

OPERATIONAL RESTRICTIONS

ACS RELAY8 module should be operated in an environment, not beyond the limit values (see Specifications).

Manufacturer prohibits usage of module in applications that can endanger human life by their action or inaction.

ACS RELAY8 Module's manufacturer is not liable for consequences of exploitation of the equipment connected to module.

User must make sure there are no short circuits caused by wires connected to device.

It is strictly prohibited to plug or unplug wires those are under high voltage! It is strictly prohibited to disassemble module and touch electrical parts and/or violate the integrity of external electrical connections!

PURPUSE

Relay module ACS RELAY8 is designed for load control, and consists of eight independent channels.

SPECIFICATIONS

Control channels	4/8
Max. load per channel	0..2000 VA
Control interface type	RS-485
- Max. RS-485 network segment length	1200 meters
- type of devices identification inside network segment	addressing
- logical level protocol	MODBUS RTU
- Bus settings	19.2kbit/s, even parity, 1 stop bit
Power supply	+12 VDC
Max. consumption current	350 mA
Working temperature	+10..50 °C
Allowed humidity	5..80 %
Protection class, IEC-952	IP20
Weight	350 g
Dimensions	162 x 90 x 58 mm
Inputs / outputs	
- Low voltage wires	Max. 16AWG (1.5 mm ²)
- High voltage wires	Max. 12AWG (4 mm ²)

MODULE CONTROL TYPES

User can control channels in two ways:

1. Low-current push buttons. Principle of operation of low-voltage push-buttons: manual input channel is a TTL level (with +5 V and a current of several milliamperes), which expects a circuit to a common signal GND. Channel input is considered activated in a moment when button is pressed. At the same time, channel input is deactivated when button is released. So it is required to use pushbuttons without fixation and with normally open contacts
2. By MODBUS protocol via RS485 bus (19200 bit/s). Supported features *see Functional characteristics point protocol MODBUS RTU*.

MODULE OPERATIONAL MODES

The module supports two modes of operation:

1. asynchronous - all channels are controlled independently (for lighting):
 - Channel state is being switched to opposite by pressing corresponding button;
 - Channel is active while corresponding button is being held pressed
2. Pairwise - all channels are combined into a pairs of mutually exclusive relays (for control of blinds and rolling shutters). Channels are combined as following: 1-2, 3-4, 5-6, 7-8. Only one channel of the pair can be enabled at one time. Also, it will automatically turn off after a set period of time. This time period is set as a request parameter. Time can be set between 0..3600 seconds. (see "*Functional Specifications Protocol MODBUS RTU*", recording parameters in registers).

In order to turn off channel from particular pair, second channel should be activated. E.g., if channel #1 was activated, it can be deactivated by timeout or by activating the channel #2 (push corresponding push-button). In order to switch from channel #1 to channel #2, you must double-press button that corresponds to channel #2 (in case MODBUS protocol is used to control channels, then it is possible to switch channels directly from channel #1 to channel #2 without an intermediate step)

While operating using buttons it has no matter how long button is being pushed and held, the channel will be enabled for a certain period of time only. Also, the channel will not be deactivated when pressing the same button once again. If the button is held, but a request from the master is received to change the state of the channel, then this request will be processed despite the fact that the button pressed and is being held pressed by the user.

If there is no communication with the master via RS485 interface, channels continue to operate using control inputs (push-buttons). When module loses power, all channels will be switched OFF. Also, channels

states may be restored to previous states when power supply is attached and/or power is restored. See “*Module configuration*” (switches 2 and 3).

MODULE STATUS INDICATION

STATUS green LED can be in the following states:

- OFF – there is no power, or failure;
- flashes three times per second - module works, there is no communication via RS-485, illegitimate query received from master (error CRC16, malformed packet size);
- lights – module is OK, there is a connection via RS-485.

BUSY red LED indicates whether module is busy with some task:

- OFF – there is no power, or has power and module is idle;
- flashes three times per second - module is in busy state, performs the function.

If “activate/deactivate channel” request comes from master or inputs, the BUSY indicator flashes three times. If commands are continue to be coming (from inputs to the master), the indicator continues to flash until module will process last command.

In pairwise mode, BUSY LED blinks all the time channel is active (time interval).

