



## MU210-401

### Input module

### User guide

MU210-401\_3-EN-43201-1.8  
© All rights reserved  
Subject to technical changes and misprints

## Contents

<b>Warning notice system .....</b>	<b>2</b>
<b>Safety.....</b>	<b>3</b>
<b>Introduction .....</b>	<b>4</b>
<b>Abbreviations .....</b>	<b>5</b>
<b>1 Overview.....</b>	<b>6</b>
<b>2 Specification.....</b>	<b>7</b>
2.1 Specifications .....	7
2.2 Isolation of the Device Components.....	8
2.3 Environmental conditions .....	8
<b>3 Installation.....</b>	<b>9</b>
<b>4 Connection .....</b>	<b>11</b>
4.1 Connection recommendations.....	11
4.2 Terminal block layout .....	11
4.3 Connectors .....	12
4.4 Power supply .....	12
4.5 Connection to the outputs .....	13
4.6 Ethernet connection .....	13
<b>5 Costruction and operation .....</b>	<b>15</b>
5.1 Operation principle .....	15
5.2 Indication and control .....	15
5.3 Real time clock.....	16
5.4 Log file .....	16
5.5 Data exchange modes .....	17
5.5.1 Modbus TCP communication .....	18
5.5.2 Error codes for Modbus protocol.....	21
5.6 Digital outputs operation modes .....	23
5.7 Safe output status .....	23
<b>6 Configuration.....</b>	<b>25</b>
6.1 Connection to akYtec Tool Pro .....	25
6.2 Configuring Network Settings .....	26
6.3 Module access password.....	27
6.4 Firmware update .....	27
6.5 Real-time clock setting.....	28
6.6 Restore default settings .....	28
<b>7 Maintenance .....</b>	<b>29</b>
7.1 General Instructions .....	29
7.2 Battery replacement .....	29
<b>8 Transportation and storage .....</b>	<b>31</b>
<b>9 Scope of delivery .....</b>	<b>32</b>
<b>Appendix A Encrypting the log file .....</b>	<b>33</b>

### Warning notice system

Explanation of the symbols and keywords used:

**DANGER**

DANGER indicates an **imminent dangerous situation** that will result in death or serious injuries if not prevented.

**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.

### 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 output module MU210-401, 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.

Order code: **MU210-401**.

### Abbreviations

- **PC** – personal computer.
- **PLC** – programmable logic controller.
- **PWM** – pulse width modulation.
- **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.

### 1 Overview

The Module is designed to control executive devices on automation facilities. The device is controlled by a PLC, panel controller, PC or other control device.

The module has 8 digital outputs of the relay type.

Modules are used in various fields of industry and agriculture.

## 2 Specification

### 2.1 Specifications

Table 2.1 Specifications

Denomination	Value	
<b>Power supply</b>		
Power supply	24 (10 ... 48) V DC	
Power consumption	6 W	
Protection against reverse polarity of supply voltage	Yes	
<b>Interfaces</b>		
Data transfer interface	Dual Port Ethernet 10/100 Mbit	
Configuration interface	USB 2.0 (MicroUSB), Ethernet 10/100 Mbit	
Date transfer protocol	Modbus TCP	
Protocol version	IPv4	
<b>Digital outputs</b>		
Number of outputs	8	
Output type	Electromagnetic relay	
Contact type	Normally open contact	
Operation mode	switching of a logic signal PWM signal generation	
Maximum voltage on relay contacts	250 V AC, 30 V DC	
Switching current	5 A (max. 250 V AC, 50 Hz and $\cos\varphi >> 0.4$ ) 3 A (max. 30 V DC)	
Turn-on time	15 ms	
Turn-off time	15 ms	
Load current interruption control	No	
<b>PWM output parameters</b>		
Maximum frequency	1 Hz (at 50% duty cycle)	
Minimum pulse width PWM	50 ms	
<b>Flash-memory (log-file)</b>		
Number of write and erase cycles	up to 100,000	
Maximum log file size	2 KB	
Maximum number of log files	1000	
Minimum log writing interval	10 s	
<b>Real time clock</b>		
Inaccuracy	at a temperature of +25 °C	max. 1 s per day
	over the entire temperature range	max. 10 s per day
Battery type	CR2032	
Lifetime of one battery	6 years	
<b>General information</b>		
Dimensions	42 × 124 × 83 mm	
IP code	IP20	
Error-free running time*	60,000 h	
Average service life time	10	
Weight	0.4 kg	

\* Excluding electromechanical switches and real-time clock battery



2.2 Isolation of the Device Components

The circuit of galvanic isolated components and the strength of galvanic isolation are shown in Figure 2.1.

