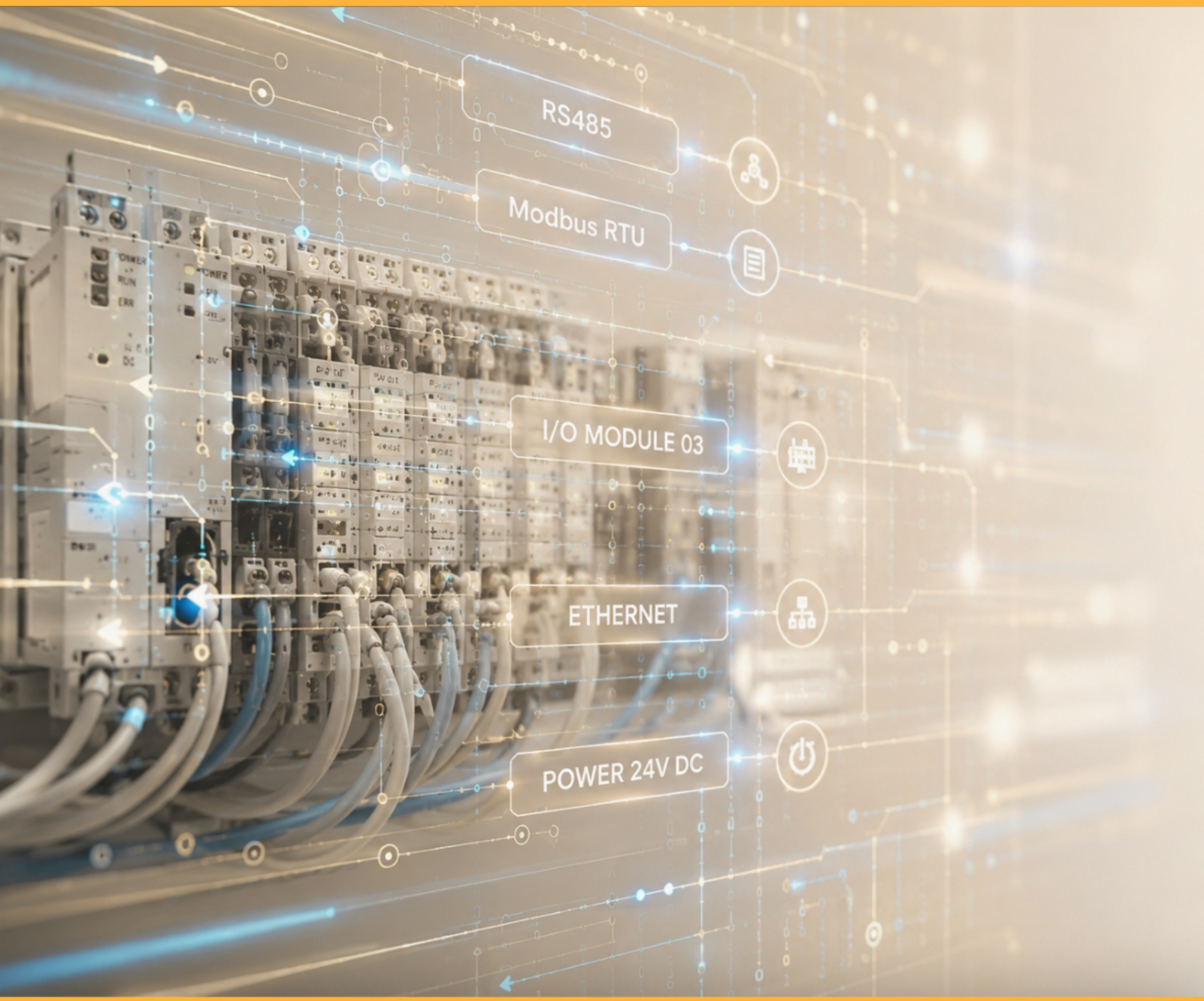
MD20003 Modbus RTU 12 SSR / 8 IN (EN)



Źródło: https://wiki.modig.pl/index.php?title=MD20003_Modbus_RTU_12_SSR_/_8_IN_(EN)

