

**MV210-212/214**

**Digital input module**

**User guide**

MV210-212/214\_3-EN-59115-1.8  
© All rights reserved  
Subject to technical changes and misprints

## Contents

<b>Warning notice system .....</b>	<b>3</b>
<b>Abbreviations .....</b>	<b>4</b>
<b>Safety.....</b>	<b>5</b>
<b>Introduction .....</b>	<b>6</b>
<b>1 Overview.....</b>	<b>7</b>
<b>2 Specifications.....</b>	<b>8</b>
2.1 Isolation of the Device Components.....	9
2.2 Environmental conditions.....	10
<b>3 Installation.....</b>	<b>11</b>
<b>4 Connection .....</b>	<b>13</b>
4.1 Connection recommendations.....	13
4.2 Connectors .....	13
4.3 Power supply .....	14
4.4 Electrical connection .....	15
4.4.1 Contact sensors .....	16
4.4.2 PNP and NPN transistors.....	16
4.4.3 PNP and NPN encoders .....	17
4.5 Ethernet connection .....	17
<b>5 Operation.....</b>	<b>19</b>
5.1 Operation principle .....	19
5.2 Indication and control .....	19
5.3 Real time clock.....	20
5.4 Log file .....	20
5.5 Data exchange modes.....	22
5.5.1 Modbus TCP communication .....	22
5.5.2 Error codes for Modbus protocol.....	32
5.6 Digital inputs operation modes .....	34
5.6.1 Logic level detection .....	35
5.6.2 Counting of high-frequency pulses.....	35
5.6.3 Frequency measurement .....	35
5.6.4 Encoder signal processing .....	35
<b>6 Configuration.....</b>	<b>36</b>
6.1 Connection to akYtec Tool Pro .....	36
6.2 Configuring Network Settings .....	37
6.3 Module access password.....	38
6.4 Firmware update .....	38
6.5 Real-time clock setting.....	39
6.6 Restore default settings .....	39
6.7 Counter reset.....	39
<b>7 Maintenance .....</b>	<b>40</b>
7.1 General Instructions .....	40
7.2 Battery replacement .....	40
<b>8 Transportation and storage .....</b>	<b>43</b>
<b>9 Scope of delivery .....</b>	<b>44</b>

**Appendix A Encrypting the log file .....45**

### Warning notice system

Explanation of the symbols and keywords used:



**WARNING**

WARNING indicates a **potentially dangerous situation** that could result in death or serious injuries.



**CAUTION**

CAUTION indicates a **potentially dangerous situation** that could result in minor injuries.



**NOTICE**

NOTICE indicates a **potentially dangerous situation** that could result in damage to property.



**NOTE**

NOTE indicates helpful tips and recommendations, as well as information for efficient and trouble-free operation.



### Abbreviations

- **PC** – personal computer.
- **PLC** – programmable logic controller.
- **RTC** – real time clock.
- **USB** – an industry standard that establishes specifications for cables, connectors and protocols for connection, communication and power supply between personal computers and their peripheral devices.
- **UTC** – coordinated universal time.

### Safety

Read this manual carefully before installing, operating or servicing the device.

The device has been designed and built solely for the intended use described in this guide, and can only be used accordingly. The technical specifications contained in this guide must be observed.

Any other use is considered improper.

**CAUTION**

This device must not be used for medical devices which receive, control or otherwise affect human life or physical health.

When installing and using this product, all applicable state, federal and local regulations must be observed.

Based on safety considerations and compliance with the data provided in the documents, the repair of the components of the products is carried out exclusively by the manufacturer.

There is a dangerous voltage on the terminal block. Any connections to the device and maintenance operations must be carried out only when the power of the device is turned off.

The device's method of protection against electric shock meets the terms of class II, IEC 61131-2-2012.

The device installation must be performed in particularized equipment cabinet, the inner access to which is allowed only to approved specialists.

**CAUTION**

It is forbidden to use the device in corrosive environments with acids, alkalis, oils, etc. in the atmosphere.

Obey all the necessary rules and instructions when using programmable logic controllers in areas where technical safety requirements apply.

Failure to obey these warnings could result in personal injury or equipment damage.

Akytec company shall not be liable for technical or editorial errors or omissions contained in this document.

### Introduction

This document provides detailed information about the operation principle, design, configuration, installation and maintenance of the digital input modules MV210-212 and MV210-214, hereinafter referred to as the Device or Module.

Connection, adjustment and maintenance of the device must be carried out only by qualified personnel after reading this operating manual.

Device designation when ordering: **MV210-212** and **MV210-214**.

### 1 Overview

The module is designed to collect data from the built-in digital inputs and transfer them via the Ethernet network to the PLC, panel controllers, computers or other devices.

The module has 32 digital inputs for connecting sensors or signals of the following types:

- For **MV210-212**: contact sensor, npn and pnp transistors.
- For **MV210-214**: dry contact sensor, npn transistor.

The Module is intended for use in industry and agriculture.

## 2 Specifications

Table 2.1 Specifications

Parameter	Value
<b>Power supply</b>	
Power supply	24 (10...48) V DC
Power consumption	9 W
Protection against reverse polarity	Yes
Digital input hysteresis	0.5 V
<b>Interfaces</b>	
Data transfer interface	Dual Port Ethernet 10/100 Mbit
Configuration interface	USB 2.0 (MicroUSB), Ethernet 10/100 Mbit
Data transfer protocol	Modbus TCP
Protocol version	IPv4
<b>Digital inputs of MV210-212</b>	
Inputs quantity	32
Signal type	<ul style="list-style-type: none"> <li>– contact sensor (external power supply required);</li> <li>– NPN / PNP transistors</li> </ul>
Operation modes	<ul style="list-style-type: none"> <li>– logical level definition;</li> <li>– counting the number of high-frequency pulses (only DI1-DI8);</li> <li>– frequency measurement (only DI1-DI8);</li> <li>– encoder signal processing (up to 3 encoders)</li> </ul>
Minimum pulse duration perceived by the digital input (only DI1-DI8)	5 $\mu$ s (up to 100 kHz)
Minimum duration of a single pulse (for inputs DI9-DI32)	1 ms (up to 400 Hz)
Current of logical 0	1.2 mA
Current of logical 1	min. 5.5 mA
Voltage range of logical 0	0 ... 6.1 V
Voltage range of logical 1	8.8 ... 30 V
<b>Digital inputs of MV210-214</b>	
Inputs quantity	32
Signal type	<ul style="list-style-type: none"> <li>– dry contact;</li> <li>– NPN transistor</li> </ul>
Operation modes	<ul style="list-style-type: none"> <li>– logical level detection;</li> <li>– pulse count</li> </ul>
Minimum duration of a single pulse	1 ms
Resistance of contacts (switch) and connecting wires connected to a discrete input, max.	100 $\Omega$
<b>Flash-memory (log-file)</b>	

Parameter	Value	
Number of write and erase cycles	Up to 100000	
Maximum log file size	2 KB	
Maximum number of log files	1000	
Minimum log writing interval	10 seconds	
<b>Real time clock</b>		
Inaccuracy	at +25 °C	max. 3 s per day
	at -40 °C and +55 °C	max. 10 s per day
Battery type	CR2032	
Lifetime of one battery	6 years	
<b>General specifications</b>		
Dimensions	82 × 124 × 83 mm	
Ingress protection rating	IP20	
Error-free running time*	60.000 h	
Average service life time	10 years	
Weight	0.4 kg	

\* Except real-time clock battery

### 2.1 Isolation of the Device Components

The circuit of galvanic isolated components and the strength of galvanic isolation are shown in [Figure 2.1](#).

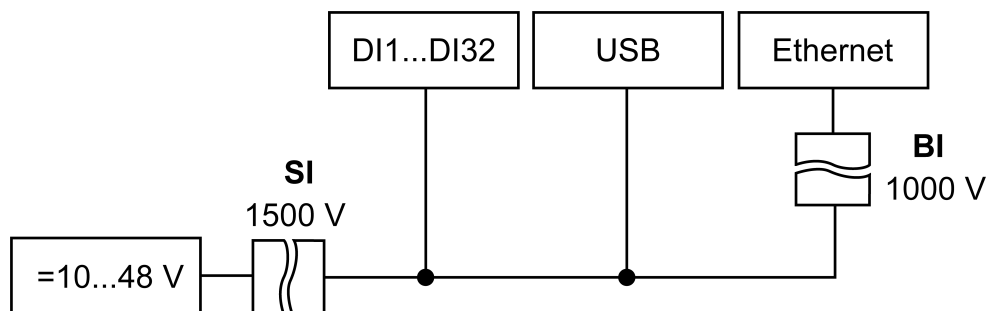


Fig. 2.1 Isolation of the Device Components

Table 2.2 Insulation types

Type	Description
Basic insulation (BI)	The insulation used for active parts of equipment to provide protection against electric shock. The electrical strength of the basic insulation is checked by type tests: by applying a test AC voltage, the value of which is different for different circuits of the Device
Supplementary insulation (SI)	Independent insulation used in addition to the basic insulation in order to guarantee protection against electric shock in the event of a failure of the main insulation. The electrical strength of the supplementary insulation is checked by type tests: by applying an alternating test AC voltage of various levels (RMS value)

**NOTICE**

The value of the insulation strength is indicated for testing under normal climatic conditions, the exposure time is 1 minute according to IEC 61131-2.

### 2.2 Environmental conditions

The Module meets the requirements for immunity to interference in accordance with IEC 61000-6-4:2006. According to the level of emission of radio interference (noise emissions) the Device complies with the standards established for equipment of class A by CISPR 22-97. The Device is designed for operation in the following conditions:

- ambient temperature:  $-40 \dots +55 \text{ }^{\circ}\text{C}$ ;
- relative humidity: up to 95% (at  $+35 \text{ }^{\circ}\text{C}$ , non-condensing);
- closed non-hazardous areas, free of corrosive or flammable gases.
- permitted pollution degree 1 according to IEC 61131-2.

The resistance to mechanical influences during operation of Device is in accordance with IEC 61131-2.

The resistance to climatic influences during the operation of Device is in accordance with IEC 61131-2.

Module is class II equipment in accordance with IEC 61131-2.

## 3 Installation

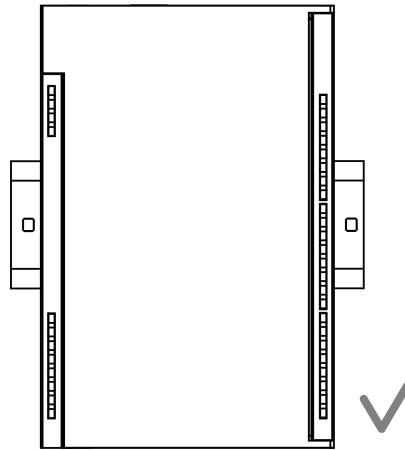


Fig. 3.1 Proper installation

The device is installed in the electrical cabinet. The design of the cabinet should protect the device from moisture, dirt or foreign objects.