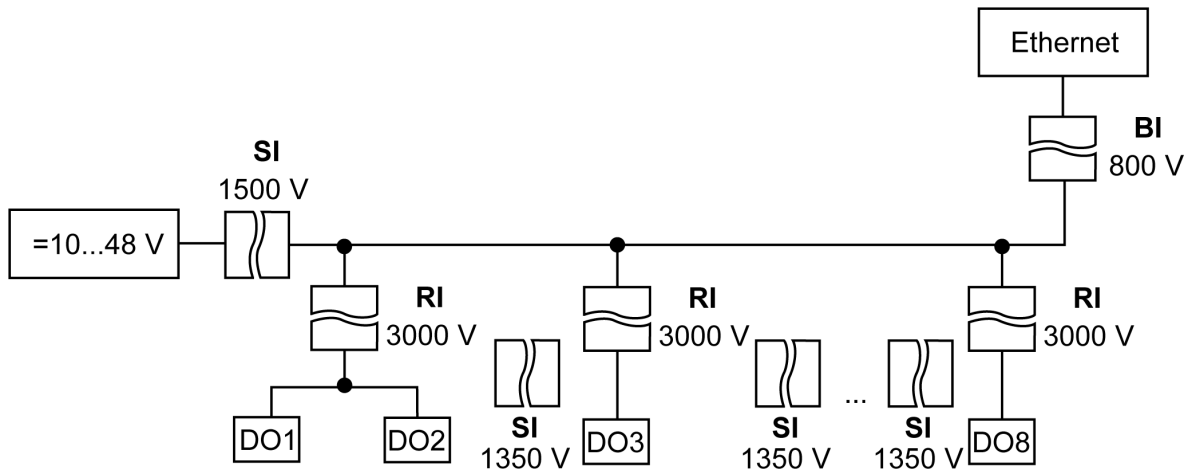


Fig. 2.1 Isolation of the Device Components

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)
Reinforced insulation (RI)	Separate insulation system used for active parts, which provides a degree of protection against electric shock, equivalent to double.



**NOTICE**

The strength of the isolation is indicated for testing under normal climatic conditions (exposure time is 1 minute).

Digital outputs (relays) are individually isolated from each other (except outputs 1 and 2, having a common terminal). Insulation strength between relay outputs is 1780 V.

2.3 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 ... +55 °C;
- relative humidity: up to 95% (at +35 °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 the Device is in accordance with IEC 61131-2-2012.

The resistance to climatic influences during the operation of the Device is in accordance with IEC 61131-2-2012.

### 3 Installation

Module is to install in the electrical cabinet. The design of the cabinet must protect Module from moisture, dirt and foreign objects.

To install Module:

1. Make sure that there is enough free space to connect the wire harness. You need 50 mm above the Module and below it.
2. Fasten the Device to the DIN rail or to a vertical surface using screws (see [Figure 3.1](#)).