MODULE CONFIGURATION

In order to determine module configuration, switch block should be used. It is located near STATUS and BUSY LEDs.

Switch #	Destination	Status	Comments
1	Channels mode	ON	Pairwise operational mode (4 channels)
		OFF	Asynchronous mode (8 channels)
2*	Setting channels state after power outage	ON	All channels will be turned ON (if switch 3 is ON, all channels will restore previous values)
		OFF	All channels will be turned OFF
3*	Restore	ON	If switch 2 is ON, then all channels will restore their previous values
		OFF	If switch 2 is OFF, then all channels will remain OFF in accordance to switch 2 description
4**	RS-485 communication presence	ON	If communication is not present for longer than 10 seconds, all channels will be turned OFF
		OFF	Does not affect channels state
5***	Channels operation mode	ON	When the input is activated, the channel state is reversed
		OFF	Channel state will change while input is active

* - only for asynchronous mode;

** - if channel is being currently controlled by button (input), then channel’s state will not change;

*** - does not affect the control via RS-485. Used only for asynchronous mode.

After changing the configuration, module needs to be restarted using RESET button.

MODULE ADDRESS SELECTION

In order to be able selecting/setting module address, top case plate needs to be removed (this will open access to address jumper-switches).

Jumper #	State/Value	Description
1	2^7	If jumper is "OFF" – corresponding bit is "0", "ON" – corresponding bit is "1"
2	2^6	
3	2^5	
4	2^4	
5	2^3	
6	2^2	
7	2^1	
8	2^0	

If address was changed, then you must reboot module using the RESET button.

Recommended Addresses

Address (decimal)	Address (binary)	Address (dec)	Address (bin)
16	00010000	24	00011000
17	00010001	25	00011001
18	00010010	26	00011010
19	00010011	27	00011011
20	00010100	28	00011100
21	00010101	29	00011101
22	00010110	30	00011110
23	00010111	31	00011111

MODULE STARTUP PROCEDURE

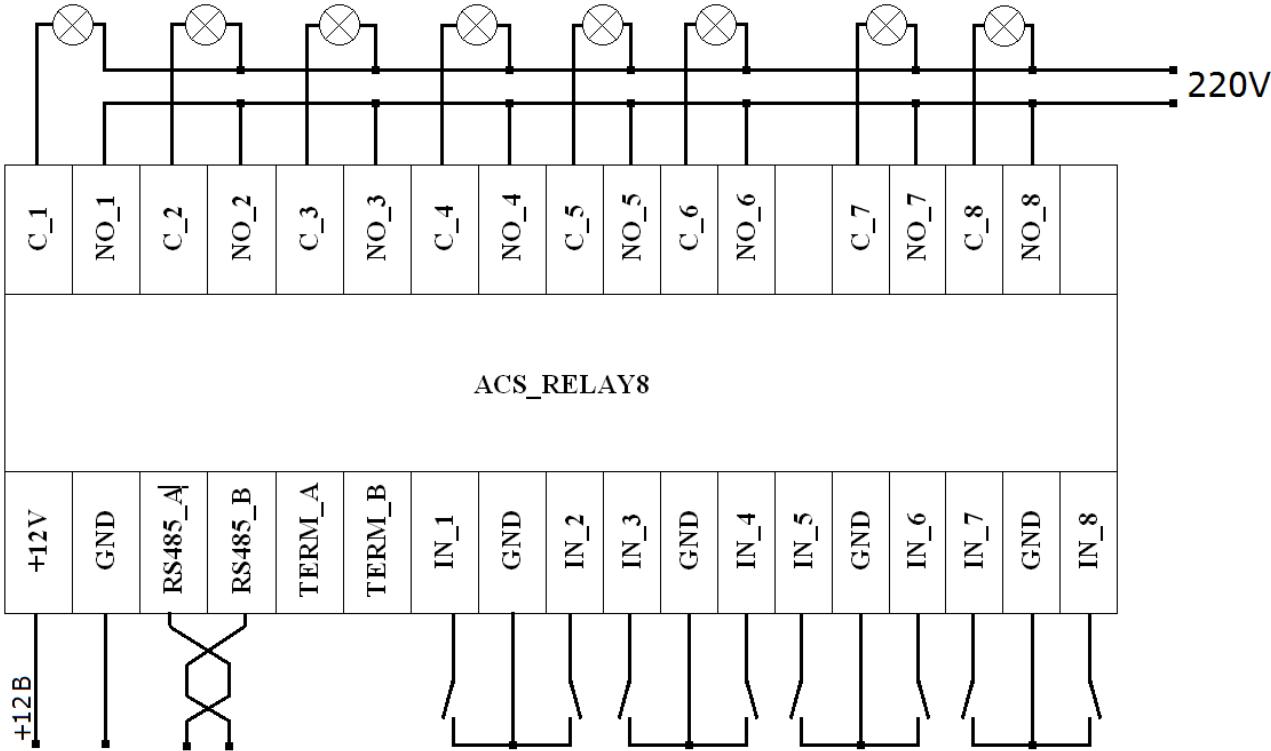
Module should be installed by qualified personnel only. Startup procedure includes:

- Module deployment/installation;
- Address and operational parameters configuration;
- Connecting external electrical circuits.

External electrical circuits/wires should be connected in the following order:

- Turn off high-voltage part;
- Check if there is a short circuit in the load circuit of the high-voltage module according to the scheme drawn up in advance;
- Connect low-voltage switches/buttons;
- Connect RS-485 interface and terminating resistor (if device is the last one on the bus) wires;
- Connect de-energized DC power supply (+12V, 400 mA);
- Turn ON power supply;
- Make sure corresponding LEDs are indicating correct module operational status;
- Turn ON high-voltage part.

TYPICAL CONNECTION SCHEMA



If relay module is connected to the end of RS-485 bus, then you need to connect terminal resistor 120ohm to TERM_A and TERM_B.

MODBUS RTU PROTOCOL FUNCTIONAL CHARACTERISTICS

Module implements MODBUS RTU protocol in order to be able communicating with master. RS-485 bus parameters: 19200 bits/s; 8 data bits, 1 stop bit, EVEN parity. Module supports the following functions:

Function #	Description	Channels operating modes
0x01	Read channels state	ALL
0x03	Read holding registers	ALL
0x05	Turn ON/OFF single channel	ALL
0x06	Write holding registers	ALL
0x0F	Turn ON/OFF multiple channels simultaneously*	Asynchronous mode
0x64	Read module unique ID	ALL

- **Read channels state**
"01 (0x01) Read Coils"

Request

Allowed start address - 0..7
Possible number of channels - 1..(8 – start address)

Response

Number of bytes - 1
Channels status - bit0..bit7

Bit #	Description
0	0 = channel #1 turned OFF
	1 = channel #1 turned ON
1	0 = channel #2 turned OFF
	1 = channel #2 turned ON

2	0 = channel #3 turned OFF
	1 = channel #3 turned ON
3	0 = channel #4 turned OFF
	1 = channel #4 turned ON
4	0 = channel #5 turned OFF
	1 = channel #5 turned ON
5	0 = channel #6 turned OFF
	1 = channel #6 turned ON
6	0 = channel #7 turned OFF
	1 = channel #7 turned ON
7	0 = channel #8 turned OFF
	1 = channel #8 turned ON

Request format: device address (0x01..0xF7) – function number (0x01) – high byte of channel number (0x00) – low byte of channel number (0x00..0x07) – high byte of number of channels to read (0x00) – low byte of number of channels to read (0x01..0x08) – two bytes of CRC16 MODBUS.

Example:

Request – 0x01 0x01 0x00 0x00 0x00 0x08 0x3D 0xCC;

Response – 0x01 0x01 0x01 0x81 0x91 0xE8.

Result – Channels №0 and №7 turned ON.

- **Turn single channel ON/OFF**

“05 (0x05) Write Single Coil”

Request

Allowed start address - 0..7
Possible values - 0x0000 – turn OFF; 0xFF00 – turn ON.

Response

Request echo

Request format: device address (0x01..0xF7) – function number (0x05) – high byte of channel number (0x00) – low byte of channel number (0x00..0x07) – Output value (0x0000 – turn OFF; 0xFF00 – turn ON) – two bytes of CRC16 MODBUS.