To install the device, perform the following actions:

1. Make sure that there is enough free space to connect the wire harness.
2. Fasten the device to the DIN rail or to a vertical surface using screws (see [Figure 3.1](#)).

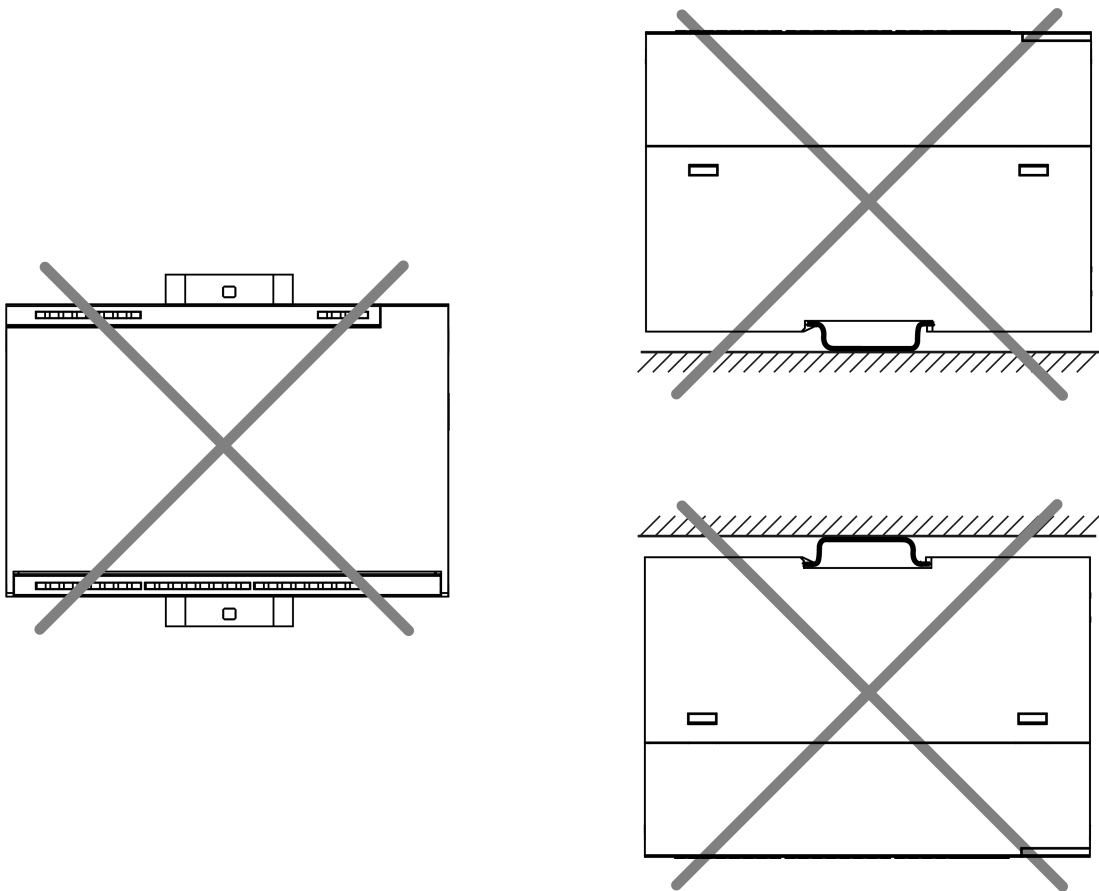


Fig. 3.2 Improper installation

**CAUTION**

During installation, a free space of 50 mm above and below the module is required.



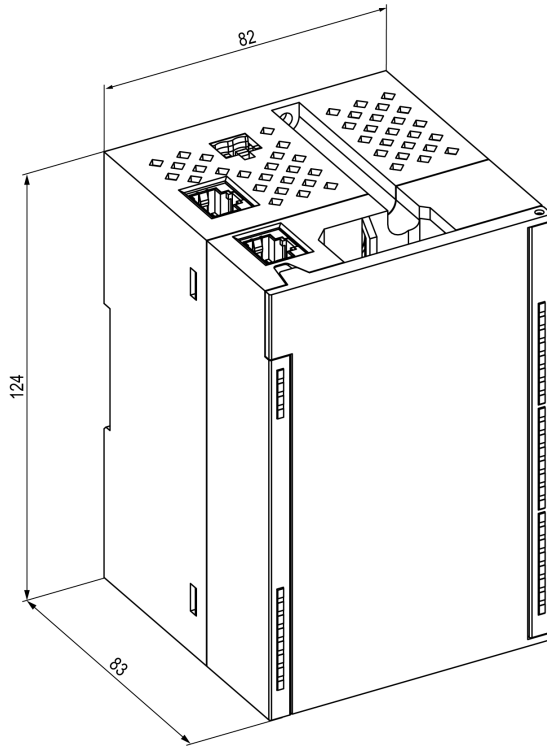


Fig. 3.3 Dimensions

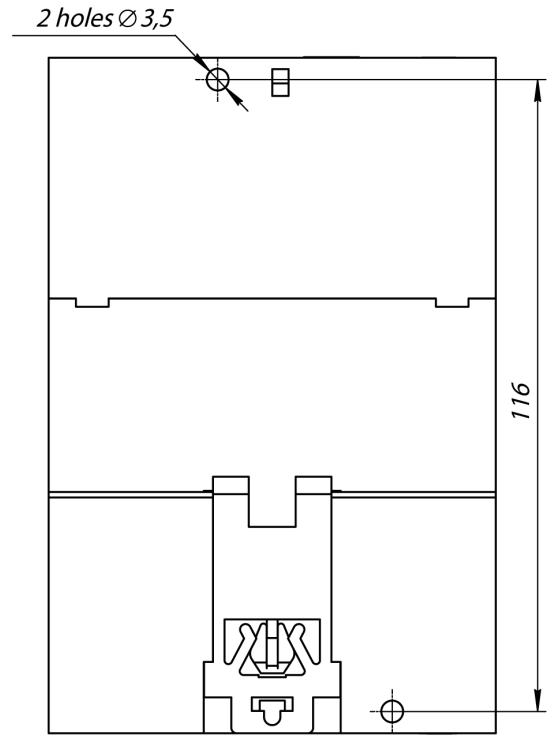


Fig. 3.4 Mounting dimensions

## 4 Connection

### 4 Connection

#### 4.1 Connection recommendations

Mounting of external connections must be carried out by a wire with a cross section of not more than 0.75 mm<sup>2</sup>.

For stranded wires, use end sleeves.

After mounting, put the wires into the cable channel of Module housing and close the cover.

If necessary, remove the terminal blocks of Module, loosen the two screws at the corners of the terminal blocks.

The power wires must be mounted using the supplied return terminal block.

**CAUTION**

Connection and maintenance is performed only when power of Module and devices connected to it is turned off.

**CAUTION**

Do not connect wires of different cross-sections to one terminal.

**CAUTION**

Do not connect more than two wires to one terminal.

#### 4.2 Connectors

The connectors of the interfaces and the power supply of the device are shown in the figure below.

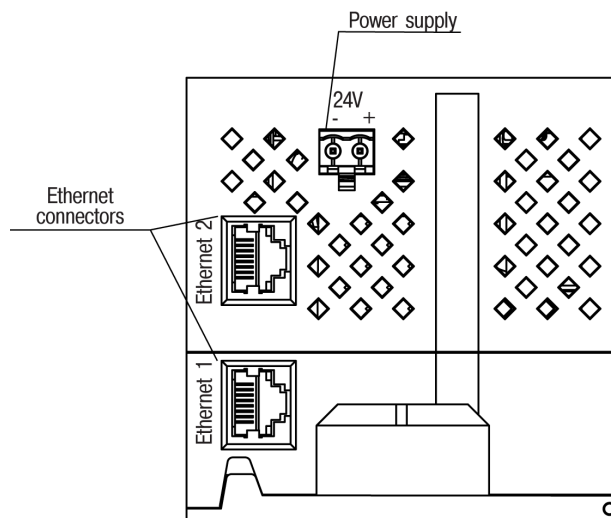


Fig. 4.1 The device's connectors

### 4.3 Power supply

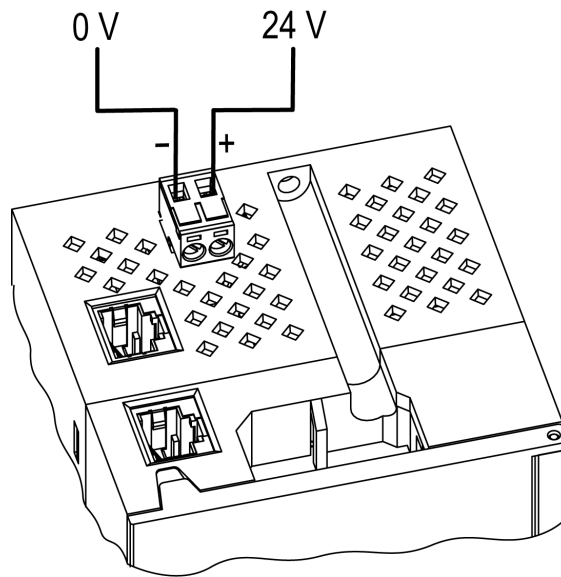


Fig. 4.2 Power supply contacts function



**CAUTION**

Using power supplies without potential isolation or with basic isolation of low voltage circuits from AC lines can lead to dangerous voltage in the circuits.

