + 4mA$$

$$\text{Set value [unit (Rg)]} = \text{Rg (set value [V])}$$

$$= \text{Rg} \left(\frac{\text{Data bytes}}{65536} \cdot 16mA + 4mA \right)$$

3.19 Actual sensor values

Address of actual sensor value : D01B
Address of actual sensor value 1 : D023
Address of actual sensor value 2 : D024

Two sensors can be connected to the fan.
Actual sensor value 1 outputs the current actual value of the external sensor at input 1 ("set value input").
Actual sensor value 2 outputs the current actual value of the external sensor at input 2 ("sensor input").

From these two values, depending on the actual sensor value source, the applicable actual sensor value (D01B) for the controller is determined (see 2.37Actual sensor value source). Only this actual value is used in "sensor control" mode.

Encoding:

$$\text{Actual value [V]} = \frac{\text{Data byte}}{65536} \cdot 10V$$

$$\text{Actual value [unit (Rg)]} = Rg\ (U)$$

$$= Rg\ \left(\frac{\text{Data byte}}{65536} \cdot 10V\right)$$

Actual value [V] = output voltage of sensor
Rg (U) = characteristic of sensor depending on voltage

3.20 Enable input status

Address : D01C

This parameter specifies the enable input status

Encoding:

Value	Enable
0	Enable off (motor stop)
1	Enable on (motor start allowed)

3.21 Current parameter set

Address : D01D

This parameter specifies which parameter set (1 or 2) is currently in use.

If the parameter "Parameter set source" has the value "Digital input Din2" (0) or "Digital input Din3" (2), the status of the external input "parameter set" is shown here.
If the parameter "Parameter set source" has the value "internal" (1), the value of the parameter "Parameter set internal" is shown here (see 2.14 Parameter set source, 2.15 Internal parameter set).

Encoding:

Value	Parameter set
0	Parameter set 1
1	Parameter set 2

3.22 Current control function

Address : D01E

If the parameter "Source of control function" has the value "Digital input Din3" (0) or "Digital input Din2" (2), the status of the external input "control function" is shown here.
If the parameter "Source of control function" has the value "internal" (1), the value of the parameter "Control function" is shown here (see 2.17 Source of control function, 2.18 Control function).

Encoding:

Value	Control function
0	Positive: Control variable = Actual value - Set value
1	Negative: Control variable = Set value - Actual value

Remarks:

For sensor control with temperature sensor, a positive control function equates to "heat" and a negative control function equates to "cool".

3.23 Current power

Address : D021

Encoding:

$$P\ [W] = \frac{Data\ bytes}{65536} \cdot Reference\ Uz\ [V] \cdot Reference\ Iz\ [A]$$

Bezug Uz .....DC-link voltage reference variable (see 2.50 Reference value of DC-link voltage).
Bezug Iz .....DC-link current reference variable (see 2.51 Reference value of DC-link current)