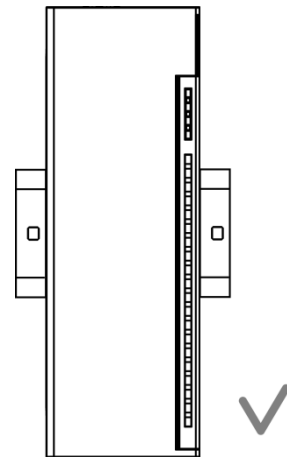


Fig. 3.1 Proper installation

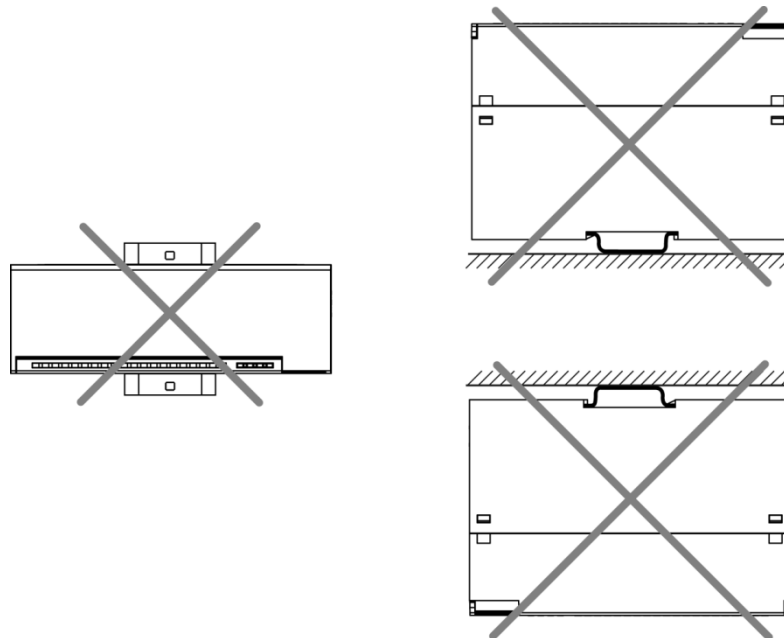


Fig. 3.2 Improper installation



**CAUTION**

Long-term operation of Module with improper installation can lead to damage (see [Figure 3.2](#)).