4.4 Electrical connection

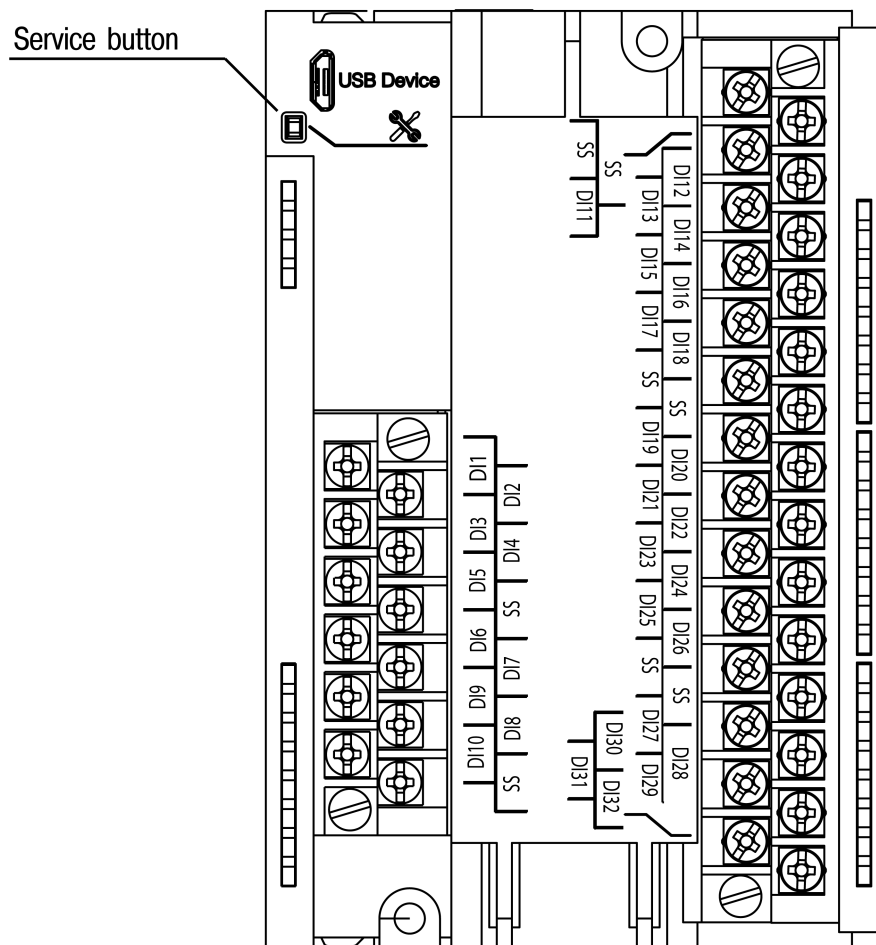


Fig. 4.3 Terminal block layout

Table 4.1 Contact function

Marking	Description
DI1–DI32	Inputs DI1-DI32
COM (for MV210-214)	Common input point
SS (for MV210-212)	

4.4.1 Contact sensors

**NOTICE**  
 The SS / COM circuits are connected in the device.

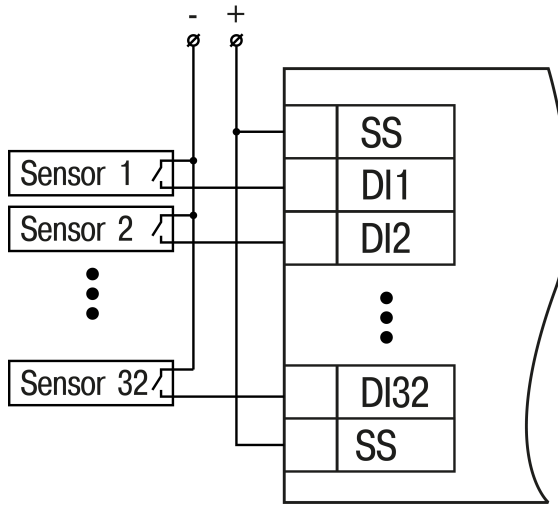


Fig. 4.4 Connection diagram to the inputs DI1–DI32 (MV210-212)

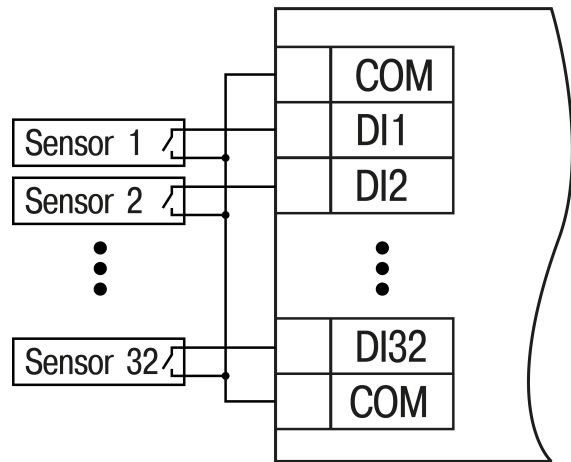


Fig. 4.5 Connection diagram to the inputs DI1 – DI32 (MV210-214)

4.4.2 PNP and NPN transistors

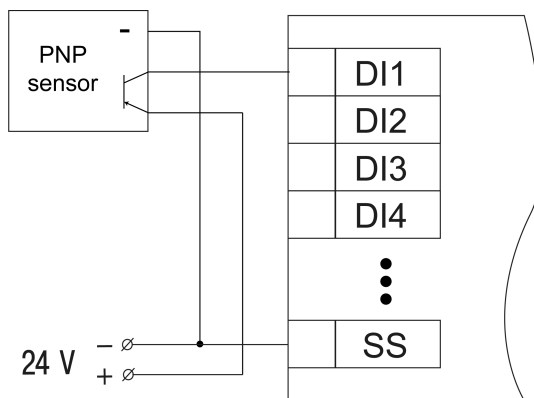


Fig. 4.6 Connection of PNP transistors (only for MV210-212)

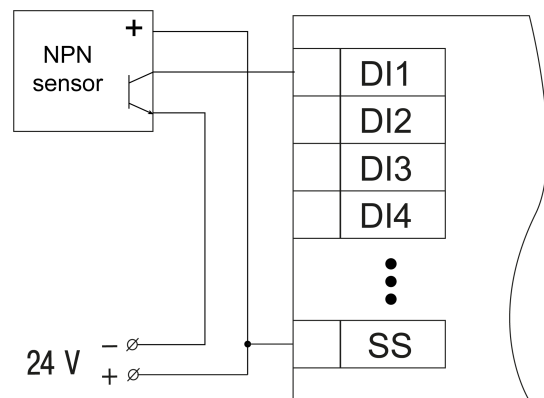


Fig. 4.7 Connection of NPN transistors (only for MV210-212)

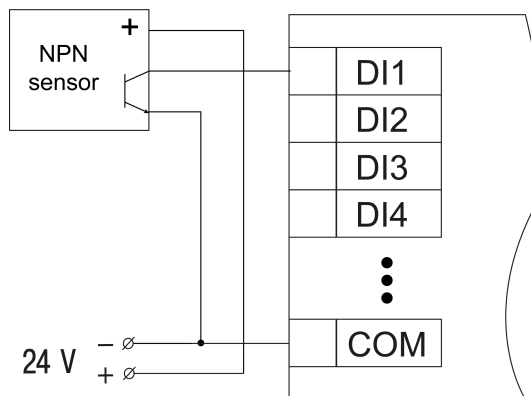


Fig. 4.8 Connection of NPN transistors (only for MV210-214)

## 4 Connection

### 4.4.3 PNP and NPN encoders

The figures below show the connection of the encoders to the module MV210-212.

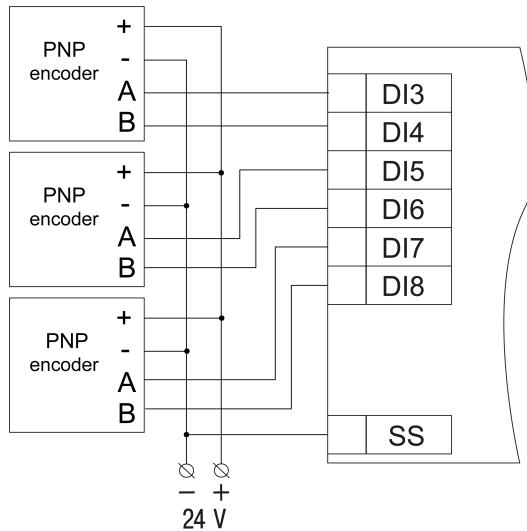


Fig. 4.9 Connection of pnp encoders

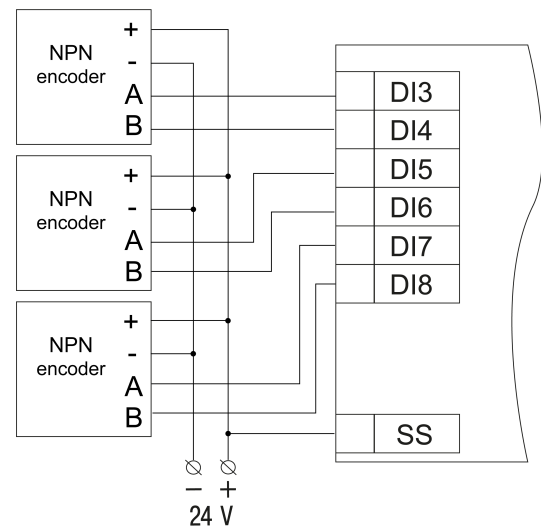


Fig. 4.10 Connection of npn encoders

### 4.5 Ethernet connection

To connect modules to an Ethernet network, you can use the following schemes:

- Wye (Figure 4.11);
- Daisy-chain (Figure 4.12).

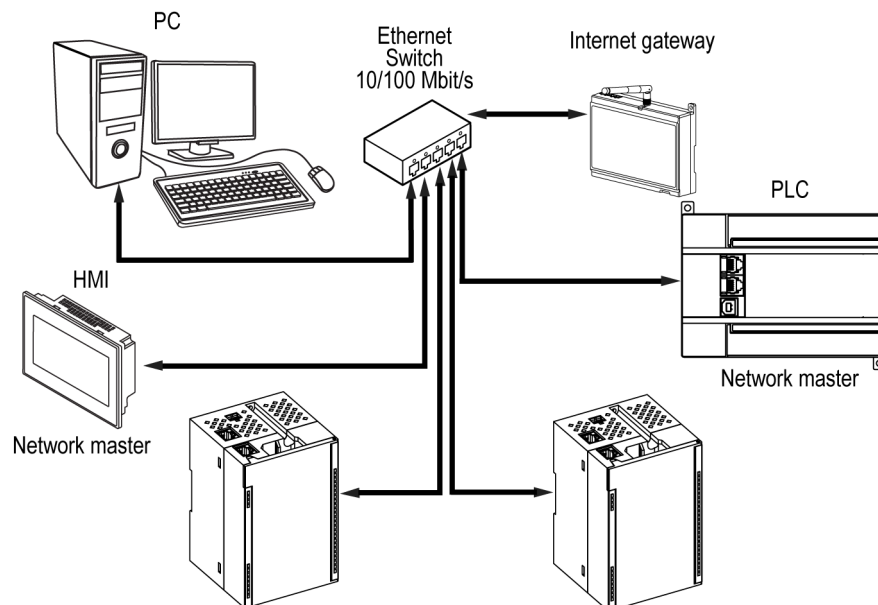


Fig. 4.11 Wye connection



#### NOTICE

1. Maximum length of communication lines: 100 m.
2. Connection is possible to any Ethernet port of the module.
3. The unused Ethernet port must be closed with a blanking plug.

