mPSU Medical Power Supply Units

MODBUS RTU Protocol Specification Document

Table of Contents

Revision Record

Changes to the previous version are marked in green.

Revision	Date	Description
1.0	07.05.2026	First release

1. Legal Information

Delta manuals undergo continuous revision in order to provide user with complete information regarding the installation and operation of this device. Therefore, before starting installation work, always consult Delta to check whether a newer version of the manual is available.

The information in this manual should be treated as confidential and no part of this manual may be reproduced without prior written permission from Delta. The information in this manual may not be used for any other purpose, directly connected to the use of this device.

All translations of this manual that are not authorized by Delta Energy System (Germany) must be marked with the words “Translation of the original operating instructions.”

Delta Energy Systems

Tscheulinstraße 21

79331 Teningen

Germany

2. Abbreviations and Acronyms

Abbreviation / Acronym	Definition
BMS	Battery Management System
CRC	Cyclic Redundancy Check
EEPROM	Electrically Erasable Programmable Read-Only Memory
EMCY	Emergency
FC	Function Code
LSB	Least Significant Byte
mPSU	Modular Power Supply Unit
MSB	Most Significant Byte
NMT	Network Management
PDU	Protocol Data Unit
Qty	Quantity
Qty Reg	Quantity of Registers
R/W	Read and Write access
RA	Register Address
RO	Read access Only
RTU	Remote Terminal Unit ()
SA	Starting Address
WO	Write access Only

Tab. 1: Abbreviations and acronyms

3. Introduction

3.1 Purpose

This document describes the MODBUS RTU protocol specification for the modular Power Supply Unit (mPSU). It defines the communication principles, supported function codes, register mapping, and usage guidelines for controlling and monitoring the mPSU via MODBUS RTU.

3.2 Scope

This document is intended for engineers and technical personnel who

- Have at least a basic knowledge of MODBUS RTU
- Have prior experience configuring and using MODBUS RTU devices
- Intend to integrate or interface the mPSU with their systems via MODBUS RTU

It is assumed that the necessary tools and equipment for MODBUS RTU communication are available and operating correctly.

This document is valid for the following mPSU:

1-phase mPSU:

- 1Ph 24V 150W
- 1Ph 24V 300W
- 1Ph 24V 600W
- 1Ph 24V 1000W
- 1Ph 48V 1000W
- 1Ph 48V 1500W

3-phase mPSU:

- 3Ph 24V 600W

NOTICE

Users must refer to the user manual for warnings, hazards and further information on the mPSU.

3.3 Overview

The mPSU MODBUS RTU interface provides a standardized communication channel for configuring and controlling.

Through this interface, users can

- Configure device parameters.
- Control the output voltage and output current.
- Monitor operating conditions via standard MODBUS RTU function codes.

To establish communication, the MODBUS RTU interface of the mPSU must be properly configured according to the parameters defined in this specification. The interface is designed to work with commonly available MODBUS RTU master tools or analyzers that support MODBUS RTU.

3.4 Supported MODBUS RTU Standards

The MODBUS RTU interface is implemented in compliance with

- MODBUS RTU over Serial Line Specification and Implementation Guide V1.02
- MODBUS RTU Application Protocol Specification V1.1b3

The implementation supports

- Standard MODBUS RTU frame format (Address, Function, Data, CRC)
- MODBUS RTU slave addressing in the range 0×00 to 0×0F (0 to 15 decimal)
- CRC-16 error checking (polynomial 0xA001)
- Configurable serial format
  - Baud rates: 9600, 19200, 38400, 57600, 115200 bps
  - Parity: None, Even, or Odd

Slave Address	Protocol Data Unit		CRC	
	Function Code	Data	2 bytes	
1 byte	1 byte	0 to 252 bytes	CRC Low	CRC High

Tab. 2: MODBUS RTU message frame structure

NOTICE

Addresses outside this range 0×00 - 0×0F (0 - 15 decimal) are not supported by the implementation.

3.5 References

Description	Location
MODBUS RTU Application Protocol Specification V1.1b3	www.modbus.org
MODBUS RTU over Serial Line Specification and Implementation Guide V1.02	

Tab. 3: References

4. Protocol Specification

This section explains the MODBUS RTU protocol for configuring and controlling the .

4.1 Supported Baud Rates and Parity Options

The mPSU must use the same baud rate and parity configuration as the MODBUS master.

Supported baud rates:

Baud rate [bps]	Supported by mPSU
115,200 (default)	Yes
57,600	Yes
38,400	Yes
19,200	Yes
9,600	Yes
Automatic baud rate detection	No

Tab. 4: Supported baud rates

Parity Options	Supported by mPSU
None (default)	Yes
Even	Yes
Odd	Yes

Tab. 5: Supported parity options

4.2 Supported MODBUS RTU Function Codes

4.2.1 Introduction

The mPSU supports a subset of the MODBUS RTU standard function codes. These function codes enable the device to read and write configuration and operational parameters through the MODBUS RTU interface. Unsupported function codes will return an exception response.

Function Code	Service Name	Supported by mPSU
03 (0×03)	Read Holding Registers	Yes
06 (0×06)	Write Single Register	Yes
16 (0×10)	Write Multiple Registers	Yes

Tab. 6: Supported MODBUS RTU function codes

4.2.2 Function Code 0×03 – Read Holding Registers

4.2.2.1 Introduction

This function code is used to read the contents of a contiguous block of holding registers in a remote device.

The request PDU specifies the starting register address and the number of registers. In the PDU, registers are addressed starting at zero. Therefore, registers numbered 1 to 16 are addressed as 0 to 15.

The register data in the response message are packed as two bytes per register, with the binary contents within each byte right-justified. For each register, the first byte contains the high-order bits and the second contains the low-order bits.