Rev. 20261024191854

Table of contents

Description 1
Basic features and parameters 1
Relay connectors 1
Input Connectors 2
RS485 Interface 2
Power Supply 3
Signaling LEDs 4
Restoring Settings 4
Configuration Registers (Holding Registers) 4
Measurement Input Registers 5
Modbus Standard Control Commands 5
List of Supported Modbus Commands 6
“Write Single Coil” Command - x05 6
“Write Multiple Coils” Command - x0F 8
“Read Coils” Command - x01 10
“Read Discrete Inputs” Command - x02 12
“Read Input Registers” Command - x04 14
“Read Holding Registers” Command - x03 16
“Write Single Holding Register” Command - x06 17
“Write Multiple Holding Registers” Command - x10 18
MODIG MODBUS APP 19
Downloading the MODIG MODBUS APP 20
USB-RS485 Converter 21
Communication Parameters 22
Launching the MODIG MODBUS APP 23

Description

Universal control card operated via RS485 and Modbus with 12 SSR relays and 8 inputs with optoisolation

The Modig Modbus RTU 12 SSR / 8 IN module is equipped with 12 solid-state relays (SSR) and 8 inputs with galvanic isolation. All outputs and inputs are controlled and read using standard Modbus commands transmitted via the RS485 interface. The control interface is equipped with galvanic isolation and surge protection, ensuring stable operation even in installations exposed to severe interference. The communication and operational parameters of the module can be flexibly configured and adapted to the requirements of various applications.

Basic features and parameters

- 12 SSR relay outputs - 100...240 V AC, 0.1...2 A, zero-cross switching
- total current of all outputs must not exceed 10 A
- 8 inputs with optoisolation - ON/OFF type inputs, active after connecting a voltage in the range of 5.5...25 V
- current consumption of optoisolated inputs - approx. 2...12 mA for voltages of 5.5...25 V
- RS485 interface with galvanic isolation and surge protection
- configurable communication parameters:
    - baud rate: 1200...1000000;
    - parity: none/even/odd;
    - stop bits: 1/1.5/2
- 5 V DC power supply (4.9...5.3 V DC) with a minimum capacity of 0.5 A via USB-C connector or DC 2.1/5.5 barrel jack
- 7...28 V power supply, min. 0.5 A via Phoenix MC 3.81 mm connector
- control and settings configuration via Modbus commands
- measurement of power supply voltage (on the 5V side) and module temperature
- dimensions: 200x110x60 mm
- mounting on DIN-35 rail

Relay connectors

> Relay circuits operate under mains voltage - nominally 230 V AC. This voltage is dangerous to human life, and all work must be carried out in compliance with safety regulations and under the supervision of a qualified professional.

The SSR relay outputs are routed to screw terminals with a 7.5 mm pitch. Each of the 12 outputs in the active state supplies an AC voltage with a maximum value of 240 V AC and a maximum current of 2 A. The total current of all active outputs must not exceed 10 A. In a situation where all outputs are active simultaneously, the current on each of them should not exceed 0.8 A (power approx. 180 W).

Solid-state relays (SSR) have specific properties that should be kept in mind:

- they operate only with alternating voltage (AC),
- they implement the zero-cross switching function (Zero Cross), thereby minimizing EMI generation and inrush currents,
- they are not suitable for switching inductive loads of medium and high power,
- they do not provide galvanic isolation - even in the off-state, they cause a small leakage current (approx. 2 mA) on the output,
- they may cause flickering of very low-power LED light bulbs.

The active state for a given output is indicated by the lighting of an LED located near the terminals of that connector. The layout of individual pinouts is shown in the figure.

One wire of the supply voltage for the output circuits must be connected to the "AC PWR IN" terminals. These are 4 terminals connected to each other; both wires of the output circuit supply voltage must not be connected there. An example circuit controlled by the Modig Modbus 12 SSR / 8 IN is shown in the figure.

Input Connectors

The Modig Modbus 12 SSR / 8 IN module is equipped with 8 inputs with optoisolation. A DC voltage ranging from 5.5...25 V with the correct polarity can be connected to the inputs, which will be read as an active state (logical 1). Current consumption in the active state is approx. 2...12 mA. The absence of voltage on the inputs will be read as an inactive state (logical 0).

Each input has '+' and '-' contacts routed to screw terminals with a 5.08 mm pitch. Connecting a voltage with the correct polarity will set the active state for the given input and cause the LED located near the terminals of that input to light up. The layout of the individual pinouts is shown in the figure.