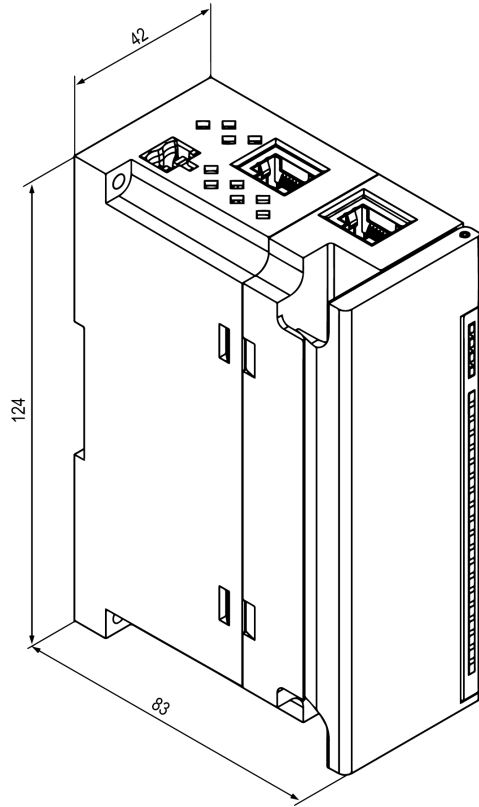


Fig. 3.3 Dimension drawing

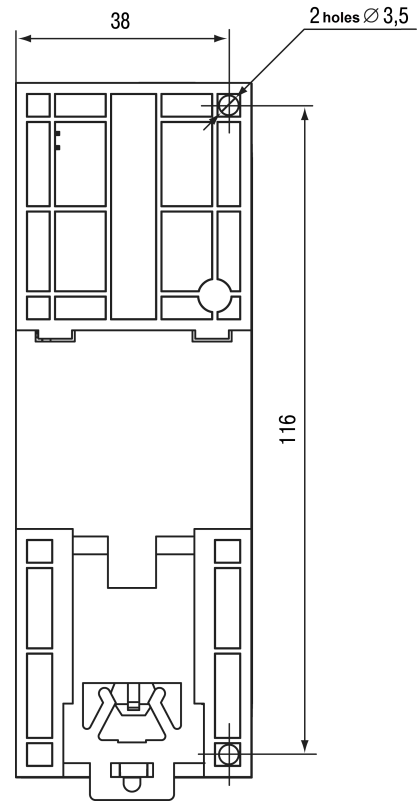


Fig. 3.4 Mounting dimensions

## 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 Terminal block layout

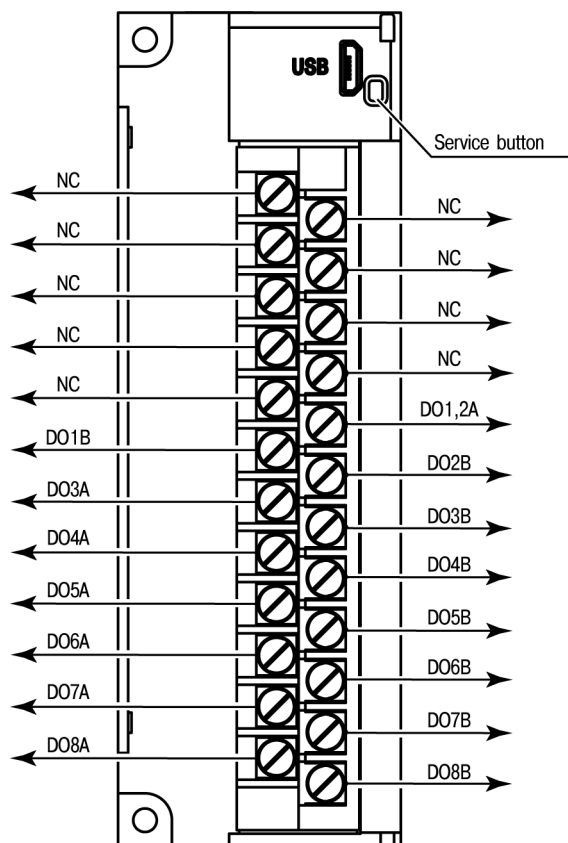


Fig. 4.1 Terminal block layout

Marking	Description
DO1A, DO1B – DO8A, DO8B	Outputs DO1 – DO8
NC (Not connected)	No connection

**CAUTION**

It is not allowed to connect wires to NC contacts (Not connected).

**4.3 Connectors**

The connectors of the interfaces and of the power supply of the Device are shown in Figure 4.2.

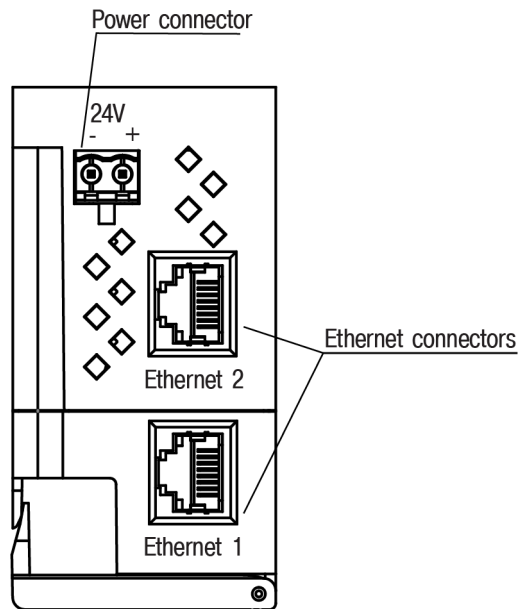


Fig. 4.2 Device's connectors

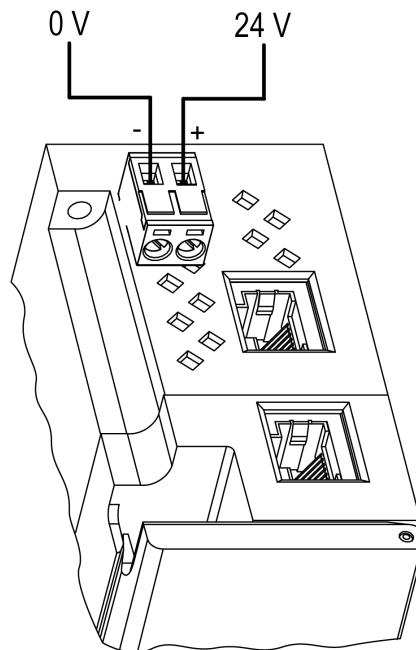
**4.4 Power supply**

Fig. 4.3 Power supply contacts

**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.

## 4.5 Connection to the outputs

Figure 4.4 shows the connection diagram to the outputs of the relay type.