4.2.2.2 Request

Function code	1 byte	0×03
Byte count	2 bytes	0×0000 to 0xFFFF
Quantity of registers (N)	2 bytes	1 to 125 (0×7D)

Tab. 7: Function code 0×03 – request

4.2.2.3 Response

Function code	1 byte	0×03
Byte count	1 byte	2 x N [1)]
Register value	N[1)] x 2 bytes	–

1) N - Number of registers

Tab. 8: Function code 0×03 – response

4.2.2.4 Error Code

Error code	1 byte	0×83
Exception code	1 byte	01 or 02 or 03 or 04, see below

Tab. 9: Function code 0×03 – Error code 0×83

When processing a Read Holding Registers (0×03) request, the mPSU may return an exception response if the request parameters are invalid.

Exception code	Description
01	Illegal Function (function code not supported)
02	Illegal Data Address (invalid starting address or address range)
03	Illegal Data Value (invalid number of registers)
04	Server Device Failure (processing error)

Tab. 10: Function code 0×03 – Exception code for error code 0×83

4.2.2.5 Sample Request

Here is an example of a request to read registers 108 to 110:

Function Code		Service Name	
Field Name	**(Hex)**	**Field Name**	**(Hex)**
Function	03	Function	03
Starting Address Hi	00	Byte Count	06
Starting Address Lo	6B	Register Value Hi (108)	02
Number of registers Hi	00	Register Value Lo (108)	2B
Number of registers Lo	03	Register Value Hi (109)	00
		Register Value Lo (109)	00
		Register Value Hi (110)	00
		Register Value Lo (11)	64

Tab. 11: Function code 0×03 – sample Read Holding Registers request

The content of register 108 is shown in the two byte values of 02 2B hex (555 decimal). The contents of registers 109 and 110 are 00 00 and 00 64 hex (0 and 100 decimal), respectively.

4.2.2.6 State Diagram

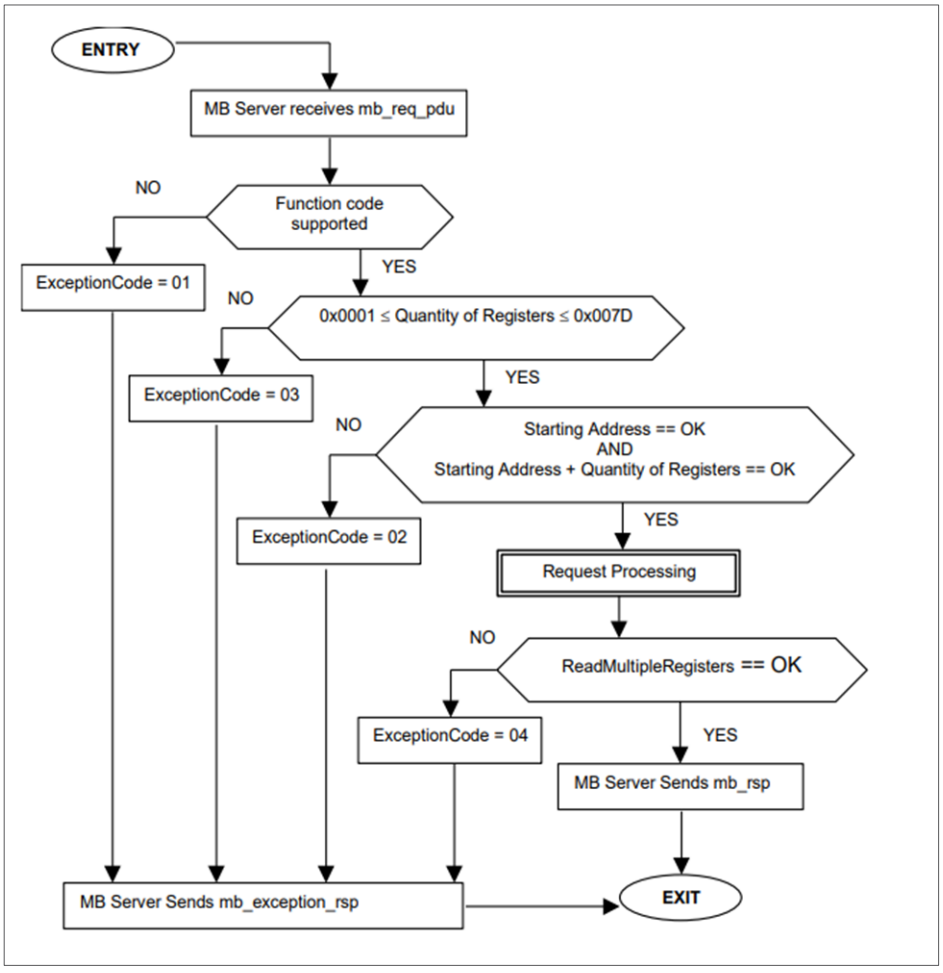

Fig. 1: State diagram for Read Holding Registers request

4.2.3 Function Code 0×06 – Write Single Register

4.2.3.1 Introduction

This function code is used to write a single holding register to the mPSU. The request specifies the address of the register and the data value to be written.

Registers are addressed starting at zero. Therefore, register number 1 is addressed as 0.

The normal response is an exact echo of the request which is returned after the contents of the register have been successfully written.

4.2.3.2 Request

Function code	1 byte	0×06
Register address	2 bytes	0×0000 to 0xFFFF
Register value	2 bytes	0×0000 to 0xFFFF

Tab. 12: Function code 0×06 – request

4.2.3.3 Response

Function code	1 byte	0×06
Register address	2 bytes	0×0000 to 0xFFFF
Register value	2 bytes	0×0000 to 0xFFFF

Tab. 13: Function code 0×06 – response

4.2.3.4 Error Code

Error code	1 byte	0×86
Exception code	1 byte	01 or 02 or 03 or 04, see below

Tab. 14: Function code 0×06 – error code 0×86

When processing a Write Single Register (0×06) request, the mPSU may return an exception response if the request parameters are invalid.