RS485 Interface

The RS485 interface is brought out in the form of a Phoenix Contact MC 3.81 mm connector with 4 pins. Depending on the installation, only lines A+ and B- or all T, A+, B-, G can be used. Their functions are described in the table.

Designation	Function
120 Ω / T	Activation of the internal 120 Ω termination resistor. The resistor is active when the **T** contact is connected to the **A+** contact, e.g., via a jumper installed directly in the plug.

A+	Line **A** of the RS485 bus (positive potential)
B-	Line **B** of the RS485 bus (negative potential)
GND / G	Bus ground **GND** of the RS485 interface.

The RS485 interface is galvanically isolated from the other components and equipped with surge protection elements, ensuring stable operation even in installations exposed to severe interference.

RS485 communication parameters are configured via Modbus commands; the default settings are: 19200, 8, none, 1.

The Modig Modbus 12 SSR / 8 IN module can be connected to an RS485 bus that links up to 64 devices (including the module). Each device should be connected in such a way that no bus branches are formed - any branch connections should be as short as possible. It is recommended to connect 120 Ω termination resistors at both ends of the bus. The Modig Modbus has such a resistor integrated, but for it to be active, the A+ and T contacts must be connected together in the plug.

The figure shows a simplified schematic of the RS485 bus, terminated on one side with a 120 Ω termination resistor, and on the other side with the termination resistor integrated into the module. Furthermore, correct and incorrect device connections to the bus are shown.

Power Supply

The Modig Modbus 12 SSR / 8 IN module is equipped with 3 different power connectors. Only one of them must be selected, and power of the appropriate parameters and correct polarity must be connected. Detailed information can be found in the table and the figure.

Connector	Function
USB-C (J3)	Standard USB-C socket, serves only as a power supply connector. A power adapter/charger with a voltage of 5 V DC (4.9...5.3 V) and a current of min. 1 A must be connected.
2.1/5.5 (J2)	2.1/5.5 barrel jack compatible with "wall wart" power adapters. A power adapter/charger with a voltage of 5 V DC (4.9...5.3 V) and a current of min. 1 A must be connected. Mind the polarity - positive in the center.
MC 3.81 mm (J1)	Connector with a plug for terminal wire connection. A power source with a voltage in the range of 7...28 V DC and a current of min. 0.5 A must be connected. Mind the polarity - positive on the side of the inputs.

Signaling LEDs

There are two indicator LEDs on the Modig Modbus 12 SSR / 8 IN board that provide information about the device operation.

Indicator	Function
STA	LED lit with a short cyclic blanking indicates correct operation of the device
BUS	Flashing LED indicates receiving data from the RS485 bus and Modbus commands

---

Restoring Settings

There is a SET microbutton located on the Modig Modbus 12 SSR / 8 IN board. Pressing and holding it for approx. 5 seconds will cause all settings (values of configuration registers) to be restored to their default values. This function is useful in situations where, due to incorrect parameter configuration, it is impossible to send Modbus commands to the module. Restoring default settings will set the RS485 communication parameters to: 19200, 8, none, 1 and the module address to "1".

Configuration Registers (Holding Registers)

All parameters of the device are defined by the values of configuration registers, so-called Holding Registers. Registers can be read and written via Modbus commands, allowing the user to change the operation of the device. The list of registers and their functions are described in the table.

Register number (hex)	Name	Description	Default value / value range
x4000	Address	Specifies the module address when controlled by Modbus commands	1 (x0001) 1...255
x4002	Coils	Contains a value specifying the number of outputs (Coils) available in the module. Read-only value	8 (x0010)
x4004	Inputs	Contains a value specifying the number of inputs (Discrete inputs) available in the module. Read-only value	12 (x0000)
x4006	Baud	Specifies the RS485 communication speed (baud). Value without the last 2 zeros, e.g.: 1200 = 12 = x000C; 2400 = 24 = x0018 4800 = 48 = x0030; 9600 = 96 = x0060 19200 = 192 = x00C0; 57600 = 576 = x0240 115200 = 1152 = x0480; 256000 = 2560 = x0A00	192 (x00C0) 12...10000