To connect to the Daisy-chain scheme, you must use both Ethernet ports of the Module. If the Module fails or the power is turned off, the data will be transferred from port 1 to port 2 without disconnecting.

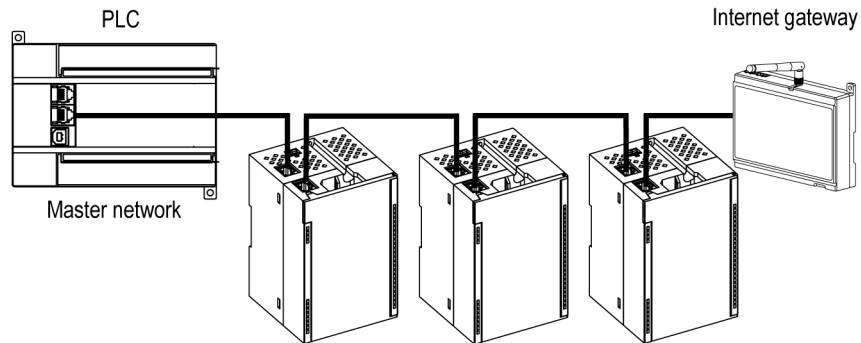


Fig. 4.12 Daisy-chain connection



### NOTICE

1. The maximum length of the communication line between two adjacent active devices when connected with “Daisy-Chain” scheme should be no more than 100 m.
2. An adjacent connection scheme is allowed.
3. The unused Ethernet port should be closed with a blanking plug.

## 5 Operation

### 5 Operation

#### 5.1 Operation principle

The Module receives commands from the Network Master and sends information about the status of digital inputs to it.

As a Network Master you can use:

- PC;
- PLC;
- Operator panel.

#### 5.2 Indication and control

On the front panel there are indication LEDs. The decoding of the LED indications is given in [Table 5.1](#).

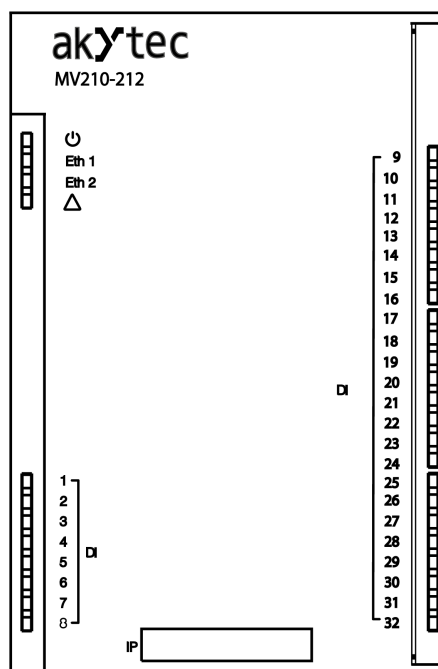


Fig. 5.1 Device faceplate



#### NOTICE

The IP field is located at the bottom of the front panel.

The IP field is intended for applying the IP address of the Module with a thin marker or on a paper sticker.

Table 5.1 Indication

LED	LED Status	Overview
Power ⏻ (green)	ON	Device operating voltage is applied
Eth 1 (green)	Flashing	Data transfer via Ethernet port 1
Eth 2 (green)	Flashing	Data transfer via Ethernet port 2
Fault ⚠ (red)	OFF	Normal operation
	ON	Main application and / or configuration failure



LED	LED Status	Overview
	Lights 200 ms once every 3 seconds	It is necessary to replace the battery of RTC
	Lights 100 ms twice a second (after a pause of 400 ms)	Module is in a safe state
	900 ms on, 100 ms off	Hardware peripheral failure (Flash, RTC, Ethernet Switch)
Input status indicators (green)	Green	Input is closed
	OFF	Input is open

Under the module faceplate there are terminal blocks and a service button.

The service button performs the following functions:

- Factory settings restore ([Section 6.6](#));
- IP-address assignment ([Section 6.2](#));
- Firmware update ([Section 6.4](#))

### 5.3 Real time clock

The Module has a built-in real time clock (RTC). Its source of power is a battery.

The timing is in seconds, starting with January 1, 2000 at midnight UTC. RTC indication is used for writing entries into the log file.

### 5.4 Log file

The module has built-in encrypted flash memory. The encryption algorithm is Data Encryption Standard (DES) in cipher block chaining (CBC) mode. The key is the string **superkey**. An initialization vector is generated using a hash function. The function argument is the password specified in akYtecToolPro. A checksum is calculated by the CRC32 algorithm and saved in the file end. The checksum is also encrypted.

Log file of Module will be saved as a few files. The log interval, the restriction on the size of one file and their number is set by the user in akYtecToolPro. If the log file is full, then the data is overwritten, starting with the oldest data in the oldest file.

A log file is a collection of records. Entries are separated by line break characters (0x0A0D). Each entry corresponds to one parameter and consists of fields separated by the “;” character (without quotes). Entry format is shown in [Table 5.2](#).

*Table 5.2 Entry format*

Parameter	Type	Size	Comment
Time stamp	binary data	4 bytes	In seconds from 00:00 01.01.2000 (UTC+0)
Separator	string	1 byte	Character “;” (without quotes)
Unique identifier of the parameter (UID)	string	8 byte	As a string of HEX characters with leading zeros
Separator	string	1 byte	Character “;” (without quotes)
Parameter value	string	depending on the parameter	As a string of HEX characters with leading zeros
Separator	string	1 byte	Character “;” (without quotes)

## 5 Operation

Parameter	Type	Size	Comment
Parameter Status	binary data	1 byte	parameter status in the log file (1 – the parameter value is correct, 0 – the parameter value is incorrect and its further processing is not recommended).
Line break	binary data	2 bytes	\n\r (0x0A0D)

Example of decrypted entry:

0x52 0x82 0xD1 0x24 **0x3B** 0x30 0x30 0x30 0x30 0x61 0x39 0x30 0x30 **0x3B** 0x30 0x30 0x30 0x30  
0x30 0x30 0x30 0x31 **0x3B** 0x31 **0x0A 0x0D**

where

- 0x52 0x82 0xD1 0x24 – time stamp. To get the date and time in UnixTime format, it is necessary to reverse the byte order and add the offset constant (number of seconds between 00:00:00 01/01/1970 and 00:00:00 01/01/2000): 0x24D18252 (HEX) + 946684800 (DEC) = 1564394971 (DEC, corresponding to July 29, 2019, 10:09:31);
- **0x3B** – separator;
- 0x30 0x30 0x30 0x30 0x30 0x61 0x39 0x30 0x30 – unique identifier of the parameter (00003ba00);
- **0x3B** – separator;
- 0x30 0x30 0x30 0x30 0x30 0x30 0x30 0x31 – parameter value (00000001);
- **0x3B** – separator;
- 0x31 – parameter status (1 – parameter value is correct);
- **0x0A 0x0D** – line break characters.

The time is written to the file from the built-in real time clock. You can also set the time zone, which will be read by external software.

The log file in the Device is written with a interval specified by user. Writing occurs with a certain frequency, calculated in such a way that the resource of Device flash memory is sufficient for a period of at least 10 years of operation.

Log file can be read:

- by akYtecToolPro;
- by user software (using Modbus function 20).

The list of logged parameters is available in the akYtecToolPro software on the **Device Information** tab. The order of writing parameters to the log file corresponds to the order of parameters on the tab.



### NOTE

After updating the firmware, all device settings except the network settings will be reset to the factory settings.

The log file is read using the Modbus function 20 (0x14). This function returns the contents of the registers of the memory file. The function allows using one request to read one or several records from one or several files.

The file read request for each entry indicates:

- link type, 1 byte (should be equal to 6);
- file number, 2 bytes;
- starting address of the register inside the file, 2 bytes;
- number of registers to read, 2 bytes.



### NOTE

The file number in the Modbus request is calculated as 4096 + the file sequence number. Sequential file numbering is new from scratch. The Last log file index contains the sequential number of the log file for the last time data were written.

The number of registers to read in the request should be selected so that the length of the response does not exceed the allowable length of the Modbus packet (256 bytes).

The size of the log file is not known in advance, so you should read portions of the data using separate queries. If in response to the request a message is received with error code 0x04 (MODBUS\_SLAVE\_DEVICE\_FAILURE), then you can conclude that the addresses of the registers

in the request are outside the file. To read the latest file data, you need to reduce the number of registers in the request.

**CAUTION**

When the Module power is turned off, the last entry made at the time of power off may not be saved.

### 5.5 Data exchange modes

The module has the following data exchange modes:

- exchange with the Master via the Modbus TCP protocol (port 502) – up to 4 simultaneous connections with different Network Masters;
- connection and data exchange with a PC using the akYtec Tool Pro.

#### 5.5.1 Modbus TCP communication

Table 5.3 Reading and writing parameters using the Modbus TCP protocol

Operation	Function
Reading	3 (0x03) or 4 (0x04)
Writing	6 (0x06) or 16 (0x10)

The list of Modbus registers can be read from the device using the akYtecToolPro program. The list of Modbus registers is also presented in the tables below.