Exception code	Description
01	Illegal Function (function code not supported)
02	Illegal Data Address (invalid starting address or address range)
03	Illegal Data Value (invalid number of registers)
04	Server Device Failure (processing error)

Tab. 15: Function code 0×06 – Exception code for error code 0×86

4.2.3.5 Sample Request

Here is an example of a request to write register 2 to 00 03 hex:

Request		Response	
Field Name	**(Hex)**	**Field Name**	**(Hex)**
Function	06	Function	06
Starting Address Hi	00	Byte Count	00
Starting Address Lo	01	Register Value Hi (108)	01
Register Value Hi	00	Register Value Lo (108)	00
Register Value Lo	03	Register Value Hi (109)	03

Tab. 16: Function code 0×06 – sample Write Single Registers request

4.2.3.6 State Diagram

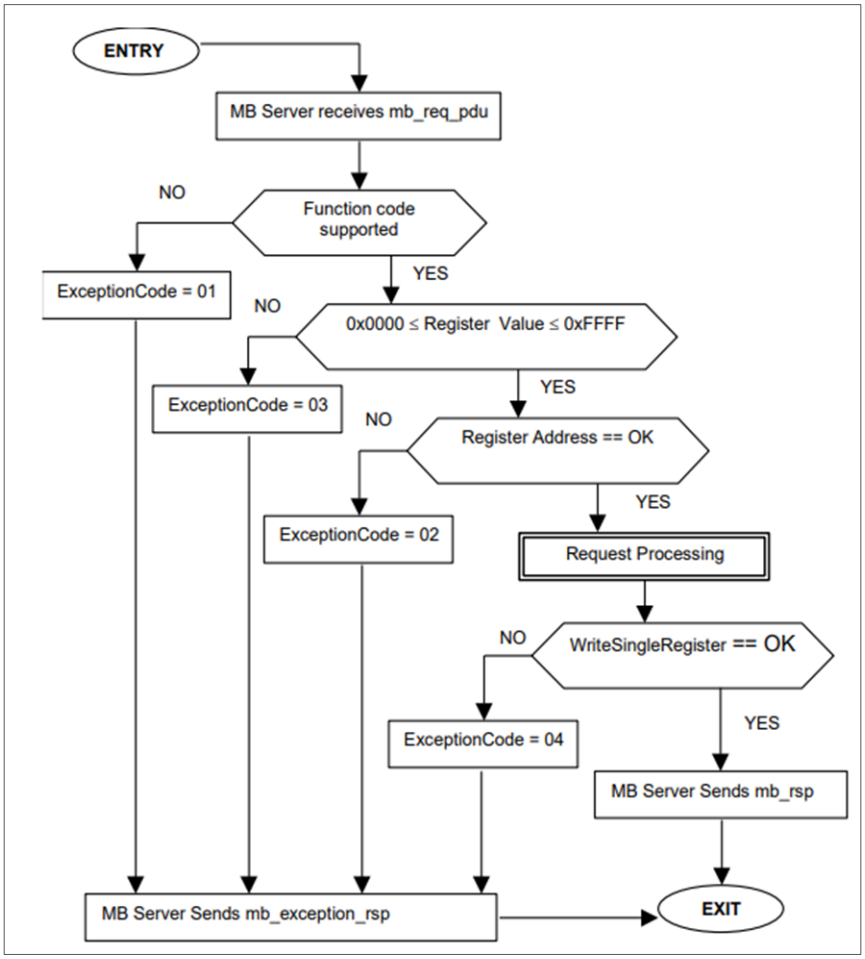

Fig. 2: State diagram for Write Single Register request

4.2.4 Function Code 0×10 – Write Multiple Registers

4.2.4.1 Introduction

This function code is used to write a block of contiguous registers (1 to 123 registers) in a remote device. The requested written values are specified in the request data field. Data is packed as two bytes per register. The normal response returns the function code, starting address, and quantity of registers written.

4.2.4.2 Request

Function code	1 byte	0×10
Starting address	2 bytes	0×0000 to 0xFFFF
Quantity of registers (N)	2 bytes	0×0001 to 0×007B
Byte count	1 byte	2 x N [1)]
Registers value	N[1)] x 2 bytes	Value

1) N - Number of registers

Tab. 17: Function code 0×10 – request

4.2.4.3 Response

Function code	1 byte	0×06
Starting address	2 bytes	0×0000 to 0xFFFF
Number of registers (N)	2 bytes	1 to 123 (0×7B)

Tab. 18: Function code 0×10 – response

4.2.4.4 Error Code

Error code	1 byte	0×90
Exception code	1 byte	01 or 02 or 03 or 04, see below

Tab. 19: Function code 0×10 – error code 0×90

When processing a Write Multiple Registers (0×10) request, the mPSU may return an exception response if the request parameters are invalid.