x4008	Parity	Specifies the parity during RS485 communication. 0 - none; 1 - even; 2 - odd	0 (x0000) 0...2
x400A	Stop	Specifies the stop bits during RS485 communication. 1 - 1 stop bit; 2 - two stop bits	1 (x0001) 1...2
x400C	Broadcast	Defines whether the device executes commands with the broadcast address of 0 (x00). 0 - no, 1 - yes	1 (x0001) 0...1
x400E	CRC	Defines whether the device verifies the checksum in Modbus commands. 0 - no, 1 - yes	0 (x0000) 0...1
x4010	Response	Allows sending responses to Modbus commands. 0 - does not send any responses, 1 - sends responses to Modbus commands	1 (x0001) 0...1
x4012	IO Offset	Specifies from what value inputs and outputs (coils) are counted 0 - inputs/outputs start from 0 (0...15) 1 - inputs/outputs start from 1 (1...16)	1 (x0001) 0...1
x4014 or x8000	Version	Contains the firmware software version. value 10 - means 1.0 etc. Read-only value	10 (x00A0)
x2000	ExBaud	Allows setting communication parameters to one of the predefined values. The lower byte corresponds to the baud rate: 0 - 4800, 1 - 9600, 2 - 19200, 3 - 38400 4 - 57600, 5 - 115200, 6 - 12800, 7 - 256000 8 - other value The upper byte specifies the parity: 0 - none, 1 - even, 2 - odd	x0002 x0000...x0208

Measurement Input Registers

The device is equipped with measurement circuits for the power supply voltage and temperature, which allow monitoring the module's operating conditions and avoiding failures. The measurement results are available in the measurement input registers, so-called Input Registers. Measurement registers can be read via Modbus commands.

Register number (hex)	Name	Description	Value range
x3000	Voltage	Power supply voltage measurement value. Value 495 = 4.95 V etc.	330...550
x3002	Temp	Module temperature measurement. Temperature value 24 (x0018) means 24 degrees C. Temperature range: -20...-1 is stored as xFF14...xFF01 Temperature range: 0...70 is stored as x0000...x0046	-20...-1 *C = xFF14...xFF01 0...70 *C = x0000...0046

Modbus Standard Control Commands

Each command in the Modbus standard has the form of a data frame, the structure of which is shown in the figure:

Consecutive frames must be separated by an interval of at least 3.5 characters (the time required to transmit a full ASCII character), while the interval between the transmission of one character and the start of transmission of the next one within the same frame must not exceed 1.5 characters. These rules apply to communication speeds (baud rates) up to 19200.

If the communication speed is set above 19200, the minimum interval between frames has a constant value equal to 1.75 ms. Meanwhile, the interval between consecutive characters of the same frame must not exceed 0.75 ms.

Each device connected to the Modbus bus must have a unique address assigned. The address field is present in every transmitted frame, has a size of 1 byte, and can have a value in the range of 1...255. An address value of 0 is the so-called broadcast address and allows addressing the command to all devices connected to the bus. In some systems, it is not recommended to set addresses above the value of 247.

By default, outputs and inputs are numbered starting from 1. In this case, the first output is output number 1 (x01), and the last one is number 16 (0x10). The command descriptions and command examples refer specifically to this variant. However, the device can be configured so that outputs and inputs are numbered starting from 0. In that case, the first output is output number 0 (x00), and the last one is number 15 (0x0F). If the output number to which the control/read command refers exceeds the number of outputs contained in the device (15 or 16, depending on configuration), the command will not be executed and will remain unanswered.

There are several standard commands in the Modbus standard. Each of them, as well as example commands and device responses, are described in the subsequent sections of the manual.

List of Supported Modbus Commands

Command name	Command code	Function
Write Single Coil	**x05**	Allows controlling the state of an output or outputs
Write Multiple Coils	**x0F**	Allows controlling the state of multiple outputs simultaneously, performing both turn-on and turn-off actions for specific outputs.
Read Coils	**x01**	Allows reading the state of an output or outputs
Read Discrete Inputs	**x02**	Allows reading the state of inputs. Since the Modig Modbus 16R device does not have inputs for reading states, it reads the state of outputs in response to this command, similarly to the "Read Coils" command - x01.
Read Input Registers	**x04**	Allows reading values from measurement input registers
Read Holding Registers	**x03**	Allows reading values from configuration registers
Write Single Holding Register	**x06**	Allows setting a value in a configuration register
Write Multiple Holding Registers	**x10**	Allows setting values for several configuration registers

---

“Write Single Coil” Command - x05

The x05 command allows controlling the state of an output or outputs.

