SD03461D/06/EN/01.25-00
71744645
2025-11-11

Special Documentation
Proline Teqwave MW 300/500 Modbus RS485

System integration

Endress+Hauser
People for Process Automation

Table of contents

1 About this document 4
1.1 Document function 4
1.2 Target group 4
1.3 Abbreviations 4
1.4 Symbols 5
1.5 Documentation 6
1.6 Registered trademarks 6

2 Modbus protocol 7
2.1 Operating principle 7
2.2 Modbus exception codes 7
2.3 Modes of operation 8
2.4 Interface configuration 8
2.5 Data types 9
2.6 Byte transmission sequence 9
2.7 Modbus data map 10
2.8 Register addressing 12

3 Description of functions 13
3.1 Modbus function codes 13
3.2 Response time 13

4 Appendix 14
4.1 Data types 14

5 Register information 15

1 About this document

1.1 Document function

This manual is special documentation; it does not replace the Operating Instructions and device parameters pertaining to the device. It serves as a reference manual and a complementary source of information for the integration of field devices with a Modbus interface into process control systems and controllers.

This supplementary documentation should only be used in conjunction with a measuring instrument with a Modbus RS485 interface.

1.2 Target group

This document is aimed at individuals who are incorporating the device into a Modbus network. It is a requirement that the reader has basic knowledge of Modbus technology.

More detailed information on Modbus technology can be found at www.modbus.org

1.3 Abbreviations

ADU	Application Data Unit
APL	Advanced Physical Layer
ASCII	American Standard Code for Information Interchange
CRC	Cyclic redundancy check
DHCP	Dynamic Host Configuration Protocol
ENP	Electronic nameplate
HTTP	Hypertext Transfer Protocol
ICMP	Internet Control Message Protocol
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet protocol
LSB	Least Significant Byte
MAC	Media access control
MBAP	Modbus application protocol header
MSB	Most Significant Byte
n/a	Not available
NaN	Not a Number (IEEE-754, 7Fh A0h 00h 00h)
ro	Read only
rw	Read and write
RS485	Recommended Standard 485
RTU	Remote terminal unit
TCP	Transmission control protocol

1.4 Symbols

1.4.1 Safety symbols

⚠ DANGER
This symbol alerts you to a dangerous situation. Failure to avoid this situation will result in serious or fatal injury.

⚠ WARNING
This symbol alerts you to a potentially dangerous situation. Failure to avoid this situation can result in serious or fatal injury.

⚠ CAUTION
This symbol alerts you to a potentially dangerous situation. Failure to avoid this situation can result in minor or medium injury.

NOTICE
This symbol alerts you to a potentially harmful situation. Failure to avoid this situation can result in damage to the product or something in its vicinity.

1.4.2 Symbols for certain types of information

Symbol	Meaning
	**Permitted** Procedures, processes or actions that are permitted.
	**Forbidden** Procedures, processes or actions that are forbidden.
	**Tip** Indicates additional information.
	Reference to documentation
	Reference to page
	Reference to graphic
▶	Notice or individual step to be observed
1., 2., 3...	Series of steps
↳	Result of a step

1.4.3 Symbols in graphics

Symbol	Meaning
1, 2, 3, ...	Item numbers
1., 2., 3., ...	Series of steps

1.5 Documentation

For an overview of the scope of the associated Technical Documentation, refer to the following:
- *Device Viewer* (www.endress.com/deviceviewer): Enter the serial number from the nameplate
- *Endress+Hauser Operations app*: Enter serial number from nameplate or scan matrix code on nameplate.

This Special Documentation and other documentation is available:
In the Download Area of the Endress+Hauser Internet site: www.endress.com → Downloads

This documentation is an integral part of the following Operating Instructions and device parameters:

Measuring instrument	Documentation code Modbus RS485	
Teqwave MW 300	BA02321D	GP01212D
Teqwave MW 500	BA02323D	GP01214D

1.6 Registered trademarks

Modbus®
Registered trademark of SCHNEIDER AUTOMATION, INC.

2 Modbus protocol

Modbus is an open, standardized communication protocol that is used in the areas of manufacturing automation, process automation and building automation.

The Modbus protocol is used to exchange data with intelligent field devices via master/slave (client/server) connections. The Modbus protocol can be used here on different transmission media and embedded in other communication protocols.

The specifications for the Modbus protocol are freely available on the Modbus organization website: https://modbus.org/

2.1 Operating principle

Modbus communication uses a simple request-response mechanism.