Exception code	Description
01	Illegal Function (function code not supported)
02	Illegal Data Address (invalid starting address or address range)
03	Illegal Data Value (invalid number of registers or byte count mismatch)
04	Server Device Failure (processing error)

Tab. 20: Function code 0×06 – exception code for error code 0×90

4.2.4.5 Sample Request

Here is an example of a request to write two registers starting at 2 to 00 0A and 01 02 hex:

Request		Response	
Field Name	**(Hex)**	**Field Name**	**(Hex)**
Function	10	Function	10
Starting Address Hi	00	Starting Address Hi	00
Starting Address Lo	01	Starting Address Lo	01
Number of Registers Hi	00	Number of Registers Hi	00
Number of Registers Lo	02	Number of Registers Lo	02
Byte Count	04		
Registers Value Hi	00		
Registers Value Lo	0A		
Registers Value Hi	01		
Registers Value Lo	02		

Tab. 21: Function code 0×10 – sample Write Multiple Registers request

4.2.4.6 State Diagram

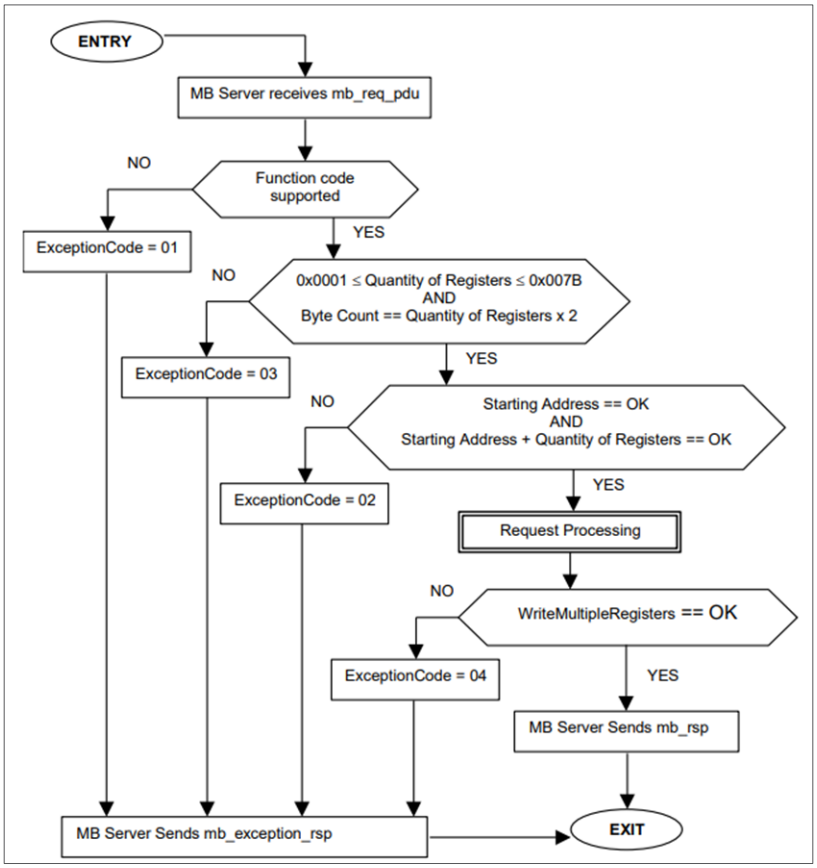

Fig. 3: State diagram for Write Multiple Registers request

4.3 Device Profile Compatibility

The mPSU provides the mandatory function codes as defined in the MODBUS RTU Application Protocol Specification V1.1b3, and implements a subset of optional function codes.

See “4.5 Register Dictionary”, p. 22 and “4.6 Register Data Descriptions”, p. 26 for details on the supported MODBUS RTU registers and function codes.

4.4 Register Access Details

4.4.1 Introduction

The MODBUS RTU interface is the communication channel for efficient data exchange of configuration and real-time monitoring data. All data access is performed exclusively through holding registers using the standard MODBUS RTU function codes.

There are two main types of register operations

- Read Holding Registers (0×03) used to read configuration and status data from the .
- Write Single Register (0×06) and Write Multiple Registers (0×10) used to write configuration parameters or control commands to the .

4.4.2 Read Holding Registers (Function Code 0×03)

The Read Holding Registers function allows the host system to cyclically poll the mPSU for operational status, measurements, and diagnostic information.

- The mPSU supports multiple defined register blocks for monitoring.
- Each read request must begin at a documented starting address.
- Requests to undefined addresses will be rejected with exception code 0×02 (Illegal Data Address).

4.4.3 Write Single Register (Function Code 0×06) and Write Multiple Registers (Function Code 0×10)

The two functions allow the host system to modify configuration parameters or send control commands to the .

- Function Code 0×06 supports writing to a single register.
- Function Code 0×10 supports writing to multiple consecutive registers.
- Safety-related parameters may require a confirmation or unlock procedure before modification.

NOTICE

For safe operation, ensure that the number of registers requested does not exceed the maximum supported by the mPSU. Oversized requests may result in an exception response.

4.4.4 MODBUS RTU Initialization

Upon delivery, the mPSU is preconfigured with factory default communication parameters as listed in “4.1 Supported Baud Rates and Parity Options”, p. 9.

Parameter	Default	Supported
Baud Rate	115,200 bps	9,600 to 115,200 bps
Data bits	8 bits	Fixed (cannot change)
Parity (1)	None	None, Even and Odd
Stop bits (2)	1 bits	Fixed (cannot change)
Slave Address (3)	0×01	0×00 to 0×0F (0 to 15 decimal)

(1)	Parity values are reported in MODBUS transactions as: 0 = None, 1 = Even, 2 = Odd.
(2)	Stop bits are reported with a factor of ×10. (e.g., 0.5 = 5, 1 = 10, 1.5 = 15, 2 = 20).
(3)	The default MODBUS RTU slave address is 0×01, configurable only via the rotary switch on the front panel.

During commissioning, the following initialization rules apply:

Slave address	Configurable only via the rotary switch located on the front panel
Baud rate / parity	May be reconfigured through the register dictionary, “4.5 Register Dictionary”, p. 22. Once a new value is written and the Save Device Configuration register (see “4.6.1.42 0×03ED – Save Device Configuration”, p. 31 is issued, the setting is permanently stored in EEPROM.
Hot change behavior	The mPSU does not require a restart after changing baud rate, data bits, parity, or stop bits. After the last active Modbus transaction completes, the UART interface is automatically re-initialized with the new configuration.