Data frame with x05 command		01 05 00 00 FF 00 8C 3A
Byte number	**Value (hex)**	**Meaning**
1	x01	Target device address
2	x05	“Write Single Coil” command - single output control
3...4	x0000	Number of the output to be controlled: 0 - none of the outputs, 1 - output 1 (first), 12 - output 12 (last) x00FF - controls all outputs - from 1 to 12
5...6	xFF00	Value specifying the task to be performed: x0000 - turn off (OFF), xFF00 - turn on (ON), x5500 - toggle the state to the opposite (TOGGLE) ON->OFF / OFF->ON
7...8	x8C3A	CRC16 checksum
Response		**A device that reads and executes the command will send an identical frame in response: 01 05 00 00 FF 00 8C 3A**

Similar commands:

- A5 05 00 08 55 00 6A 7C - x05 command to the device with address 165

(xA5), output no. 8 (x0008) will toggle its state (x5500)

Response: A5 05 00 08 55 00 6A 7C

- 14 05 00 10 00 00 9E CF - x05 command to the device with address 20

(x14), output no. 1 (x0001) will be turned off - OFF (x0000)

Response: 14 05 00 10 00 00 CE CF

- 29 05 00 FF FF 00 BA 22 - x05 command to the device with address 41

(x29), all outputs (x00FF) will be turned on - ON (xFF00)

Response: 29 05 00 FF FF 00 BA 22

“Write Multiple Coils” Command - x0F

The x0F command allows controlling the state of multiple outputs simultaneously, performing both turn-on and turn-off actions for specific outputs.

Data frame with x0F command		**01 0F 00 01 00 03 02 00 05 27 76**
Byte number	**Value (hex)**	**Meaning**
1	x01	Target device address
2	x0F	“Write Multiple Coils” command - multiple output control
3...4	x0001	Number of the output from which the control will start: 0 - none of the outputs will be controlled, 1 - from output 1 (first), 12 - from output 12 (last)
5...6	x0003	Number of outputs to be controlled: 0 - none of the outputs will be controlled 1 - one output, 12 - to control 12 outputs
7	x02	Number of data bytes containing the control sequence - must be 2
8...9	x0005	Control sequence, LSB -> output no. 1. Bits with a value of 0 cause outputs to turn off, bits with a value of 1 cause outputs to turn on. x0005 = 00000000 00000101 - outputs 1 and 3 are turned on
10...11	x27 76	CRC16 checksum
Response		**A device that reads and executes the command will send the following frame in response:** **01 0F 00 01 00 03 44 0A**
1	x01	Address of the responding device
2	x0F	“Write Multiple Coils” command - output control
3...4	x0001	Number of the output from which the control was started
5...6	x0003	Number of outputs that were actually controlled
7...8	x440A	CRC16 checksum

If the number of outputs to be set exceeds the number of outputs contained in the device (after considering the starting output number from which control begins), the command will be executed correctly for the existing outputs. The response will return the number of outputs actually controlled.

Similar commands:

- A5 0F 00 03 00 04 02 00 07 6C E6 - x0F command to the device with

address 165 (xA5), controls 4 (x0004) outputs starting from output no. 3 (x0003), the control sequence is: x0007 (00000000 00000111), which means it will turn on outputs 3, 4, 5 and turn off output 6.

Response: A5 0F 00 03 00 04 2C BD

- 14 0F 00 06 00 05 02 00 00 14 DA - x0F command to the device with

address 20 (x14), controls 5 (x0005) outputs starting from output no. 6 (x0006), the control sequence is: x0000 (00000000 00000000), which means it will turn off outputs 6, 7, 8, 9, 10.

Response: 14 0F 00 06 00 05 77 0C (5 outputs controlled)

- 29 0F 00 01 00 10 02 FF FF 1C 40 - x0F command to the device with

address 41 (x29), controls 16 (x0010) outputs starting from output no. 1 (x0001), the control sequence is: xFFFF (11111111 11111111), which means it will turn on all outputs from 1 to 12.

Response: 29 0F 00 01 00 0C 02 26

“Read Coils” Command - x01

The x01 command allows reading the state of an output or outputs.