Table 5.4 General Registers for Online Modbus Communication

Name	Register	Size/type/description
Device name (DEV)	0xF000	Character string up to 32 bytes, Win1251 encoding
Firmware version (VER)	0xF010	Character string up to 32 bytes, Win1251 encoding
Platform name	0xF020	Character string up to 32 bytes, Win1251 encoding
Platform version	0xF030	Character string up to 32 bytes, Win1251 encoding
Hardware version	0xF040	Character string up to 16 bytes, Win1251 encoding
Additional character information	0xF048	Character string up to 16 bytes, Win1251 encoding
Time and date	0xF080	4 bytes, in seconds since 2000
Time zone	0xF082	2 bytes, signed short, offset in minutes from Greenwich
Serial number	0xF084	Character string 32 bytes, encoding Win1251, 17 characters are used

Table 5.5 ModBus communication registers

Parameter	Value (unit)	Register address		Access	Data format
		DEC	HEX		
Time and date (UTC)	since 2000 (s)	61568	0xF080	Read only	UINT 32
New time	since 2000 (s)	61565	0xF07D	Read and write	UINT 32

Parameter	Value (unit)	Register address		Access	Data format
		DEC	HEX		
Save new time	0 – do not write; 1 – write	61567	0xF07F	Read and write	UINT 16
Time zone	offset in minutes from Greenwich	61570	0xF082	Read and write	UINT 16
Time (ms)	—	61563	0xF07B	Read only	UINT 32
Current IP address	—	26	0x1A	Read only	UINT 32
Current subnet mask	—	28	0x1C	Read only	UINT 32
Current gateway IP address	—	30	0x1E	Read only	UINT 32
Assign IP address	—	20	0x14	Read and write	UINT 32
Enter subnet mask	—	22	0x16	Read and write	UINT 32
Set the gateway IP address	—	24	0x18	Read and write	UINT 32
DHCP Mode	0 – absolute prohibition 1 – read only 2 – write only	32	0x20	Read and write	UINT 16
Safe state activation timeout	0...60 (s)	700	0x2BC	Read and write	UINT 8
Device status	—	61620	0xF0B4	Read only	UINT 32
MAC-address	—	61696	0xF100	Read only	UINT 48
DNS server 1	—	12	0xC	Read and write	UINT 32
DNS server 2	—	14	0xE	Read and write	UINT 32
Battery status (power supply)	0...3300 (mV)	801	0x321	Read only	UINT 16
Log interval	10...3600 (s) default value – 30 s	900	0x384	Read and write	UINT 16
Status of digital inputs, bit mask	0...4294967295	51	0x33	Read only	UINT 32
Additional mode for Input DI1*	0 – off; 1 – pulse count; 2 – period measurement; 3 – frequency measurement	64	0x40	Read and write	UINT 16
Additional mode for Input DI2*	0 – off; 1 – pulse count; 2 – period measurement; 3 – frequency measurement	65	0x41	Read and write	UINT 16

Parameter	Value (unit)	Register address		Access	Data format
		DEC	HEX		
Additional mode for Input DI3*	0 – off; 1 – pulse count; 2 – period measurement; 3 – frequency measurement; 4 – encoder signal processing	66	0x42	Read and write	UINT 16
Additional mode for Input DI4*	0 – off; 1 – pulse count; 2 – period measurement; 3 – frequency measurement; 4 – encoder signal processing	67	0x43	Read and write	UINT 16
Additional mode for Input DI5*	0 – off; 1 – pulse count; 2 – period measurement; 3 – frequency measurement; 4 – encoder signal processing	68	0x44	Read and write	UINT 16
Additional mode for Input DI6*	0 – off; 1 – pulse count; 2 – period measurement; 3 – frequency measurement; 4 – encoder signal processing	69	0x45	Read and write	UINT 16
Additional mode for Input DI7*	0 – off; 1 – pulse count; 2 – period measurement; 3 – frequency measurement; 4 – encoder signal processing	70	0x46	Read and write	UINT 16
Additional mode for Input DI8*	0 – off; 1 – pulse count; 2 – period measurement; 3 – frequency measurement; 4 – encoder signal processing	71	0x47	Read and write	UINT 16
Debounce filter for input DI1*	0 – off; 1 – on	96	0x60	Read and write	UINT 16

Parameter	Value (unit)	Register address		Access	Data format
		DEC	HEX		
Debounce filter for input DI2*	0 – off; 1 – on	97	0x61	Read and write	UINT 16
Debounce filter for input DI3*	0 – off; 1 – on	98	0x62	Read and write	UINT 16
Debounce filter for input DI4*	0 – off; 1 – on	99	0x63	Read and write	UINT 16
Debounce filter for input DI5*	0 – off; 1 – on	100	0x64	Read and write	UINT 16
Debounce filter for input DI6*	0 – off; 1 – on	101	0x65	Read and write	UINT 16
Debounce filter for input DI7*	0 – off; 1 – on	102	0x66	Read and write	UINT 16
Debounce filter for input DI8*	0 – off; 1 – on	103	0x67	Read and write	UINT 16
Debounce filter for input DI9*	0 – off; 1 – on	104	0x68	Read and write	UINT 16
Debounce filter for input DI10*	0 – off; 1 – on	105	0x69	Read and write	UINT 16
Debounce filter for input DI11*	0 – off; 1 – on	106	0x6A	Read and write	UINT 16
Debounce filter for input DI12*	0 – off; 1 – on	107	0x6B	Read and write	UINT 16
Debounce filter for input DI13*	0 – off; 1 – on	108	0x6C	Read and write	UINT 16
Debounce filter for input DI14*	0 – off; 1 – on	109	0x6D	Read and write	UINT 16
Debounce filter for input DI15*	0 – off; 1 – on	110	0x6E	Read and write	UINT 16
Debounce filter for input DI16*	0 – off; 1 – on	111	0x6F	Read and write	UINT 16
Debounce filter for input DI17*	0 – off; 1 – on	112	0x70	Read and write	UINT 16
Debounce filter for input DI18*	0 – off; 1 – on	113	0x71	Read and write	UINT 16
Debounce filter for input DI19*	0 – off; 1 – on	114	0x72	Read and write	UINT 16
Debounce filter for input DI20*	0 – off; 1 – on	115	0x73	Read and write	UINT 16
Debounce filter for input DI21*	0 – off; 1 – on	116	0x74	Read and write	UINT 16
Debounce filter for input DI22*	0 – off; 1 – on	117	0x75	Read and write	UINT 16
Debounce filter for input DI23*	0 – off; 1 – on	118	0x76	Read and write	UINT 16

Parameter	Value (unit)	Register address		Access	Data format
		DEC	HEX		
Debounce filter for input DI24*	0 – off; 1 – on	119	0x77	Read and write	UINT 16
Debounce filter for input DI25*	0 – off; 1 – on	120	0x78	Read and write	UINT 16
Debounce filter for input DI26*	0 – off; 1 – on	121	0x79	Read and write	UINT 16
Debounce filter for input DI27*	0 – off; 1 – on	122	0x7A	Read and write	UINT 16
Debounce filter for input DI28*	0 – off; 1 – on	123	0x7B	Read and write	UINT 16
Debounce filter for input DI29*	0 – off; 1 – on	124	0x7C	Read and write	UINT 16
Debounce filter for input DI30*	0 – off; 1 – on	125	0x7D	Read and write	UINT 16
Debounce filter for input DI31*	0 – off; 1 – on	126	0x7E	Read and write	UINT 16
Debounce filter for input DI32*	0 – off; 1 – on	127	0x7F	Read and write	UINT 16
Frequency measuring period for input DI1*	0 – 10 ms; 1 – 100 ms; 2 – 1 s; 3 – 10 s	128	0x80	Read and write	UINT 16
Frequency measuring period for input DI2*	0 – 10 ms; 1 – 100 ms; 2 – 1 s; 3 – 10 s	129	0x81	Read and write	UINT 16
Frequency measuring period for input DI3*	0 – 10 ms; 1 – 100 ms; 2 – 1 s; 3 – 10 s	130	0x82	Read and write	UINT 16
Frequency measuring period for input DI4*	0 – 10 ms; 1 – 100 ms; 2 – 1 s; 3 – 10 s	131	0x83	Read and write	UINT 16
Frequency measuring period for input DI5*	0 – 10 ms; 1 – 100 ms; 2 – 1 s; 3 – 10 s	132	0x84	Read and write	UINT 16
Frequency measuring period for input DI6*	0 – 10 ms; 1 – 100 ms; 2 – 1 s; 3 – 10 s	133	0x85	Read and write	UINT 16
Frequency measuring period for input DI7*	0 – 10 ms; 1 – 100 ms; 2 – 1 s;	134	0x86	Read and write	UINT 16

Parameter	Value (unit)	Register address		Access	Data format
		DEC	HEX		
	3 – 10 s				
Frequency measuring period for input DI8*	0 – 10 ms; 1 – 100 ms; 2 – 1 s; 3 – 10 s	135	0x87	Read and write	UINT 16
Value on input DI1 in additional mode*	0 ... 4294967295 (pulses) – mode 1; 10...4294967295 (ms) – mode 2 100 ... 4294967295 (Hz) – mode 3; 0 ... 4294967295 (pulses) – mode 4	160	0xA0	Read only	UINT 32
Value on input DI2 in additional mode*	0 ... 4294967295 (pulses) – mode 1; 10...4294967295 (ms) – mode 2 100 ... 4294967295 (Hz) – mode 3; 0 ... 4294967295 (pulses) – mode 4	162	0xA2	Read only	UINT 32
Value on input DI3 in additional mode*	0 ... 4294967295 (pulses) – mode 1; 10...4294967295 (ms) – mode 2 100 ... 4294967295 (Hz) – mode 3; 0 ... 4294967295 (pulses) – mode 4	164	0xA4	Read only	UINT 32
Value on input DI4 in additional mode*	0 ... 4294967295 (pulses) – mode 1; 10...4294967295 (ms) – mode 2 100 ... 4294967295 (Hz) – mode 3; 0 ... 4294967295 (pulses) – mode 4	166	0xA6	Read only	UINT 32
Value on input DI5 in additional mode*	0 ... 4294967295 (pulses) – mode 1; 10...4294967295 (ms) – mode 2	168	0xA8	Read only	UINT 32



Parameter	Value (unit)	Register address		Access	Data format
		DEC	HEX		
	100 ... 4294967295 (Hz) – mode 3; 0 ... 4294967295 (pulses) – mode 4				
Value on input DI6 in additional mode*	0 ... 4294967295 (pulses) – mode 1; 10...4294967295 (ms) – mode 2 100 ... 4294967295 (Hz) – mode 3; 0 ... 4294967295 (pulses) – mode 4	170	0xAA	Read only	UINT 32
Value on input DI7 in additional mode*	0 ... 4294967295 (pulses) – mode 1; 10...4294967295 (ms) – mode 2 100 ... 4294967295 (Hz) – mode 3; 0 ... 4294967295 (pulses) – mode 4	172	0xAC	Read only	UINT 32
Value on input DI8 in additional mode*	0 ... 4294967295 (pulses) – mode 1; 10...4294967295 (ms) – mode 2 100 ... 4294967295 (Hz) – mode 3; 0 ... 4294967295 (pulses) – mode 4	174	0xAE	Read only	UINT 32
Value of pulse counter on input DI9	0...4294967295 (pulses)	176	0xB0	Read only	UINT 32
Value of pulse counter on input DI10	0...4294967295 (pulses)	178	0xB2	Read only	UINT 32
Value of pulse counter on input DI11	0...4294967295 (pulses)	180	0xB4	Read only	UINT 32
Value of pulse counter on input DI12	0...4294967295 (pulses)	182	0xB6	Read only	UINT 32
Value of pulse counter on input DI13	0...4294967295 (pulses)	184	0xB8	Read only	UINT 32