Tab. 22: MODBUS RTU initialization rules

The master device must resume communication using the updated parameters immediately after the change is applied.

NOTICE

For safe operation, each slave device must have a unique address on the RS485 bus. Duplicate slave addresses will cause communication conflicts.

Communication will fail if master/slave parameters do not match.

4.4.5 Control the via MODBUS RTU

4.4.5.1 Introduction

The mPSU can be controlled through MODBUS RTU by writing configuration and control parameters into specific registers using function code 0×10 (Write Multiple Registers).

- Control parameters such as output voltage and output current limit are represented as scaled integer values (e.g., value × 1000).
- Each parameter is written into two consecutive registers (Quantity = 2) to support 32-bit values.
- After a successful write, the acknowledges by echoing the function code, the starting address, and quantity of registers written.

4.4.5.2 Stand-alone Power Supply

When the mPSU operates as a stand-alone unit without cyclic communication from a host, a minimum configuration must be provided via MODBUS RTU registers:

- Set Output Voltage (Register: 0×0048, Function Code: 0×10, Quantity of registers = 2)
- Set Output Current Limit (Register: 0×004A, Function Code: 0×10, Quantity of registers = 2)

This ensures that the power supply can operate with predefined safe limits.

NOTICE

Each control value must be scaled according to the defined resolution, see “4.5 Register Dictionary”, p. 22.

4.4.5.3 **Example:** Set Output Voltage

Object Configuration	
Parameter	**Value**
Function Code	0×10
Register Address	0×0048
Quantity	2
Register Value	Set voltage * 1000

Tab. 23: Example for object configuration: Set Output Voltage

The following example shows a MODBUS RTU frame where the master writes Set Output Voltage 24.00V at address 0×0048 to slave address 0×01.

Register Value = 24.00 * 1000 mV = 24,000 mV = 0×00005DC0

Request		Response	
Field Name	**(Hex)**	**Field Name**	**(Hex)**
Slave Address	0×01	Slave Address	0×01
Function	0×10	Function	0×10
Starting Address Hi	0×00	Starting Address Hi	0×00
Starting Address Lo	0×48	Starting Address Lo	0×48
Quantity of Registers Hi	0×00	Quantity of Register Hi	0×00
Quantity of Registers Lo	0×02	Quantity of Register Lo	0×02
Byte Count	0×04	CRC Lo	0×00
Register Value Hi	0×00	CRC Hi	0×02
Register Value Lo	0×00		
Register Value Hi	0×5D		
Register Value Lo	0xC0		
CRC Lo	0xCE		
CRC Hi	0xF9		

Tab. 24: Sample MODBUS RTU frame for Set Output Voltage request

4.4.5.4 **Example:** Set Maximum Output Current

Object Configuration	
Parameter	**Value**
Function Code	0×10
Register Address	0×004A
Quantity	2
Register Value	Set current * 1000

Tab. 25: Example for object configuration: Set Maximum Output Current

The following example shows a MODBUS RTU frame where the master writes Set Maximum Output Current 24.00A at address 0×004A to slave address 0×01.

Register Value = 25.00 * 1000 mA = 25,000 mA = 0×000061A8

Request		Response	
Field Name	**(Hex)**	**Field Name**	**(Hex)**
Slave Address	0×01	Slave Address	0×01
Function	0×10	Function	0×10
Starting Address Hi	0×00	Starting Address Hi	0×00
Starting Address Lo	0×4A	Starting Address Lo	0×4A
Quantity of Registers Hi	0×00	Quantity of Register Hi	0×00
Quantity of Registers Lo	0×02	Quantity of Register Lo	0×02
Byte Count	0×04	CRC Lo	0×60
Register Value Hi	0×00	CRC Hi	0×1E
Register Value Lo	0×00		
Register Value Hi	0×61		
RegistersValue Lo	0xA8		
CRC Lo	0×5F		
CRC Hi	0xCE		

Tab. 26: Sample MODBUS RTU frame for Set Maximum Output Current request

4.5 Register Dictionary

The MODBUS RTU interface has an object dictionary, which is used for configuration and communication with the mPSU. An entry in the object dictionary is defined by a Function Code and a Starting Address.

4.5.1 Mapping for Function Code 0×03 – Read Holding Registers

4.5.1.1 Communication Segment

Function Code	Starting Address	Description	Quantity	Data Size (bytes)	Mode	Factor/ Unit	Data Type
0×03	0×03F6	Device Type	0002	4 Bytes	RO	None	Uint32
0×03	0×03F8	Device Vendor-ID	0002	4 Bytes	RO	None	Uint32
0×03	0×03FA	Device Product Code	0002	4 Bytes	RO	None	Uint32
0×03	0×03FC	Device Revision Code	0002	4 Bytes	RO	None	Uint32
0×03	0×03FE	Device Serial Number	0002	4 Bytes	RO	None	Uint32

Tab. 27: Mapping for Function Code 0×03 – Read Holding Registers – Communication Segment