Data frame with x01 command		**01 01 00 02 00 04 9C 09**
Byte number	**Value (hex)**	**Meaning**
1	x01	Target device address
2	x01	“Read Coils” command - reading output state
3...4	x0002	Number of the output from which the output state reading will start: 2 - reading from output no. 2
5...6	x0004	Value specifying how many outputs will be read: 4 - reading 4 outputs, namely: 2, 3, 4, and 5
7...8	x9C09	CRC16 checksum
Response		**A device that reads and executes the command will send the following frame in response: 01 01 02 00 0C B9 F9**
1	x01	Address of the responding device
2	x01	“Read Coils” command - reading output state
3	x02	Number of data bytes containing the response - 2
4...5	x000C	Output state: 2 - OFF (0), 3 - OFF (0), 4 - ON (1), 5 - ON (1) Binary: 00000000 00001100
6...7	xB9F9	CRC16 checksum

Similar commands:

- A5 01 00 06 00 02 44 EE - x01 command to the device with address 165

(xA5), reading the state of 2 (x0002) outputs, starting from output 6 (x0006).

Response: A5 01 02 00 02 49 E4 - only output no. 7 is turned on (x0002)

- 14 01 00 08 00 01 7E CD - x01 command to the device with address 20

(x14), reading the state of one (x0001) output, starting from output 8 (x0008).

Response: 14 01 02 00 00 B4 3F - output is turned off (x0000)

- 29 01 00 20 00 08 3A 2E - x01 command to the device with address 41

(x29), reading the state of 8 (x0008) outputs, starting from output 32 (x0020).

Response: no response - cannot read the state starting from output 32

If the number of outputs to be read exceeds the number of outputs contained in the device (after considering the starting output number from which the state reading begins), non-existent outputs in the response will always have the OFF (0) state.

“Read Discrete Inputs” Command - x02

The x02 command allows reading the state of inputs. Since the KAmod Modbus 16R device does not have inputs for reading states, it reads the state of outputs in response to this command, similarly to the "Read Coils" command - x01.

Data frame with x02 command		**01 02 00 02 00 04 D8 09**
Byte number	**Value (hex)**	**Meaning**
1	x01	Target device address
2	x02	“Read Discrete Inputs” command - reading input state
3...4	x0002	Number of the input from which the input state reading will start: 2 - reading from input no. 2
5...6	x0004	Value specifying how many inputs will be read: 4 - reading 4 inputs, namely: 2, 3, 4, and 5
7...8	xD809	CRC16 checksum
Response		**A device that reads and executes the command will send the following frame in response: 01 02 02 00 0C B9 BD**
1	x01	Address of the responding device
2	x02	“Read Discrete Inputs” command - reading input state
3	x02	Number of data bytes containing the response - 2
4...5	x000C	Input state: 2 - inactive (0), 3 - inactive (0), 4 - active (1), 5 - active (1); Binary: 00000000 00001100
6...7	xB9BD	CRC16 checksum

Similar commands:

- A5 02 00 02 00 06 40 EC - x02 command to the device with address 165

(xA5), reading the state of 6 (x0006) inputs, starting from input 2 (x0002).

Response: A5 02 02 00 02 49 A0 - only input no. 3 is active (x0002)

- 14 02 00 08 00 01 3A CD - x02 command to the device with address 20

(x14), reading the state of one (x0001) input, starting from input 8 (x0008).

Response: 14 02 02 00 00 B4 7B - input is inactive (x0000)

- 29 02 00 20 01 00 7E 78 - x02 command to the device with address 41

(x29), reading the state of 16 (x0010) inputs, starting from input 32 (x0020).

Response: no response - cannot read the state starting from input 32

If the number of inputs to be read exceeds the number of inputs contained in the device (after considering the starting input number from which the state reading begins), non-existent inputs in the response will always have the inactive (0) state.

“Read Input Registers” Command - x04

The x04 command allows reading values from measurement input registers. In the Modig Modbus device, it is possible to measure the power supply voltage on the 5 V input (it does not measure the voltage on the 7....28 V input) and the module temperature.