Client/Master	Request → ← Response	Server/Slave

A distinction is made between three cases:
- If the device receives a valid telegram with a valid request and can respond to this, then it responds with a response telegram with the function code of the request.
- If the device receives an invalid telegram (e.g. CRC error), it does not respond. The application must carry out troubleshooting after a defined time (time-out) has elapsed.
- If the device receives a valid telegram with an invalid request, then it responds with a Modbus exception. The application must carry out troubleshooting.

2.2 Modbus exception codes

If the Modbus slave detects an error in the request telegram from the master, it sends an error message to the master as a response, consisting of the slave address, function code, error code (exception code) and checksum. To indicate that this is an error message, the leading bit of the returned function code is set. The cause of the error is transmitted to the master via the error code (exception code).

Exception code	Description
01	ILLEGAL_FUNCTION The function code sent by the master is not supported by the measuring instrument (slave).
02	ILLEGAL_DATA_ADDRESS The register addressed by the master is not assigned (i.e. it does not exist) or the length of the data queried is too large.
03	ILLEGAL_DATA_VALUE The master attempts to write to a register which allows only read access. The value that appears in the data field is not permitted, e.g. span limits exceeded or incorrect data format.
04	SLAVE_DEVICE_FAILURE The slave did not respond to the request telegram from the master or an error occurred while processing the request telegram.

2.3 Modes of operation

2.3.1 Modbus RS485

ASCII

The data is transmitted as ASCII characters and can therefore be read by humans.

Start	Address	Function	Data	LRC	End
:	Two characters	Two characters	n characters	Two characters	CR LF

Field	Length	Description	Client	Server
Address	2 bytes	Identifies the target device.	Set by client.	Copied from the request to the response.
CRC	2 bytes	Modbus specification	Set by client.	Set by server.

RTU

The data is transmitted in binary form and is therefore more compact.

Address	Function	Data	CRC16
Two characters [1)]	One character	n characters	Two characters

1) 0 = broadcast, 247 = unicast

A reaction time of at least 3.5 characters must be maintained between two Modbus telegrams.

Field	Length	Description	Client	Server
Address	1 byte	Identifies the target device.	Set by client.	Copied from the request to the response.
CRC	2 bytes	Modbus specification	Set by client.	Set by server.

2.4 Interface configuration

2.4.1 Modbus RS485

Settings

Setting	Factory setting	Selection
Bus address	247	1 to 247
Transmission type	RTU	RTU, ASCII
Baud rate > bps	19200	1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200
Parity	Even (one stop bit)	Even (one stop bit), uneven (one stop bit), none (two stop bits)

If a valid address is set on the hardware DIP switch, the bus address cannot be changed in the menu. Setting via the hardware DIP switch is given priority.

Path
Setup → Communication

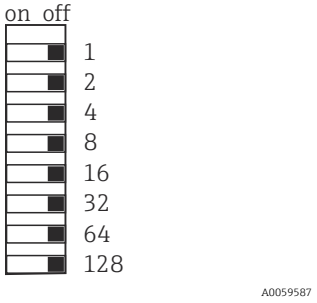
Setting the bus address via the software

Requirement
The hardware DIP switch is set to an address outside the valid range of 1 to 247.

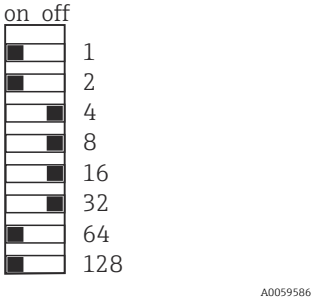
Path
Setup → Communication → Bus address

Setting the fieldbus address via the hardware

Valid fieldbus addresses are between 1 and 247. If an invalid address is set, software addressing is automatically enabled via the local setting.



1 As-delivered state



2 Example of a valid Modbus address 195

2.5 Data types

The measuring instrument supports the following data types:

FLOAT (floating point number IEEE 754) Data length = 4 bytes (2 registers)			
Byte 3	Byte 2	Byte 1	Byte 0
SEEEEEEE	EMMMMMMM	MMMMMMMM	MMMMMMMM
S = sign, E = exponent, M = mantissa			

INTEGER Data length = 2 bytes (1 register)	
Byte 1	Byte 0
Most Significant Byte (MSB)	Least Significant Byte (LSB)

STRING Data length = depends on the device parameter, e.g. presentation of a device parameter with a data length = 18 bytes (9 registers)				
Byte 17	Byte 16	...	Byte 1	Byte 0
Most Significant Byte (MSB)		...		Least Significant Byte (LSB)

2.6 Byte transmission sequence

Byte addressing, i.e. the transmission sequence of the bytes, is not set in the Modbus specification. It is therefore important to agree or adjust the addressing mode between the

master and slave or client and server when commissioning. This can be configured in the measuring instrument using the **Byte order** parameter.