4.5.2 Manufacturer Segment

Function Code	Starting Address	Description	Quantity	Data Size (bytes)	Mode	Factor/ Unit	Data Type
0×03	0×0001	Power good status	0001	2 Bytes	RO	None	Uint16
0×03	0×0002	Current limitation status	0001	2 Bytes	RO	None	Uint16
0×03	0×0003	Protection status	0002	4 Bytes	RO	None	Uint32
0×03	0×0005	Set fan speed	0001	2 Bytes	RO	%	Uint16
0×03	0×0006	Fan 1 speed value	0002	4 Bytes	RO	Hz	Uint32
0×03	0×0008	Fan 2 speed value	0002	4 Bytes	RO	Hz	Uint32
0×03	0×000A	Fan 1 status	0001	2 Bytes	RO	None	Uint16
0×03	0×000B	Fan 2 status	0001	2 Bytes	RO	None	Uint16
0×03	0×000C	Input voltage L1	0002	4 Bytes	RO	mV	Uint32
0×03	0×000E	Input voltage L2	0002	4 Bytes	RO	mV	Uint32
0×03	0×0010	Input voltage L3	0002	4 Bytes	RO	mV	Uint32
0×03	0×0012	Input power	0002	4 Bytes	RO	mW	Uint32
0×03	0×03E8	Baudrate	0002	4 Bytes	RO	bps	Uint32
0×03	0×03EA	Data bits	0001	2 Bytes	RO	Bits	Uint16
0×03	0×03EB	Parity	0001	2 Bytes	RO	Mode	Uint16
0×03	0×03EC	Stop bits	0001	2 Bytes	RO	0.1 Bits	Uint16
0×03	0×03ED	Save Device Configuration	0002	4 Bytes	RO	None	Uint32
0×03	0×03EF	Firmware version - Major	0001	2 Bytes	RO	None	Uint16
0×03	0×03F0	Firmware version - Minor	0001	2 Bytes	RO	None	Uint16
0×03	0×03F1	Firmware version - Revision	0001	2 Bytes	RO	None	Uint16
0×03	0×03F2	Hardware version - Major	0001	2 Bytes	RO	None	Uint16
0×03	0×03F3	Hardware version - Minor	0001	2 Bytes	RO	None	Uint16
0×03	0×03F4	Hardware version - Revision	0001	2 Bytes	RO	None	Uint16
0×03	0×03F5	Manufacturer	0001	2 Bytes	RO	None	Uint16
0×03	0×0040	Fan Start temperature	0001	2 Bytes	RO	°C	Uint16

Tab. 28: Mapping for Function Code 0×03 – Read Holding Registers – Manufacturer Segment

4.5.3 Device Profile Segment

Function Code	Starting Address	Description	Quantity	Data Size (bytes)	Mode	Factor/ Unit	Data Type
0×03	0×0014	Device status condition	0002	4 Bytes	RO	None	Uint32
0×03	0×0016	Device failure status condition	0002	4 Bytes	RO	None	Uint32
0×03	0×0018	Temperature actual values	0001	2 Bytes	RO	0.1 °C	Sint16
0×03	0×0019	Operation hours	0002	4 Bytes	RO	None	Uint32
0×03	0×001B	Input 1 status condition	0002	4 Bytes	RO	None	Uint32
0×03	0×001D	Input voltage	0002	4 Bytes	RO	mV	Sint32
0×03	0×001F	Input current	0002	4 Bytes	RO	mA	Sint32
0×03	0×0021	Input power	0002	4 Bytes	RO	mW	Sint32
0×03	0×0023	Link voltage	0002	4 Bytes	RO	mV	Sint32
0×03	0×0025	Ambient temperature	0001	2 Bytes	RO	0.1 °C	Sint16
0×03	0×0026	Output 1 – command ON/OFF	0001	2 Bytes	RO	None	Uint16
0×03	0×0027	Output 1 status condition	0002	4 Bytes	RO	None	Uint32
0×03	0×0029	Output 1 ranges – maximum voltage of device	0002	4 Bytes	RO	mV	Sint32
0×03	0×002B	Output 1 ranges – minimum voltage from device	0002	4 Bytes	RO	mV	Sint32
0×03	0×002D	Output 1 ranges – maximum current from device	0002	4 Bytes	RO	mA	Sint32
0×03	0×002F	Output 1 ranges – minimum current from device	0002	4 Bytes	RO	mA	Sint32
0×03	0×0031	Output 1 ranges – maximum power from device	0002	4 Bytes	RO	mW	Sint32
0×03	0×0033	Output 1 ranges – minimum power from device	0002	4 Bytes	RO	mW	Sint32
0×03	0×0035	Actual output voltage	0002	4 Bytes	RO	mV	Sint32
0×03	0×0037	Actual output current	0002	4 Bytes	RO	mA	Sint32
0×03	0×0039	Actual output power	0002	4 Bytes	RO	mW	Sint32
0×03	0×003B	Actual temperature (Thermal load)	0001	2 Bytes	RO	%	Sint16
0×03	0×003C	Set voltage	0002	4 Bytes	RO	mV	Sint32
0×03	0×003E	Set current	0002	4 Bytes	RO	mA	Sint32
0×03	0×0400	Self-start status	0001	2 Bytes	RO	None	Uint16

Tab. 29: Mapping for **Function Code 0×03 – Read Holding Registers** – Device Profile Segment

4.5.4 Mapping for Function Code 0×06 – Write Single Register

4.5.4.1 Communication Segment

The communication segment contains no entries.

Function Code	Starting Address	Description	Quantity	Data Size (bytes)	Mode	Factor/ Unit	Data Type
–	–	–	–	–	–	–	–

Tab. 30: Mapping for Function Code 0×10 – Write Multiple Registers – Communication Segment

4.5.5 Manufacturer Segment

Function Code	Starting Address	Description	Quantity	Data Size (bytes)	Mode	Factor/ Unit	Data Type
0×06	0×0005	Set fan speed	0001	2 bytes	WO	%	Uint16
0×06	0×0040	Set fan start temperature	0001	2 bytes	WO	°C	Uint16

Tab. 31: Mapping for Function Code 0×10 – Write Multiple Registers – Manufacturer Segment

4.5.6 Device Profile Segment

Function Code	Starting Address	Description	Quantity	Data Size (bytes)	Mode	Factor/ Unit	Data Type
0×06	0×0027	Output 1 - command ON/OFF	0001	2 bytes	WO	None	Uint16

Tab. 32: Mapping for Function Code 0×10 – Write Multiple Registers – Device Profile Segment

4.5.7 Mapping for Function Code 0×10 – Write Multiple Registers

4.5.7.1 Communication Segment

The communication segment contains no entries.