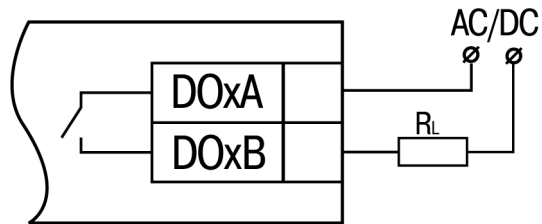


Fig. 4.4 External connections to digital outputs of the relay type

## 4.6 Ethernet connection

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

- Wye (Figure 4.5),
- Daisy-chain (Figure 4.6).

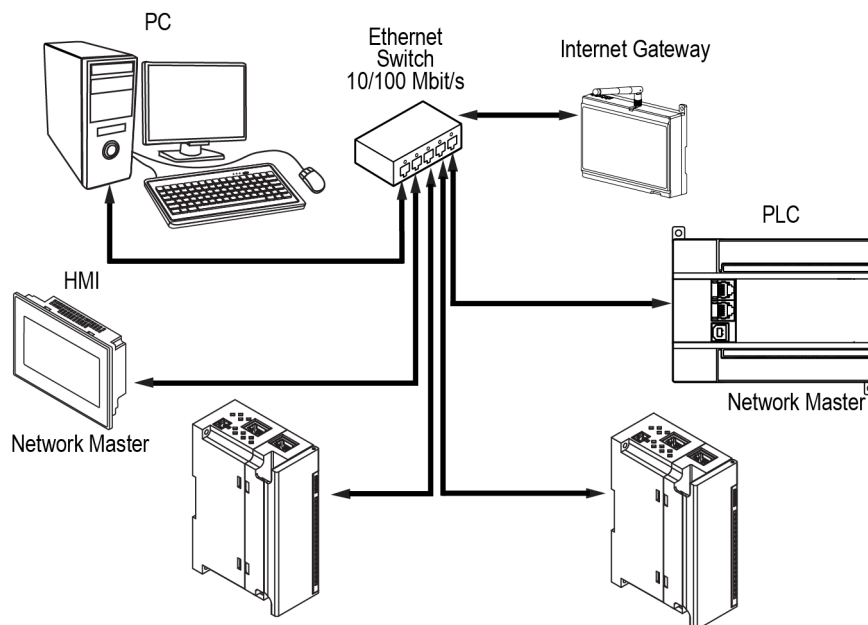


Fig. 4.5 Wye connection

**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 Module. If Module fails or the power is turned off, the data will be transferred from port 1 to port 2 without disconnecting.

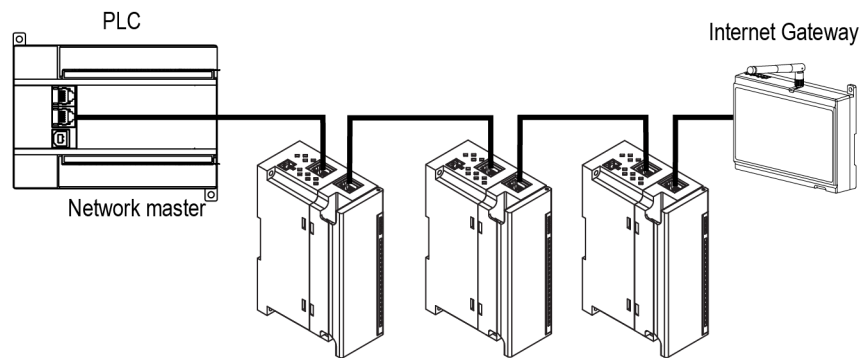


Fig. 4.6 Daisy-chain connection

**NOTICE**

1. The maximum length of the communication line between two adjacent active devices when connected with the "Daisy-Chain" must be not more than 100 m.
2. An adjacent connection scheme is allowed.
3. The unused Ethernet port should be closed with a blanking plug.

### 5 Costruction and operation

#### 5.1 Operation principle

The Module receives commands to control outputs from the Network Master.

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 LED indication is given in [Table 5.1](#).

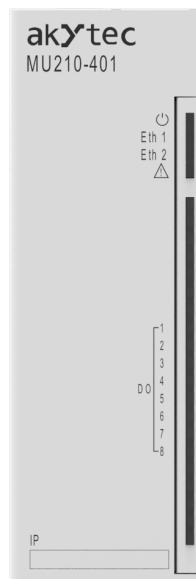


Fig. 5.1 Device faceplate



#### 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	Function
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
	Lights 200 ms once every 3 seconds	It is necessary to replace the battery of RTC



LED	LED Status	Function
	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)
Output status indicators (red-green)	Green	Output is closed
	OFF	Output is open

Under the module faceplate there are terminal blocks and a service button ([Figure 4.1](#)).