The bytes are transmitted depending on the selection in the **Byte order** parameter:

FLOAT				
	Sequence			
Selection	1.	2.	3.	4.
1 – 0 – 3 – 2 *	Byte 1 (MMMMMMMM)	Byte 0 (MMMMMMMM)	Byte 3 (SEEEEEEE)	Byte 2 (EMMMMMMM)
0 – 1 – 2 – 3	Byte 0 (MMMMMMMM)	Byte 1 (MMMMMMMM)	Byte 2 (EMMMMMMM)	Byte 3 (SEEEEEEE)
2 – 3 – 0 – 1	Byte 2 (EMMMMMMM)	Byte 3 (SEEEEEEE)	Byte 0 (MMMMMMM)	Byte 1 (MMMMMMM)
3 – 2 – 1 – 0	Byte 3 (SEEEEEEE)	Byte 2 (EMMMMMMM)	Byte 1 (MMMMMMMM)	Byte 0 (MMMMMMMM)
* = factory setting, S = sign, E = exponent, M = mantissa				

INTEGER		
	Sequence	
Selection	1.	2.
1 – **0** – 3 – 2 * 3 – 2 – **1** – **0**	Byte 1 (MSB)	Byte 0 (LSB)
0 – **1** – 2 – 3 2 – 3 – **0** – **1**	Byte 0 (LSB)	Byte 1 (MSB)
* = factory setting, MSB = Most Significant Byte, LSB = Least Significant Byte		

2.7 Modbus data map

2.7.1 Function of the Modbus data map

The measuring instrument offers a special memory area, the Modbus data map (for a maximum of 16 device parameters), to allow users to call up multiple device parameters via Modbus and not only individual device parameters or a group of consecutive device parameters.

Grouping of device parameters is flexible and the Modbus master can read or write to the entire data block simultaneously with a single request telegram.

2.7.2 Structure of the Modbus data map

The Modbus data map consists of two data sets:

- Scan list: Configuration area
  The device parameters to be grouped are defined in a list, with their Modbus register addresses being entered in the list.
- Data area
  The measuring instrument reads out the register addresses entered in the scan list cyclically and writes the associated device data (values) to the data area.

2.7.3 Scan list configuration

For configuration, the Modbus register addresses of the device parameters to be grouped must be entered in the scan list. Please note the following basic requirements of the scan list:

Max. entries	16 device parameters
Supported device parameters	Only parameters with the following characteristics are supported: ▪ Access type: Read or write access ▪ Data type: Float or integer

Configuring the scan list via web server, FieldCare or DeviceCare

Carried out using the operating menu of the measuring instrument:
Expert → Communication → Modbus data map → Scan list register 0 to 15

Scan list	
No.	**Configuration register**
0	Scan list register 0
…	…
15	Scan list register 15

Configuring the scan list via Modbus

Carried out using register addresses 5001 to 5016

Scan list			
No.	**Modbus register**	**Data type**	**Configuration register**
0	5001	Integer	Scan list register 0
…	…	Integer	…
15	5016	Integer	Scan list register 15

2.7.4 Reading out data

The data area of the Modbus data map is accessed to read out the current values of the device parameters defined in the scan list register.

Access to data area	Via register addresses 5051 to 5081

Data area					
Device parameter value	**Modbus register**	**I/O data**	**As-delivered state of registers 5001 to 5016 for Modbus RS485**	**Data type ***	**Access ****
Value of scan list register 0	5051	Total solids	5089	Integer/Real	Read
Value of scan list register 1	5053	Load Rate	5091	Integer/Real	Read
Value of scan list register 2	5055	Corrected conductivity	3977	Integer/Real	Read
Value of scan list register 3	5057	Temperature	2017	Integer/Real	Read
Value of scan list register 4	5059	Totalizer 1	2610	Integer/Real	Read
Value of scan list register 5	5061	Actual diagnostics (no.)	6801	Integer/Real	Read
Value of scan list register 6	5063	Tot1 Control	2608	Integer/Real	Write

Data area					
Device parameter value	**Modbus register**	**I/O data**	**As-delivered state of registers 5001 to 5016 for Modbus RS485**	**Data type ***	**Access ****
Value of scan list register 7	–	–	–	Integer/Real	–
Value of scan list register 8	–	–	–	Integer/Real	–
Value of scan list register 9	–	–	–	Integer/Real	–
Value of scan list register 10	–	–	–	Integer/Real	–
Value of scan list register 11	–	–	–	Integer/Real	–
Value of scan list register 12	–	–	–	Integer/Real	–
Value of scan list register 13	–	–	–	Integer/Real	–
Value of scan list register 14	–	–	–	Integer/Real	–
Value of scan list register 15	–	–	–	Integer/Real	–
* Data type depends on the device parameters entered in the scan list. ** Data access depends on the device parameters entered in the scan list. If the device parameter entered supports read and write access, the parameter can also be accessed via the data area.					