Example:

Request – 0x01 0x05 0x00 0x00 0xFF 0x00 0x8C 0x3A;

Response – 0x01 0x05 0x00 0x00 0xFF 0x00 0x8C 0x3A.

Result - Channel #1 turned ON.

- **Turn multiple channels ON/OFF**

“15 (0x0F) Write Multiple Coils”

Request

Allowed start address - 0..7
Possible number of channels - 1..(8 – start address)
Possible number of bytes - always ‘1’
Outputs’ values - 0 or 1 in one byte (bit0..bit7)

Response

Function code, start address, number of outputs

Request format: device address (0x01..0xF7) – function number (0x0F) – high byte of channel number (0x00) – low byte of channel number (0x00..0x07) – high byte of number of outputs (0x00) – low byte of number of outputs (0x01..0x08) – data length (0x01) – outputs values (0-7) (0x00..0xFF) – reserved (0x00) – two bytes of CRC16 MODBUS.

Пример:Request – 0x01 0x0F 0x00 0x00 0x00 0x08 0x01 0xAA 0x00 0x6A 0x20;

Response – 0x01 0x0F 0x00 0x00 0x00 0x08 0x54 0x0D.

Result – Channels #2, #4, #6, #8 are turned ON.

- **Reading holding registers**
“03 (0x03) Read Holding Registers”

Request

Allowed startaddress - 0..3
Possible number of registers - 1..(4 – start address)

Response

Function code, number of data bytes, values

Registers map (one register is 16 bits)	
Address	Description
0x0000	Module configuration, operational modes
0x0001	Pairwise mode timeout, in seconds
0x0002	Up time timer, high part
0x0003	Up time timer, low part

Request format: device address (0x01..0xF7) – function number (0x03) – high byte of register address (0x00) – low byte of register address (0x00..0x03) – high byte of number of registers (0x00) – low byte of number of registers (0x01..0x04) – two bytes of CRC16 MODBUS.

Example:

Request – 0x01 0x03 0x00 0x00 0x00 0x04 0x44 0x09;

Response – 0x01 0x03 0x08 0x00 0x09 0x00 0x3C 0x00 0x00 0x0A 0x51 0x9B 0x8E.

Result – device configuration 0x0009. Pairwise mode timeout = 0x003C (60 seconds). Up-timer = 0x0000A51 (2641 seconds).

- **Writing single holding register**
“06 (0x06) Write Single Registers”

Request

Allowed start address - 1
Possible values - 0x0000..0x0E10

Response

Request echo

Request format: device address (0x01..0xF7) – function number (0x06) – high byte of register address (0x00) – low byte of register address (0x01) – data (0x0000..0x0E10) – two bytes of CRC16 MODBUS.

Example:

Request – 0x01 0x06 0x00 0x01 0x0E 0x10 0xDD 0xA6;

Response – 0x01 0x06 0x00 0x01 0x0E 0x10 0xDD 0xA6.

Result – pairwise mode timeout is set to 0x0E10 (3600 seconds).

- **Module identification upon request**
“100 (0x64) Module Identify”

Request

Module address, function number, 0x00..0xFF, 0x00..0xFF, 0x00..0xFF, 0x00..0xFF, CRC16

Response

Module address, function number, unique ID (0x00000000..0xFFFFFFFF), CRC16

Request format: device address (0x01..0xF7) – function number (0x64) – any 4 bytes (0x00000000..0xFFFFFFFF) – two bytes of CRC16 MODBUS.

Example:

Request – 0x01 0x64 0x12 0x34 0x56 0x78 0x0B 0x36;

Response – 0x01 0x64 0x10 0xE8 0x0B 0x00 0xF3 0xC6.

Result – there is a module on address 0x01, it has the following unique ID: 0x10E80B00.

- **Possible error codes:**

Error response syntax: device address – function code + 0x80 – error code.

Error code	Description
0x01	Function is not supported
0x02	Incorrect address
0x03	Incorrect data
0x04	It is impossible to process request (collisions, incorrect parameter value)

COLLISIONS BETWEEN MANUAL CONTROL AND MASTER

Pairwise mode – if master will send request to set both channels from one pair to ON, then module will return error code 0x04. Outputs' state will not be changed.