Function Code	Starting Address	Description	Quantity	Data Size (bytes)	Mode	Factor/ Unit	Data Type
–	–	–	–	–	–	–	–

Tab. 33: Mapping for Function Code 0×10 – Write Multiple Registers – Communication Segment

4.5.8 Manufacturer Segment

Function Code	Starting Address	Description	Quantity	Data Size (bytes)	Mode	Factor/ Unit	Data Type
0×10	0×03E8	Baud rate	0002	4 bytes	WO	bps	Uint32
0×10	0×03EB	Parity	0001	2 bytes	WO	Mode	Uint16
0×10	0×03ED	Save device configuration	0002	4 bytes	WO	None	Uint32

Tab. 34: Mapping for Function Code 0×10 – Write Multiple Registers – Manufacturer Segment

4.5.9 Device Profile Segment

Function Code	Starting Address	Description	Quantity	Data Size (bytes)	Mode	Factor/ Unit	Data Type
0×10	0×0048	Set voltage	0002	4 bytes	WO	mV	Sint32
0×10	0×004A	Set current	0002	4 bytes	WO	mA	Sint32

Tab. 35: Mapping for Function Code 0×10 – Write Multiple Registers – Device Profile Segment

4.6 Register Data Descriptions

4.6.1 Function Code 0×03 – Read Holding Registers

4.6.1.1 0×0001 – Power Good Status

Status of Power Good.

1	Output voltage out of range or input voltage failing for > 15ms
0	Output voltage in range and input voltage present

4.6.1.2 0×0002 - Current Limitation Status

Status of output current limitation.

1	Output current limited to “Set current” → value: Function Code: 0×10, Starting Address: 0×0048
0	Output current not limited

4.6.1.3 0×0003 - Protection Status

Bit-field of the Protection Status:

Protection Status		
Bit	**Group**	**Description**
0	Power Input	Undervoltage
1		Overvoltage
2		Reserved
3		Reserved
4		Reserved
5		Reserved
6		Reserved
7		Reserved
8	Power Output	Undervoltage
9		Overvoltage
10		Overcurrent
11		Reserved
12		Reserved
13		Reserved
14		Reserved
15		Reserved
16	Thermal	Fan 1 Failure
17		Fan 2 Failure
18		Overtemperature
19		Reserved
20		Reserved
21		Reserved
22		Reserved
23		Reserved
24	General	Internal Failure
25		Aux Output Overcurrent
26		Reserved
27		Reserved
28		Reserved
29		Reserved
30		Reserved
31		Reserved

4.6.1.4 0×0005 – Set Fan Speed

To set the fan speed. If it is lower than the internal calculation, the internal calculation is used. Only a higher setting has an effect.

Range	0 to 100%

4.6.1.5 0×0006 – Fan 1 Speed Value

Current speed of Fan 1. Unit: [Hz]

4.6.1.6 0×0008 – Fan 2 speed value

Current speed of Fan 2. Unit: [Hz]

4.6.1.7 0×000A – Fan 1 Status

The current status of Fan 1.

0	Ok
1	Error

4.6.1.8 0×000B – Fan 2 Status

The current status of Fan 1.

0	Ok
1	Error

4.6.1.9 0×000C – Input Voltage L1

Measured current value of Input Voltage L1.

4.6.1.10 0×000E – Input Voltage L2

Measured current value of input voltage L2.

4.6.1.11 0×0010 – Input Voltage L3

Measured current value of input voltage L3.

NOTICE

Registers Input Voltage L2 (0×000E) and Input Voltage L3 (0×0010) are valid only with 3-phase input. For 1-phase systems, these registers are not used and return 0xFFFF.

4.6.1.12 0×0012 - Input Power

Measured current value of input power.

4.6.1.13 0×0014 – Device Status Condition

The status of the output circuitry.

0×01	ON
0×00	OFF

4.6.1.14 0×0016 – Device Failure Status Condition

The output failure status.

0×01	Failure detected
0×00	No failure condition

4.6.1.15 0×0018 – Current Temperature Value

Measured current ambient temperature.

4.6.1.16 0×0019 – Operation Hours

Total accumulated number of operating hours of the device.

4.6.1.17 0×001B – Input 1 Status Condition

The status of the inhibit signal (INH_T).

0×01	ON
0×00	OFF

4.6.1.18 0×001D – Input Voltage

Measured current voltage at the input terminals.

4.6.1.19 0×001F – Input Current

Current input current value in real time.

4.6.1.20 0×0021 – Input Power

Measured current power at the input.

4.6.1.21 0×0023 – Link Voltage

The voltage level of the bulk capacitor.

4.6.1.22 0×0025 – Ambient Temperature

Measured current ambient temperature.

4.6.1.23 0×0026 – Output 1 – Command ON/OFF

Sets the inhibit enable command (INH_BUS).

0×01	ON
0×00	OFF

4.6.1.24 0×0027 – Output 1 Status Condition

Status of Output 1, including its current operational state.

4.6.1.25 0×0029 – Output 1 Ranges – Maximum Output Voltage

The maximum permissible output voltage of the device.

4.6.1.26 0×002B – Output 1 Ranges – Minimum Output Voltage

The minimum permissible output voltage of the device.

4.6.1.27 0×002D – Output 1 Ranges – Maximum Output Current

The maximum permissible output current of the device.