Parameter	Value (unit)	Register address		Access	Data format
		DEC	HEX		
Value of pulse counter on input D114	0...4294967295 (pulses)	186	0xBA	Read only	UINT 32
Value of pulse counter on input D115	0...4294967295 (pulses)	188	0xBC	Read only	UINT 32
Value of pulse counter on input D116	0...4294967295 (pulses)	190	0xBE	Read only	UINT 32
Value of pulse counter on input D117	0...4294967295 (pulses)	192	0xC0	Read only	UINT 32
Value of pulse counter on input D118	0...4294967295 (pulses)	194	0xC2	Read only	UINT 32
Value of pulse counter on input D119	0...4294967295 (pulses)	196	0xC4	Read only	UINT 32
Value of pulse counter on input D120	0...4294967295 (pulses)	198	0xC6	Read only	UINT 32
Value of pulse counter on input D121	0...4294967295 (pulses)	200	0xC8	Read only	UINT 32
Value of pulse counter on input D122	0...4294967295 (pulses)	202	0xCA	Read only	UINT 32
Value of pulse counter on input D123	0...4294967295 (pulses)	204	0xCC	Read only	UINT 32
Value of pulse counter on input D124	0...4294967295 (pulses)	206	0xCE	Read only	UINT 32
Value of pulse counter on input D125	0...4294967295 (pulses)	208	0xD0	Read only	UINT 32
Value of pulse counter on input D126	0...4294967295 (pulses)	210	0xD2	Read only	UINT 32
Value of pulse counter on input D127	0...4294967295 (pulses)	212	0xD4	Read only	UINT 32
Value of pulse counter on input D128	0...4294967295 (pulses)	214	0xD6	Read only	UINT 32
Value of pulse counter on input D129	0...4294967295 (pulses)	216	0xD8	Read only	UINT 32
Value of pulse counter on input D130	0...4294967295 (pulses)	218	0xDA	Read only	UINT 32

Parameter	Value (unit)	Register address		Access	Data format
		DEC	HEX		
Value of pulse counter on input DI31	0...4294967295 (pulses)	220	0xDC	Read only	UINT 32
Value of pulse counter on input DI32	0...4294967295 (pulses)	222	0xDE	Read only	UINT 32
Reset the value of additional mode for input DI1*	0 – reset 1 – do not reset	224	0xE0	Read and write	UINT 16
Reset the value of additional mode for input DI2*	0 – reset 1 – do not reset	225	0xE1	Read and write	UINT 16
Reset the value of additional mode for input DI3*	0 – reset 1 – do not reset	226	0xE2	Read and write	UINT 16
Reset the value of additional mode for input DI4*	0 – reset 1 – do not reset	227	0xE3	Read and write	UINT 16
Reset the value of additional mode for input DI5*	0 – reset 1 – do not reset	228	0xE4	Read and write	UINT 16
Reset the value of additional mode for input DI6*	0 – reset 1 – do not reset	229	0xE5	Read and write	UINT 16
Reset the value of additional mode for input DI7*	0 – reset 1 – do not reset	230	0xE6	Read and write	UINT 16
Reset the value of additional mode for input DI8*	0 – reset 1 – do not reset	231	0xE7	Read and write	UINT 16
Reset pulse counter on input DI9	0 – reset 1 – do not reset	232	0xE8	Read and write	UINT 16
Reset pulse counter on input DI10	0 – reset 1 – do not reset	233	0xE9	Read and write	UINT 16
Reset pulse counter on input DI11	0 – reset 1 – do not reset	234	0xEA	Read and write	UINT 16
Reset pulse counter on input DI12	0 – reset 1 – do not reset	235	0xEB	Read and write	UINT 16
Reset pulse counter on input DI13	0 – reset 1 – do not reset	236	0xEC	Read and write	UINT 16

Parameter	Value (unit)	Register address		Access	Data format
		DEC	HEX		
Reset pulse counter on input DI14	0 – reset 1 – do not reset	237	0xED	Read and write	UINT 16
Reset pulse counter on input DI15	0 – reset 1 – do not reset	238	0xEE	Read and write	UINT 16
Reset pulse counter on input DI16	0 – reset 1 – do not reset	239	0xEF	Read and write	UINT 16
Reset pulse counter on input DI17	0 – reset 1 – do not reset	240	0xEF	Read and write	UINT 16
Reset pulse counter on input DI18	0 – reset 1 – do not reset	241	0xF0	Read and write	UINT 16
Reset pulse counter on input DI19	0 – reset 1 – do not reset	242	0xF2	Read and write	UINT 16
Reset pulse counter on input DI20	0 – reset 1 – do not reset	243	0xF3	Read and write	UINT 16
Reset pulse counter on input DI21	0 – reset 1 – do not reset	244	0xF4	Read and write	UINT 16
Reset pulse counter on input DI22	0 – reset 1 – do not reset	245	0xF5	Read and write	UINT 16
Reset pulse counter on input DI23	0 – reset 1 – do not reset	246	0xF6	Read and write	UINT 16
Reset pulse counter on input DI24	0 – reset 1 – do not reset	247	0xF7	Read and write	UINT 16
Reset pulse counter on input DI25	0 – reset 1 – do not reset	248	0xF8	Read and write	UINT 16
Reset pulse counter on input DI26	0 – reset 1 – do not reset	249	0xF9	Read and write	UINT 16
Reset pulse counter on input DI27	0 – reset 1 – do not reset	250	0xFA	Read and write	UINT 16
Reset pulse counter on input DI28	0 – reset 1 – do not reset	251	0xFB	Read and write	UINT 16
Reset pulse counter on input DI29	0 – reset 1 – do not reset	252	0xFC	Read and write	UINT 16
Reset pulse counter on input DI30	0 – reset 1 – do not reset	253	0xFD	Read and write	UINT 16

## 5 Operation

Parameter	Value (unit)	Register address		Access	Data format
		DEC	HEX		
Reset pulse counter on input DI31	0 – reset 1 – do not reset	254	0xFE	Read and write	UINT 16
Reset pulse counter on input DI32	0 – reset 1 – do not reset	255	0xFF	Read and write	UINT 16

\* Only for MV210-212.

### 5.5.2 Error codes for Modbus protocol

When working on the Modbus protocol, errors may occur. These errors are described in [Table 5.6](#). In case of an error, Module sends a response to the Network Master with an error code.

*Table 5.6 List of possible errors*

Name	Code	Description
MODBUS_ILLEGAL_FUNCTION	01 (0x01)	Illegal function code. The error occurs if Module does not support the Modbus function specified in the request.
MODBUS_ILLEGAL_DATA_ADDRESS	02 (0x02)	Illegal register address. The error occurs if the request contains register addresses that are not in Module.
MODBUS_ILLEGAL_DATA_VALUE	03 (0x03)	Illegal data value. The error occurs if the request contains an invalid value for writing to the register
MODBUS_SLAVE_DEVICE_FAILURE	04 (0x04)	The error occurs if the requested action cannot be completed.

During the exchange via the Modbus protocol, Module checks the compliance of the requests with the Modbus specification. Requests that fail verification are ignored by the module. Requests that specify an address that does not match the module address are also ignored.

Next, the function code is checked. If a request is received by the module with a function code not specified in [Table 5.7](#), a MODBUS\_ILLEGAL\_FUNCTION error occurs.

*Table 5.7 List of supported functions*

Name	Code	Description
MODBUS_READ_HOLDING_REGISTERS	3 (0x03)	Reading values from one or more holding registers
MODBUS_READ_INPUT_REGISTERS	4 (0x04)	Reading values from one or more input registers
MODBUS_WRITE_SINGLE_REGISTER	6 (0x06)	Writing a value to single register
MODBUS_WRITE_MULTIPLE_REGISTERS	16 (0x10)	Writing values to multiple registers
MODBUS_READ_FILE_RECORD	20 (0x14)	Reading log from file
MODBUS_WRITE_FILE_RECORD	21 (0x15)	Writing log to file

Situations leading to errors during operation with registers are described in [Table 5.8](#).

Table 5.8 Errors while working with registers

Function	Error name	Possible causes
MODBUS_READ_HOLDING_REGISTERS	MODBUS_ILLEGAL_DATA_ADDRESS	<ul style="list-style-type: none"> <li>– number of requested registers is greater than the maximum possible number (125);</li> <li>– request for nonexistent parameter</li> </ul>
MODBUS_READ_INPUT_REGISTERS	MODBUS_ILLEGAL_DATA_ADDRESS	<ul style="list-style-type: none"> <li>– number of requested registers is greater than the maximum possible number (125);</li> <li>– request for nonexistent parameter</li> </ul>
MODBUS_WRITE_SINGLE_REGISTER	MODBUS_ILLEGAL_DATA_ADDRESS	<ul style="list-style-type: none"> <li>– attempt to write a parameter whose size exceeds 2 bytes;</li> <li>– attempt to write a parameter, access to which is denied;</li> <li>– attempt to write a parameter of this type, which cannot be written to by this function. Supported Types:               <ul style="list-style-type: none"> <li>– signed and unsigned integers (max. 2 bytes);</li> <li>– enumerated type;</li> <li>– float16 (currently this type is not used for Module).</li> </ul> </li> <li>– request for nonexistent parameter</li> </ul>
	MODBUS_ILLEGAL_DATA_VALUE	<ul style="list-style-type: none"> <li>– value outside the parameter limits</li> </ul>
MODBUS_WRITE_MULTIPLE_REGISTERS	MODBUS_ILLEGAL_DATA_ADDRESS	<ul style="list-style-type: none"> <li>– writing of a nonexistent parameter;</li> <li>– attempt to write a parameter, access to which is denied;</li> <li>– number of writable registers is greater than the maximum possible number (123)</li> </ul>
	MODBUS_ILLEGAL_DATA_VALUE	<ul style="list-style-type: none"> <li>– no terminating character (\0) was found in the string parameter;</li> <li>– size of the requested data is less than the size of the first or last parameter in the request;</li> <li>– value outside the parameter limits</li> </ul>

Situations leading to errors during operation with the log file are described in [Table 5.9](#).

Table 5.9 Errors while working with the log file

Function	Error name	Possible causes
MODBUS_READ_FILE_RECORD	MODBUS_ILLEGAL_FUNCTION	<ul style="list-style-type: none"> <li>– illegal data size (0x07 &lt;= data length &lt;= 0xF5)</li> </ul>
	MODBUS_ILLEGAL_DATA_ADDRESS	<ul style="list-style-type: none"> <li>– reference type does not meet specification;</li> <li>– could not open the file for reading (it may be missing)</li> </ul>
	MODBUS_ILLEGAL_DATA_VALUE	<ul style="list-style-type: none"> <li>– could not move to the desired offset in the file</li> </ul>

Function	Error name	Possible causes
	MODBUS_SLAVE_DEVICE_FAILURE	<ul style="list-style-type: none"> <li>– file deletion error when deleting;</li> <li>– request too much data (more than 250 bytes);</li> <li>– illegal record number (more than 0x270F);</li> <li>– illegal record length (more than 0x7A)</li> </ul>
MODBUS_WRITE_FILE_RECORD	MODBUS_ILLEGAL_FUNCTION	– illegal data size (0x09 <= data length <= 0xFB)
	MODBUS_ILLEGAL_DATA_ADDRESS	<ul style="list-style-type: none"> <li>– reference type does not meet specification;</li> <li>– could not open file for writing</li> </ul>
	MODBUS_SLAVE_DEVICE_FAILURE	<ul style="list-style-type: none"> <li>– requested file is missing;</li> <li>– requested file is read-only;</li> <li>– failed to write the required number of bytes</li> </ul>

### 5.6 Digital inputs operation modes

The inputs of the module operate in the logical level detection mode.  
For each input, the counter of incoming pulses is used.