Data frame with x04 command		**01 04 30 00 00 02 7E CB**
Byte number	**Value (hex)**	**Meaning**
1	x01	Target device address
2	x04	“Read Input Registers” command - reading measurement input registers
3...4	x3000	Number of the register from which the read will start. x3000 - reading the power supply voltage measurement x3002 - reading the module temperature measurement
5...6	x0002	Value specifying how many registers will be read: 1 - one, e.g., voltage only; 2 - both measurement registers
7...8	x7ECB	CRC16 checksum
Response		**A device that reads and executes the command will send the following frame in response: 01 04 04 02 06 00 18 1A 37**
1	x01	Address of the responding device
2	x04	“Read Input Registers” command - reading measurement input registers
3	x04	Number of data bytes containing the response - 4
4...5	x0206	Voltage value - x0206 = 518, which means 5.18 V
6...7	x0018	Temperature value - x0018 = 24, which means 24 degrees C Temperature range: -20...-1 is stored as xFF14...xFF01 Temperature range: 0...70 is stored as x0000...x0046
8...9	x1A37	CRC16 checksum

Similar commands:

- A5 04 30 00 00 01 27 EE - x04 command to the device with address 165

(xA5), reading one register (x0001) number x3000, which is the power supply voltage measurement register.

Response: A5 04 02 01 F0 49 A0 - voltage is 4.96 V (x01F0 = 496)

- 14 04 30 08 00 02 FD CC - x04 command to the device with address 20

(x14), reading two (x0002) measurement registers starting from number x3008.

Response: no response - there is no measurement register numbered x3008.

- 29 04 30 02 00 02 7E 78 - x04 command to the device with address 41

(x29), reading two (x0002) measurement registers starting from number x3002.

Response: A5 04 02 00 18 C8 E3 - the device does not have measurement registers

numbered higher than x3002, therefore only one register numbered x3002 was read - module temperature = 24 (x0018) and there are only two data bytes (x02) in the response

---

“Read Holding Registers” Command - x03

The x03 command allows reading values from configuration registers.

Data frame with x03 command		**01 03 40 00 00 01 91 CA**
Byte number	**Value (hex)**	**Meaning**
1	x01	Target device address
2	x03	“Read Holding Registers” command - reading configuration registers
3...4	x4000	Number of the register from which the read will start. Registers in the range x4000...x4014 as well as x2000, x8000 are available
5...6	x0001	Value specifying how many registers will be read: 1 - one, etc. maximum 8
7...8	x91CA	CRC16 checksum
Response		**A device that reads and executes the command will send the following frame in response: 01 03 02 00 01 79 84**
1	x01	Address of the responding device
2	x03	“Read Holding Registers” command - reading configuration registers
3	x02	Number of data bytes containing the response - 2
4...5	x0001	Register numbered x4000 contains the address for Modbus commands, which in this case is 1 (x0001)
6...7	x7984	CRC16 checksum

Similar commands:

- A5 03 40 06 00 04 28 EE - x03 command to the device with address 165

(xA5), reading four configuration registers (x0004), starting from number (x4006).

Response: 01 03 08 00 C0 00 00 00 01 00 01 C5 DB - values of 4 configuration registers, i.e., 8 bytes of data

- 14 03 80 00 00 01 AF 0F - x03 command to the device with address 20

(x14), reading one (x0001) configuration register numbered x8000, which is the firmware version.

Response: 14 03 02 00 A0 B5 FF - response 2 bytes (x00A0) = 10 which means firmware version 1.0

“Write Single Holding Register” Command - x06

The x06 command allows setting a value in a configuration register.

Data frame with x06 command		**01 06 40 00 00 01 5D CA**
Byte number	**Value (hex)**	**Meaning**
1	x01	Target device address
2	x06	“Write Single Holding Register” command - writing a configuration register
3...4	x4000	Number of the register to which the new value will be written. Registers in the range x4000, x4006...x4012 as well as x2000 are available
5...6	x0001	Value to be written to the register
7...8	x5DCA	CRC16 checksum
Response		**A device that reads and executes the command will send an identical frame in response: 01 06 40 00 00 01 5D CA**

Similar commands:

- A5 06 40 00 00 04 84 ED - x06 command to the device with address 165

(xA5), writing configuration register numbered x4000 with value 4 (x0004), i.e., assigning a new address value of 4.

Response: A5 06 40 00 00 04 84 ED

- 14 06 40 0C 00 00 5E CC - x06 command to the device with address 20

(x14), writing configuration register numbered x400C with value 0 (x0000), i.e., disabling broadcast command handling.

Response: 14 06 40 0C 00 00 5E CC

“Write Multiple Holding Registers” Command - x10

The x10 command allows setting values for several configuration registers.