2.8 Register addressing

Register	Data type	Access	Address range [1)]	Description
Discrete inputs	Bit	Read only	1XXXX	Is not used by the device.
Coils	Bit	Read/Write	2XXXX	Is not used by the device.
Input register	16-bit word	Read only	3XXXX	This data can only be read by the device.
Holding register	16-bit word	Read/Write	4XXXX	This data can be read by the device and changed by the application.

1) The register address is transmitted as a 16-bit word, with the first register being addressed with the value 0. The register addresses are specified in this document in the same way as they are transmitted in the Modbus telegrams.

It is also customary to use a notation in Modbus applications which counts the register addresses starting from a value of 1. In this case, the (register address – 1) is transmitted in the Modbus telegram.

3 Description of functions

3.1 Modbus function codes

Function codes are used to define which read or write action is carried out via the Modbus protocol. The measuring instrument supports the following function codes:

Code	Name	Description	Application
03	Read holding register	Controller reads one or more Modbus registers of the measuring instrument. A maximum of 125 consecutive registers can be read with one telegram: One register = 2 bytes The measuring instrument does not make a distinction between function codes 03 and 04; these codes therefore yield the same result.	Read device parameters with read and write access Example: Read mass flow
04	Read input register	Controller reads one or more Modbus registers of the measuring instrument. A maximum of 125 consecutive registers can be read with one telegram: One register = 2 bytes The measuring instrument does not make a distinction between function codes 03 and 04; these codes therefore yield the same result.	Read device parameters with read access Example: Read totalizer value
06	Write single registers	Controller writes a new value to **one** Modbus register of the measuring instrument. Use function code 16 to write multiple registers with just one telegram.	Write only one device parameter Example: Reset totalizer
08	Diagnostics	Controller checks the communication connection to the measuring instrument. The following "Diagnostics codes" are supported: ▪ Sub-function 00 = return query data (loopback test) ▪ Sub-function 02 = return diagnostics register	
16	Write multiple registers	Controller writes a new value to several Modbus registers of the measuring instrument. A maximum of 120 consecutive registers can be written with one telegram. If the required device parameters are not available as a group, yet must nevertheless be addressed with a single telegram, use Modbus data map → 10	Write multiple device parameters Example: ▪ Mass flow unit ▪ Mass unit
23	Read/Write multiple registers	Controller reads and writes a maximum of 118 Modbus registers of the measuring instrument simultaneously with one telegram. Write access is executed **before** read access.	Write and read multiple device parameters Example: ▪ Read, e.g. the instantaneous value ▪ Write, e.g. reset totalizer
Modbus TCP only			
43/14	Read device identification	Controller reads device identification	Read multiple device parameters

Broadcast messages are only allowed with function codes 06, 16 and 23.

3.2 Response time

Response time of the measuring instrument to the request telegram of the Modbus master: Typically 3 to 5 ms

4 Appendix

4.1 Data types

4.1.1 Floating point numbers (IEEE-754)

Byte 3	Byte 2	Byte 1	Byte 0
SEEEEEEE	EMMMMMMM	MMMMMMMM	MMMMMMMM
S = sign, E = power of ten, M = mantissa			

4.1.2 Integer

Byte 1	Byte 0
Most Significant Byte (MSB)	Least Significant Byte (LSB)

4.1.3 Character string

Byte n	Byte n-1	...	Byte 1	Byte 0
Most Significant Byte (MSB)		...		Least Significant Byte (LSB)

Example

Register	Hex	Character	
40001	0x4C69	L	i
40002	0x7175	q	u
40003	0x696C	i	l
40004	0x696E	i	n
40005	0x6500	e	0

4.1.4 Other data types

Other data types with a data length of more than two are treated like character strings.