4.6.1.28 0×002F – Output 1 Ranges – Minimum Output Current

The minimum permissible output current of the device.

4.6.1.29 0×0031 – Output 1 Ranges – Maximum Output Power

The maximum permissible output power of the device.

4.6.1.30 0×0033 – Output 1 Ranges – Minimum Output Power

The minimum permissible output power of the device.

4.6.1.31 0×0035 – Current Output Voltage

The current output voltage in real-time.

4.6.1.32 0×0037 – Current output current

The current output current in real-time.

4.6.1.33 0×0039 – Current Output Power

The current output power in real-time.

4.6.1.34 0×003B – Current Temperature (Thermal Load)

In this register there are the value of the thermal load listed.

Range	0 to 130%

4.6.1.35 0×003C – Set Voltage

Defines the target output voltage setpoint for the device.

4.6.1.36 0×003E – Set Current

Defines the target output current setpoint for the device.

4.6.1.37 0×0040 - Fan Start Temperature

The currently set temperature threshold at which the fan starts.

Range	50 to 80 °C

4.6.1.38 0×03E8 – Baud Rate

Current baud rate set for the communication interface.

4.6.1.39 0×03EA – Data Bits

Current data bits (fixed 8 bits) for the communication interface.

4.6.1.40 0×03EB – Parity

Current parity setting for the communication interface.

4.6.1.41 0×03EC – Stop bits

Current stop bits setting for the communication interface.

4.6.1.42 0×03ED – Save Device Configuration

This register will be set to 0×0001 when writing to the Baud Rate (0×03E8) or Parity (0×03EB) registers by using function code 0×10. After writing to the Save Device Configuration register (0×03ED) by using function code 0×10, the Save Device Configuration register is reset to 0×0000, indicating that the save operation is complete.

Register will trigger alert 0×0001 when the configuration save is complete.

4.6.1.43 0×03EF – Firmware version – Major

Firmware version – used internally.

4.6.1.44 0×03F0 – Firmware version – Minor

Firmware version – used internally.

4.6.1.45 0×03F1 – Firmware version – Revision

Firmware version – used internally.

4.6.1.46 0×03F2 – Hardware version – Major

Hardware version – used internally.

4.6.1.47 0×03F3 – Hardware version – Minor

Hardware version – used internally.

4.6.1.48 0×03F4 – Hardware version – Revision

Hardware version – used internally.

4.6.1.49 0×03F5 – Manufacturer

Internally used.

4.6.1.50 0×3F6 – Device Type

Internally used.

4.6.1.51 0×3F8 – Device Vendor – ID

Internally used.

4.6.1.52 0×3FA – Product Code

Internally used.

4.6.1.53 0×03FC – Revision Number

Delta internal used.

4.6.1.54 0×03FE - Device Serial Number

Internally used.

4.6.2 Function Code 0×06 – Write Single Register

4.6.2.1 0×0005 – Set Fan Speed

Sets the target fan speed as a percentage (%).

Range	0 to 100%

4.6.2.2 0×0027 – Output 1 – Enable / Disable command

Enables / disables Output 1.

0×01	Enable Output 1
0×00	Disable Output 1

4.6.2.3 0×0040 – Set Fan Start Temperature

Sets the temperature threshold at which the fans start.

Defines the temperature threshold (in °C) at which the fan will start operating.

Range	50 to 80 °C

NOTICE

To permanently save this value, write to Save Device Configuration register (0×03ED), see “4.6.1.42 0×03ED – Save Device Configuration”, p. 31.

4.6.3 Function Code 0×10 – Write Multiple Registers

4.6.3.1 0×0048 – Set Output Voltage

Sets the output voltage setpoint for the device (see “4.4.4 MODBUS RTU Initialization”, p. 18).

4.6.3.2 0×004A – Set Output Current

Sets the output current setpoint for the device (see “4.4.4 MODBUS RTU Initialization”, p. 18).

4.6.3.3 0×03E8 – Set Baud Rate

Sets the communication baud rate for the device.

NOTICE

To permanently save this value, write to Save Device Configuration register (0×03ED), see “4.6.1.42 0×03ED – Save Device Configuration”, p. 31.

4.6.3.4 0×03EB – Parity

Sets the communication parity configuration.

4.6.3.5 0×03ED – Save Device Configuration

Example.

Function Code: 0×10

Starting Address: 0×03ED

Quantity of Registers: 0×02

Registers values: 0×65766173

Request		Response	
Field Name	**(Hex)**	**Field Name**	**(Hex)**
Slave Address	**0×01**	Slave Address	**0×01**
Function	**0×10**	Function	**0×10**
Starting Address Hi	**0×03**	Starting Address Hi	**0×00**
Starting Address Lo	**0xED**	Starting Address Lo	**0×48**
Quantity of Registers Hi	**0×00**	Quantity of Register Hi	**0×00**
Quantity of Registers Lo	**0×02**	Quantity of Register Lo	**0×02**
Byte Count	**0×04**	CRC Lo	**0×00**
Register Value Hi	**0×65**	CRC Hi	**0×02**
Register Value Lo	**0×76**		
Register Value Hi	**0×61**		
Register Value Lo	**0×73**		
CRC Lo	**0xCE**		
CRC Hi	**0xF9**		

Tab. 36: Sample MODBUS RTU frame for Save Device Configuration request

USA

Delta Electronics (Americas) Ltd.

Raleigh Office

5101 Davis Dr, Morrisville, NC 2756

U.S.A.

+1-919-767-3800

IM.sales@deltaww.com

www.delta-americas.com

Europe / Global

Delta Energy Systems (Germany) GmbH

Tscheulinstraße 21

79331 Teningen

Germany

+49-7641-455-0

IM.sales@deltaww.com

www.deltaww.com