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)
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:

## 5 Costruction and operation

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:

## 5 Costruction and operation

- 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 is read from the device using the akYtecToolPro program in the “Device Parameters” tab. This list of Modbus registers is 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		
Mode for output DO1	0 – switching of a logic signal; 1 – PWM	272	0x110	Read and write	UINT 16
Mode for output DO2	0 – switching of a logic signal; 1 – PWM	273	0x111	Read and write	UINT 16
Mode for output DO3	0 – switching of a logic signal; 1 – PWM	274	0x112	Read and write	UINT 16
Mode for output DO4	0 – switching of a logic signal; 1 – PWM	275	0x113	Read and write	UINT 16

Parameter	Value (unit)	Register address		Access	Data format
		DEC	HEX		
Mode for output DO5	0 – switching of a logic signal; 1 – PWM	276	0x114	Read and write	UINT 16
Mode for output DO6	0 – switching of a logic signal; 1 – PWM	277	0x115	Read and write	UINT 16
Mode for output DO7	0 – switching of a logic signal; 1 – PWM	278	0x116	Read and write	UINT 16
Mode for output DO8	0 – switching of a logic signal; 1 – PWM	279	0x117	Read and write	UINT 16
Period of PWM generator on output DO1	1000...60000 (ms)	308	0x134	Read and write	UINT 16
Period of PWM generator on output DO2	1000...60000 (ms)	309	0x135	Read and write	UINT 16
Period of PWM generator on output DO3	1000...60000 (ms)	310	0x136	Read and write	UINT 16
Period of PWM generator on output DO4	1000...60000 (ms)	311	0x137	Read and write	UINT 16
Period of PWM generator on output DO5	1000...60000 (ms)	312	0x138	Read and write	UINT 16
Period of PWM generator on output DO6	1000...60000 (ms)	313	0x139	Read and write	UINT 16
Period of PWM generator on output DO7	1000...60000 (ms)	314	0x13A	Read and write	UINT 16
Period of PWM generator on output DO8	1000...60000 (ms)	315	0x13B	Read and write	UINT 16
Duty cycle of PWM-generator for output DO1	0...1000 (0.10%)	340	0x154	Read and write	UINT 16
Duty cycle of PWM-generator for output DO2	0...1000 (0.10%)	341	0x155	Read and write	UINT 16
Duty cycle of PWM-generator for output DO3	0...1000 (0.10%)	342	0x156	Read and write	UINT 16
Duty cycle of PWM-generator for output DO4	0...1000 (0.10%)	343	0x157	Read and write	UINT 16

Parameter	Value (unit)	Register address		Access	Data format
		DEC	HEX		
Duty cycle of PWM-generator for output DO5	0...1000 (0.10%)	344	0x158	Read and write	UINT 16
Duty cycle of PWM-generator for output DO6	0...1000 (0.10%)	345	0x159	Read and write	UINT 16
Duty cycle of PWM-generator for output DO7	0...1000 (0.10%)	346	0x15A	Read and write	UINT 16
Duty cycle of PWM-generator for output DO8	0...1000 (0.10%)	347	0x15B	Read and write	UINT 16
Bit mask of the outputs DO1-DO8 state	0...255	468	0x1D4	Read and write	UINT 16
Bit mask of setting the outputs DO1-DO8 state	0...255	470	0x1D6	Read and write	UINT 16
Safe state value for output DO1	0...1000 (0.10 %)	474	0x1DA	Read and write	UINT 16
Safe state value for output DO2	0...1000 (0.10 %)	475	0x1DB	Read and write	UINT 16
Safe state value for output DO3	0...1000 (0.10 %)	476	0x1DC	Read and write	UINT 16
Safe state value for output DO4	0...1000 (0.10 %)	477	0x1DD	Read and write	UINT 16
Safe state value for output DO5	0...1000 (0.10 %)	478	0x1DE	Read and write	UINT 16
Safe state value for output DO6	0...1000 (0.10 %)	479	0x1DF	Read and write	UINT 16
Safe state value for output DO7	0...1000 (0.10 %)	480	0x1E0	Read and write	UINT 16
Safe state value for output DO8	0...1000 (0.10 %)	481	0x1E1	Read and write	UINT 16
Safe state activation timeout	0...60 (s)	700	0x2BC	Read and write	UINT 8
Battery status (power supply)	0...3300 (mV)	801	0x321	Read and write	UINT 16
Log interval	10...3600 (s) default value – 30	900	0x384	Read and write	UINT 16
Time (ms)	—	61563	0xF07B	Read only	UINT 32
New time	Date/Time in seconds from January 1, 2000.	61565	0xF07D	Read and write	UINT 32
Write the new time	0 – do not save;	61567	0xF07F	Read and write	UINT 16