#### NOTE

By default, counters are always on. Counters on inputs with additional modes are enabled only if the additional mode at the input is disabled.

Table 5.10 Pulse counter parameters

Parameter	Value
Capacity	32 bit
Maximum input frequency	400 Hz
Debounce	On/Off
Debounce time	25 ms (not configurable)



#### CAUTION

To work with signals with a frequency of more than 40 Hz at a duty cycle of 0.5 or less, do not turn on the debounce, since the useful signal will be considered as bounce and skipped.

If the counter overflows, then the corresponding register is reset automatically. The procedure for forced resetting is given in [Section 6.7](#).



#### NOTICE

Counters are non-volatile, their values are saved after a reboot. Counters on inputs with additional modes after reboot are reset.

The state values of the digital inputs are stored as a bitmask and read from the corresponding register.

The first 8 inputs of the MV220-212 Module support the following additional modes of operation:

- counting the number of high-frequency pulses (see [Section 5.6.2](#));
- frequency measurement (see [Section 5.6.3](#));
- encoder signal processing (see [Section 5.6.4](#)). Up to 3 encoders can be connected.



#### NOTICE

Only one additional operation mode can be switched on at one input.

Select and configure modes:

- in the program akYtecToolPro (parameter **Input mode**);
- via Modbus TCP Protocol.

### 5.6.1 Logic level detection

Module inputs perform the logic level detection.

By polling, the Module returns to the Modbus register a bit mask corresponding to the state of the inputs. The logic level detection is performed continuously and does not depend on the activation of the pulse counter mode, frequency measurement, or processing of the encoder signals.

### 5.6.2 Counting of high-frequency pulses

The counting of the number of high-frequency pulses works for the inputs DI1 – DI8.

The maximum frequency of the input signal is 100 kHz with duty ratio of 0.5.

A 32-bit counter is used for each input. The values of the input counter are read from the register corresponding to this input or from the status and log files. The LSB corresponds to one pulse.

If the counter overflows, the value is reset. To force a counter reset, write 0 to input value reset register. The counter will be reset within 10 ms after writing the value 0.

### 5.6.3 Frequency measurement

The first 8 inputs can be set to a frequency measurement mode, in which a square wave signal with a frequency from 0 Hz to 100 kHz is fed to the input.

For this mode, you must set the measurement range:

- from 0 Hz to 100 Hz;
- 100 Hz to 100 kHz.

In the range from 0 to 100 Hz, the period of the input signal is measured in ms.

In the range from 100 Hz to 100 kHz, the frequency is calculated by counting the number of pulses received at the input during the measurement period.

The value of the measurement period is set during configuration using the program akYtecToolPro in the parameter **Measuring period**.

Possible measurement period values: 10 ms, 100 ms, 1 s, 10 s. Default value is 1 s.

A 32-bit register is enabled for each input. Input values are read from the register corresponding to this input or from the status and log files.

### 5.6.4 Encoder signal processing

Up to three dual-channel encoders can be connected to the module (without the Z channel).  
Connection input numbers: 3-4, 5-6, 7-8.

The maximum frequency of the encoder signals is 100 kHz.

For each input, a 32-bit register is used, in which the accumulated number of pulses is located, taking into account the direction of rotation after reset, recorded by the encoder. If, after reset, the direction of rotation is reversed, the counting starts from the maximum and decreases by the number of accumulated pulses.

Input values are read from the register corresponding to this input or from the log file.

If the counter overflows, the value is reset. To force a counter reset, write 0 to input value reset register. The counter will be reset within 10 ms after writing the value 0.



## 6 Configuration

### 6.1 Connection to akYtec Tool Pro

The Module is configured in the akYtec Tool Pro program.

The Device can be connected to a PC using the following interfaces:

- USB (microUSB),
- Ethernet.

To select an interface:

1. Connect the module to the PC using a USB cable or Ethernet interface.



#### NOTICE

If the module is connected to the USB port, the main module power supply is not required.  
In case of connecting via Ethernet interface it is necessary to supply the main power to the module.

2. Run akYtec Tool Pro.
3. Click the icon **Add devices** on the tool bar.
4. In the drop-down menu "Interface" select:
  - Ethernet (or other network card to which the module is connected): for Ethernet connection;
  - STMicroelectronics Virtual COM Port: for USB connection.

Connection settings

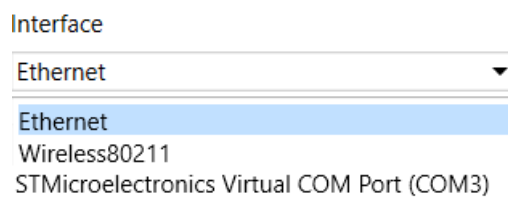


Fig. 6.1 Interface selection menu

The next steps for finding a Device depend on the choice of interface.

To find and add a Device connected to the Ethernet interface to the project:

1. Select "Find device"
2. Enter IP-address of the connected device.
3. Click Find. A Module with the specified IP address is displayed in the window.



#### NOTICE

The default IP address (factory setting) is **192.168.1.99**.

4. Select the Device (place a check mark) and click OK. If the Device is password protected, you must enter the correct password. The Device will be added to the project.

To find and add a device connected via USB interface to the project:

1. In the drop-down menu, select the akYtec Auto Detection Protocol.

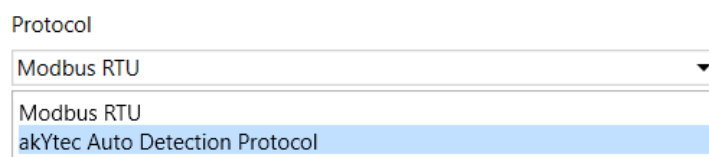


Fig. 6.2 Select Protocol

2. Select **Find device**.

## 6 Configuration

3. Enter the address of the connected device (default is **1**).
4. Click **Find**. The module with the specified address is displayed in the window.
5. Select the device (place a check mark) and click OK. If the device is password protected, you must enter the correct password. The device will be added to the project.

For more information on connecting and working with Devices, see the AkYtec Tool Pro Help. To call help in the program, press **F1**.

### 6.2 Configuring Network Settings

The parameters shown in the table must be set for the Module for Ethernet connection:

*Table 6.1 Module's network parameters*

Parameter	Note
MAC-address	Set at the factory and is unchanged
IP address	Can be static or dynamic. The factory setting is <b>192.168.1.99</b> .
Subnet mask	Specifies the subnet of the IP addresses of other devices visible by the Module. The factory setting is <b>255.255.255.0</b>
Gateway IP Address	Specifies the gateway address to access the Internet. The factory setting is <b>192.168.1.1</b>

The IP address can be:

**Static.** This IP address is set using the akYtecToolPro program or the Service button.

To set a static IP address using the akYtecToolPro, proceed as follows:

1. Click on the line **Connection Settings** in the parameter field.
2. Select the **Ethernet settings**.
3. Specify new values for fields **Enter IP Address**, **Enter Subnet Mask** and **Assign gateway IP address**.
4. Set **DHCP Mode** to **Off**.
5. Click on symbol "Save data" in the menu bar.

With help of service button, you can set IP addresses immediately for the group of Modules.

To assign IP address using service button:

1. Connect a module or group of Modules to an Ethernet network.
2. Start the akYtecToolPro program on a PC connected to the same Ethernet network.
3. Click on symbol "IP addresses" in akYtecToolPro program.
4. Enter IP address for the first Module or Module group.
5. Push the service buttons sequentially on the Modules, monitoring the result in the program window. In this case, the information about the Module on which the button was pressed will be displayed in the program window, this Module will be assigned a specified static IP address and other network parameters. The program automatically increments the address by 1.



#### **NOTICE**

If the IP address assignment with service button does not work, set the **DHCP mode** to **One-off setting with service button** in the akYtecToolPro program (default setting).

Ethernet settings	
Current IP address	10.2.20.15
Current subnet mask	255.255.0.0
Current gateway IP address	10.2.1.1
Enter IP address	10.2.11.122
Enter subnet mask	255.255.0.0
Assign gateway IP address	10.2.1.1
DHCP mode	One-off setting with service butt <input type="button" value="v"/>
	Off
	On
	One-off setting with service button

Fig. 6.3 DHCP mode settings

**Dynamic.** A dynamic IP address is used to work with the cloud service (not available yet) and does not imply working with the Modbus TCP Master. The IP address of the Module is set by the DHCP server of the Ethernet network.



**CAUTION**

Check with the system administration services if there is a DHCP server in the network area to which Module is connected. If using a dynamic IP address, you must enable the DHCP mode.



**NOTICE**

To use the new network settings, you need to restart Module. If the module is connected via USB, it must also be disconnected.

### 6.3 Module access password

To limit access to read and write configuration parameters, a password is used. You can set or change the password when configuring using the akYtec Tool Pro. If the password is lost, the factory settings must be restored (see [Section 6.6](#)). By default, the password is not set.

### 6.4 Firmware update

Firmware can be updated by the following ways:

- by using the USB interface;
- by using Ethernet (recommended).

To update via USB, follow the steps:

1. When the Module is powered on, press and hold the service button. The Module will enter the downloader mode.
2. Update the software with a special utility. The utility is available at [www.akytec.de](http://www.akytec.de).

To update via Ethernet, follow the steps:

1. In the akYtec Tool Pro, click on **Firmware update**.
2. Follow the program's instructions (the firmware file is available on [www.akytec.de](http://www.akytec.de));
3. Restart the Module.

During the update via Ethernet, the integrity of the firmware file and the checksum are checked.



**NOTICE**

A restart of the Module is required to complete the update. If the Module is connected via USB, it must also be disconnected.

### 6.5 Real-time clock setting

The real-time clock (RTC) can be set or read from the Module via Modbus registers and also using the akYtec Tool Pro program (see the program help).

To set a new time via Modbus registers:

1. Write the time value in the appropriate registers.
2. Set the value **1** in the current time update register for at least 1 second.
3. Write the value **0** in the current time update register.