Data frame with x10 command		**01 10 40 0C 00 03 06 00 01 00 00 00 01 8F 7E**
Byte number	**Value (hex)**	**Meaning**
1	x01	Target device address
2	x10	“Write Multiple Holding Registers” command - writing several configuration registers
3...4	x400C	Number of the register from which the write will start. Registers in the range x4000, x4006...x4012 as well as x2000 are available
5...6	x0003	Value specifying how many registers will be written: 1 - one, etc. maximum 8
7	x06	Number of data bytes containing the values to be written
8...13	x0001 x0000 x0001	Three values to be entered into the configuration registers
14...17	x8F7E	CRC16 checksum
Response		**A device that reads and executes the command will send the following frame in response: 01 10 40 0C 00 03 55 CB**
1	x01	Address of the responding device
2	x10	“Write Multiple Holding Registers” command - writing several configuration registers
3	x400C	Number of the register from which the value writing was started
4...5	x0003	Value specifying how many registers were written
6...7	x55CB	CRC16 checksum

Similar commands:

- A5 10 40 00 00 01 02 00 03 6D 52 - x10 command to the device with

address 165 (xA5), writing one (x0001) configuration register starting from number x4000 with data of two bytes length (x02) equal to x0003, i.e., assigning a new address value of 3.

Response: A5 10 40 00 00 03 8C EC

- 14 10 40 0C 00 02 04 00 01 00 00 D6 F5 - x10 command to the device with

address 20 (x14), writing two (x0002) configuration registers starting from number x400C with values x0001 and x0000, i.e., enabling broadcast command handling and disabling CRC checksum verification.

Response: 14 10 40 0C 00 02 96 CE

MODIG MODBUS APP

The MODIG MODBUS APP allows you to control selected MD20xxx series modules connected to a PC via a USB-RS485 converter.

Downloading the MODIG MODBUS APP

The latest version of the MODIG MODBUS APP can be downloaded from the following link:

- Download MODIG MODBUS APP (MMAPP.exe)

The application's source code is available in the official GitHub repository:

- GitHub Repository: MODIG-CODE/Modig-Modbus-App

USB-RS485 Converter

The MODIG MODBUS APP requires a USB-RS485 converter of type:

MD11001 - https://www.modig.pl/produkt/konwerter-rs485-usb-iso/

or MD11002 - https://www.modig.pl/produkt/konwerter-rs485-usb-mini/

It can also work with other similar devices.

The converter must be connected to the computer via a USB interface and installed as a serial port, e.g., COM5:

The converter output (RS485 interface) must be connected to the MD20xxx series module(s) in accordance with the requirements for RS485/Modbus installations.

Communication Parameters

The MODIG MODBUS APP works with the following communication settings:

- Baud - 19200,
- Parity - none,
- Stop bits - 1.

The modules to be controlled by the application must have the same communication parameters set (these are the default values for the MD20xxx series modules).

Launching the MODIG MODBUS APP

The application will launch after clicking on the MMAPP.exe file. First, specify the serial port assigned to the USB-RS485 converter, e.g., COM5:

If the selected serial port is ready for operation, its name will change color — as shown in the figure:

Now, select the module type (the `MODULES` section on the left side of the window) that the application is to control:

Finally, set the module address (the `Modbus Address (Slave ID)` section on the right side of the window). After entering the correct Modbus address, confirm the selection by clicking the `Save` button.

After correctly configuring the module type and address, the main panel of the application will display active switches for changing output states (`OUTPUTS`) and indicators showing input states (`INPUTS`):

MODIG Sp. z o.o.
15-540 Białystok
ul. Żurawia 71
tel.: +48 798 063 314
e-mail: biuro@modig.pl
https://modig.pl

Zastrzegamy prawo do wprowadzania zmian bez uprzedzenia.
Oferowane przez nas płytki drukowane mogą się różnić od prezentowanej w dokumentacji, przy czym zmianom nie ulegają jej właściwości użytkowe.
MODIG Sp. z o.o. gwarantuje zgodność produktu ze specyfikacją.
MODIG Sp. z o.o. nie ponosi odpowiedzialności za jakiekolwiek szkody powstałe bezpośrednio lub pośrednio w wyniku użycia lub nieprawidłowego działania produktu.
MODIG Sp. z o.o. zastrzega sobie prawo do modyfikacji niniejszej dokumentacji bez uprzedzenia.