Parameter	Value (unit)	Register address		Access	Data format
		DEC	HEX		
	1 – save				
Time and date (UTC)	Date/Time in seconds from January 1, 2000.	61568	0xF080	Read only	UINT 32
Time zone	Offset in minutes from Greenwich	61570	0xF082	Read and write	INT 16
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 1	—	14	0xE	Read and write	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
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
DHCP Mode	0 – absolute prohibition 1 – read only 2 – write only	32	0x20	Read and write	UINT 16

### 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>



Function	Error name	Possible causes
	MODBUS_ILLEGAL_DATA_VALUE	<ul style="list-style-type: none"> <li>– no terminating character (�) 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	– illegal data size (0x07 <= data length <= 0xF5)
	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	– could not move to the desired offset in the file
	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 outputs operation modes

Each digital output can operate in one of the following modes:

- switching of a logic signal;
- PWM signal generation.

You can change the output operation mode in the “akYtec Tool Pro” program or by writing values to the corresponding Modbus registers (see [Section](#) ).

### 5.7 Safe output status

A safe output status is possible for each output.

The output switches to the safe state, if at the time of turn-on or during the time-out time there are no commands from the Network Master. The value of the **Safe status** is set in percentage (from 0 to 100%) at the module output. This value determines the duty ratio of the PWM.



Time-out is set by user. If the time-out is set to **0**, safe output state is off. When the Module is switched on, it will go into the state that was last set before shutdown, and will remain in it until a new command is received from the Network Master.

**NOTE**

The factory setting for **Safe state activation timeout** is activated is **30 seconds**.

## 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.  
If the Module is powered with USB, the outputs do not function.  
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

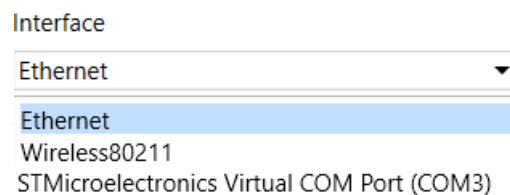


Fig. 6.1 Interface selection menu

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.

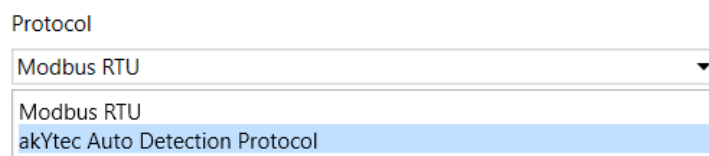


Fig. 6.2 Select Protocol

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.

## 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 with a 3 second interval).
- 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 *Figure 7.1*.