The next writing of the current time can be made after 1 second.

### 6.6 Restore default settings



#### **CAUTION**

After restoring the factory settings, all previously configured settings, except network settings, will be deleted.

To restore the factory settings and reset the installed password:

1. Switch on the power.
2. Press and hold the service button for more than 12 seconds.
3. Switch off and switch on the Device.

After turning on, the Device will work with default settings.

### 6.7 Counter reset

If the input state counter overflows, then the corresponding register is reset automatically. To force the counter to zero, write the value 0 to the counter reset register.

## 7 Maintenance

### 7.1 General Instructions

During the maintenance work on the Device, the safety requirements must be considered. Maintenance of the Device is carried out at least once every 6 months and includes the following procedures:

- checking the Device mounting;
- checking the screw connections;
- removal of dust and dirt from the device terminal block.

### 7.2 Battery replacement

A replaceable CR2032 type battery is used to power the real-time clock.

Replace the battery if at least one of the following events occurs:

- LED **Fault** blinks (it lights for 200 ms within a 3 second interval). After power is turned off, RTC can still work for about 2 weeks without power supply.
- The last battery change was 6 years ago.

To replace the battery:

1. Power off the Module and all connected devices.
2. Remove the module from the DIN rail.
3. Raise cover 1.
4. Remove two screws 3.
5. Remove the terminal block 2 as shown in the Figure 7.1.

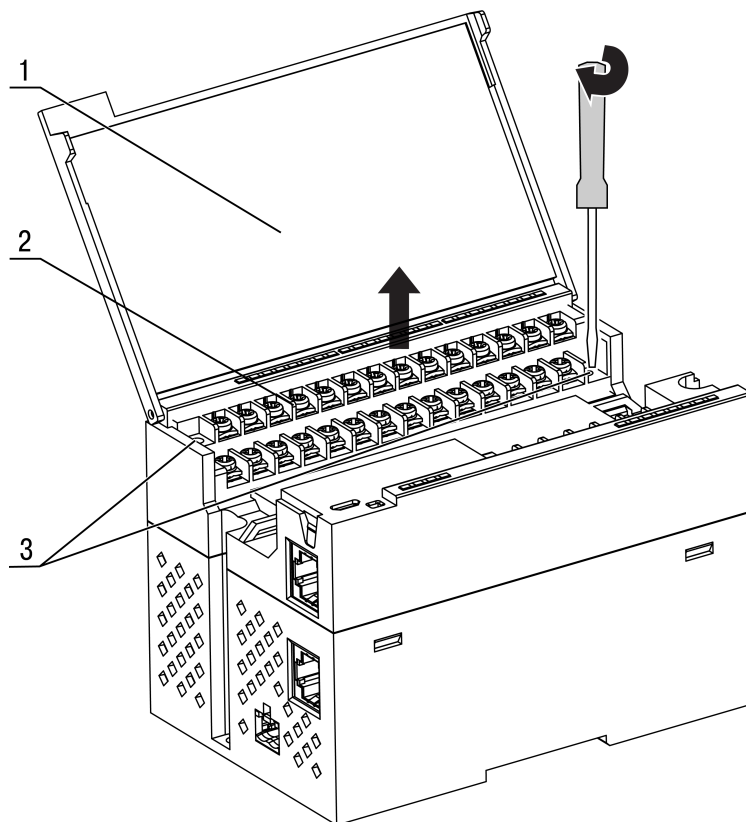


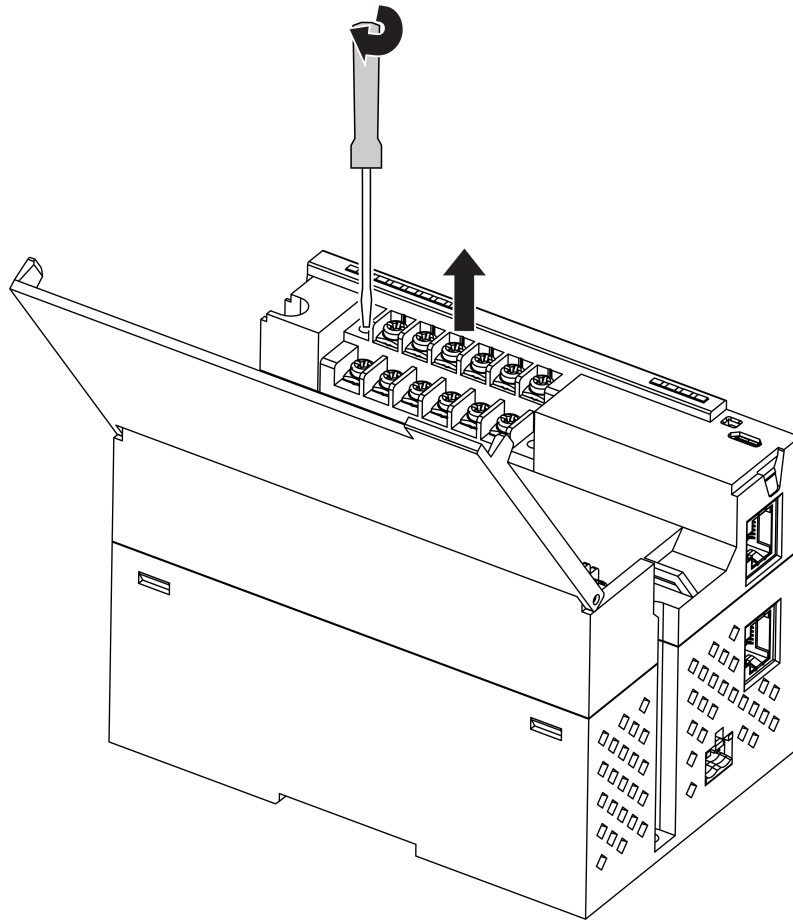
Fig. 7.1 Remove the first terminal block



#### NOTICE

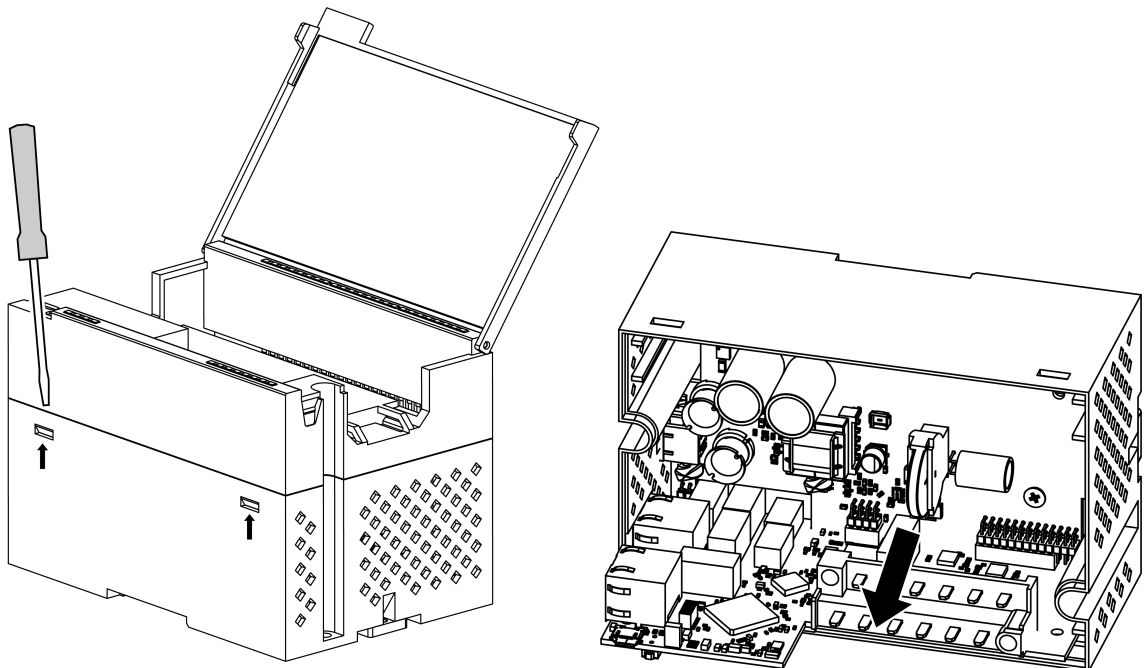
When loosening the fastening screws, the terminal block rises, so it is recommended to unscrew the screws by turns several turns at a time.

6. Remove the second terminal block as shown in *Figure 7.2*.



*Fig. 7.2 Remove the second terminal block*

7. Alternately remove the hooks from the holes on the both sides of Module and remove the top cover.



*Fig. 7.3 Battery replacement*

8. Replace the battery as shown in Figure 7.3.

**NOTE**

Recommended time for replacing the battery is not more than 1 minute. If the battery is missing for a longer time, the real-time clock must be set up again

9. Installation and mounting should be done in reverse order.

**CAUTION**

Do not use a different type of battery. When installing the battery, pay attention to the polarity.

After the Module is assembled and turned on, make sure that the system time is correct. If necessary, configure the real-time clock in the akYtecToolPro program.

### 8 Transportation and storage

Pack the device in such a way as to protect it reliably against impact for storage and transportation. The original packaging provides optimum protection.

If the device is not taken immediately after delivery into operation, it must be carefully stored at a protected location. The device should not be stored in an atmosphere with chemically active substances.

Permitted storage temperature: – 40...+55 °C



#### **NOTICE**

***The device may have been damaged during transportation.***

***Check the device for transport damage and completeness!***

***Report the transport damage immediately to the shipper and akYtec GmbH!***



## 9 Scope of delivery

---

### 9 Scope of delivery

Device	1
Short guide	1
UTP patch cable 5e 150 mm	1
Power supply terminal 2EGTK-5-02P-14	1
Ethernet blanking plug	1

**NOTICE**

The manufacturer reserves the right to introduce amendments to the scope of delivery.

### Appendix A Encrypting the log file

When decrypting the log file, a hash function should be used as the initialization vector. The hash function returns 8 bytes (type long long).

An example implementation of a hash function in C:

```
typedef union {
    struct {
        unsigned long lo;
        unsigned long hi;
    };
    long long hilo;
}LONG_LONG;

long long Hash8(const char *str) {    // Based on Rot13
    LONG_LONG temp;
    temp.lo = 0;
    temp.hi = 0;

    for ( ; *str; )
    {
        temp.lo += (unsigned char) (*str);
        temp.lo -= (temp.lo << 13) | (temp.lo >> 19);
        str++;
        if (!str) break;
        temp.hi += (unsigned char) (*str);
        temp.hi -= (temp.hi << 13) | (temp.hi >> 19);
        str++;
    }
    return temp.hilo;
}
```