5 Register information

Parameter group	Parameter label	Register (1-based)	Data Type	Size (Bytes)	Access	Range
Advanced setup / Configuration backup	Actual diagnostics	20190	Uint16	2	ro	
Advanced setup / Totalizer	Process variable unit	4604	Enum16	2	rw	0...30 (mA)
Application / Totalizer	Preset value	2590	Float	4	rw	0 : Cancel 1 : Reset + totalize
Heartbeat Technology / Performing verification	Verification result	2355	Enum16	2	ro	250 : Not supported 2 : Passed 3 : Not done 254 : Not plugged 1 : Failed
Heartbeat Technology / Verification results	Verification ID	2315	Uint16	2	ro	0...65535
Measured values / Process variables	Temperature	2017	Float	4	ro	-1.4E+21...1.4E+21 (Celsius / Temperature unit)
Measured values / Process variables	Conductivity	2099	Float	4	ro	-1.4e+25...1.4e+25 (microSiemens_per_centiMeter / Conductivity unit)
Measured values / Process variables	Total solids	5089	Float	4	ro	
Measured values / Process variables	Corrected conductivity	3977	Float	4	ro	-1.4e+25...1.4e+25 (microSiemens_per_centiMeter / Conductivity unit)
Measured values / Process variables	Load rate	5091	Float	4	ro	-1.4E+21...1.4E+21 (kiloGram_per_Hour / Mass flow unit)
Measured values / Totalizer	Totalizer overflow	2612	Float	4	ro	105 : %TS 139 : ppm 97 : g/l 170 : mg/l 4 : kg/m³ 22 : mg/cm³ 12 : lb/gal (us) 11 : lb/ft³
Measured values / Totalizer	Totalizer value	2610	Float	4	ro	105 : %TS 139 : ppm 97 : g/l 170 : mg/l 4 : kg/m³ 22 : mg/cm³ 12 : lb/gal (us) 11 : lb/ft³
Operation / Totalizer handling	Reset totalizer	2609	Enum16	2	rw	0.0...12500.0 (Hertz)
Operation / Totalizer handling	Totalizer control	2608	Enum16	2	rw	0...3.4E+38 (Hertz)
Output values / Current output	Output current	5931, 5933, 5935	Float	4	ro	1 : Open 6 : Closed
Output values / Current output	Measured current	5779, 5781, 5783	Float	4	ro	0...65535
Output values / Pulse/frequency/ switch output	Output frequency	3462, 3464, 9910	Float	4	ro	97 : g/l 170 : mg/l 4 : kg/m³ 22 : mg/cm³ 12 : lb/gal (us) 11 : lb/ft³

Parameter group	Parameter label	Register (1-based)	Data Type	Size (Bytes)	Access	Range
Output values / Pulse/frequency/ switch output	Pulse output	3082, 3084, 4718	Float	4	ro	4 : kg/s 6 : kg/h 7 : kg/d 10 : t/h 11 : t/d 14 : oz/h 15 : oz/d 18 : lb/h 19 : lb/d 22 : STon/h 23 : STon/d
Output values / Relay output	Switch state	3518, 3519, 9875	Enum16	2	ro	0 : °C 2 : °F 1 : K 3 : °R
Setup / System units	Total solids unit	52093	Enum16	2	rw	16 : l/s 17 : l/min 18 : l/h 19 : l/d 4 : dm^3/s 5 : dm^3/min 6 : dm^3/h 7 : dm^3/d 8 : m^3/s 9 : m^3/min 10 : m^3/h 11 : m^3/d 44 : gal/s (us) 45 : gal/min (us) 46 : gal/h (us) 47 : gal/d (us) 48 : Mgal/s (us) 49 : Mgal/min (us) 50 : Mgal/h (us) 51 : Mgal/d (us) 88 : kgal/s (us) 89 : kgal/min (us) 90 : kgal/h (us) 91 : kgal/d (us)
Setup / System units	Density unit	2107	Enum16	2	rw	51 : kg 52 : t 53 : oz 54 : lb 55 : STon
Setup / System units	Mass flow unit	2101	Enum16	2	rw	10 : nS/cm 67 : µS/cm 7 : µS/m 9 : µS/mm 5 : mS/m 66 : mS/cm 4 : S/cm 3 : S/m
Setup / System units	Temperature unit	2109	Enum16	2	rw	-32000.0...32000.0
Setup / System units	Volume flow unit	2103	Enum16	2	rw	-3.4E+38...3.4E+38 (kiloGram / Process variable unit)
Setup / System units	Mass unit	2102	Enum16	2	rw	0 : Totalize 3 : Reset + hold 2 : Preset + hold 1 : Reset + totalize 4 : Preset + totalize 5 : Hold

Parameter group	Parameter label	Register (1-based)	Data Type	Size (Bytes)	Access	Range
Setup / System units	Conductivity unit	2121	Enum16	2	rw	-3.0e+38...3.0e+38 (kiloGram / Process variable unit)
Total solids adjustment / 2 - Enter lab value	Lab unit	52094	Enum16	2	rw	51 : kg 52 : t 53 : oz 54 : lb 55 : STon 251 : None

www.addresses.endress.com

Endress+Hauser

People for Process Automation