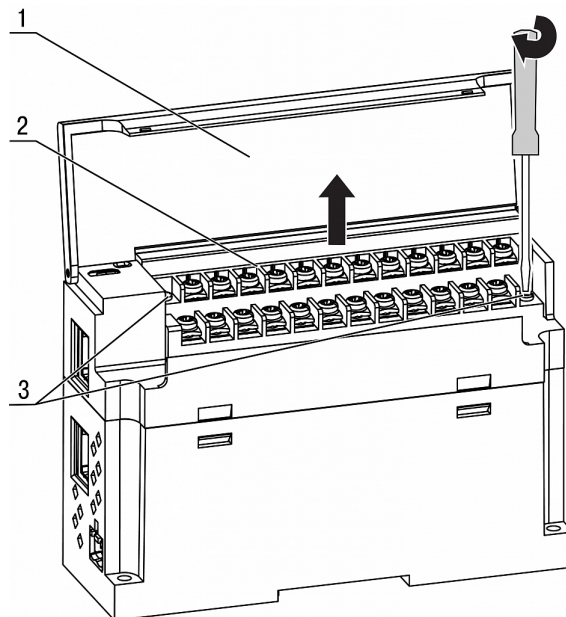


Fig. 7.1 Remove terminal block

6. Alternately remove the hooks from the holes on one side and on the other side and remove the top cover.

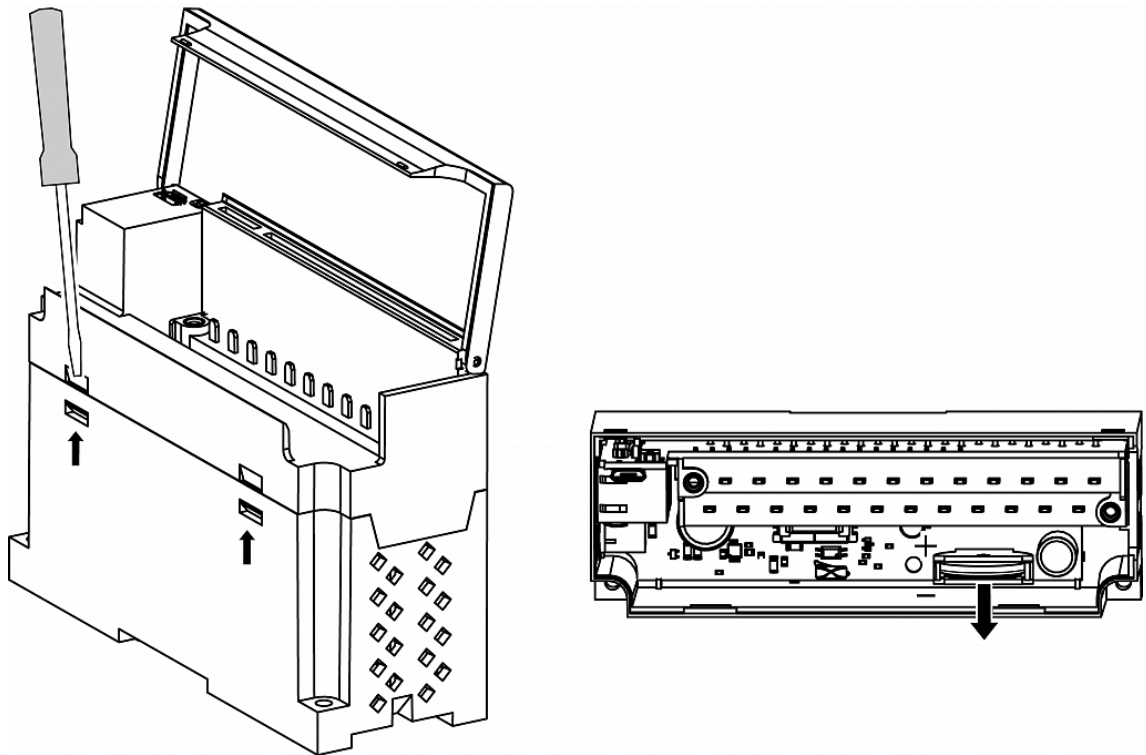


Fig. 7.2 Battery replacement

7. Replace the battery. 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.
8. Assembly and installation should be carried out in the reverse order.

**NOTICE**

Do not use a different type of battery. When installing the battery, observe the polarity.

After 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.

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

### 8 Transportation and storage

Device must be transported in closed transport of any kind. The fastening of containers during transport should be carried out in accordance with the rules applicable to the respective modes of transport.

The transport conditions must be in accordance with IEC 61131-2-2012 at ambient temperature from – 40 to + 55 °C in compliance with the protection measures against impacts and vibrations.

Transportation should be carried out in a shipping container individually or in over-packs.

The storage conditions in the packaging at the manufacturer's and consumer's warehouse must comply with IEC 61131-2-2012. In the air no aggressive impurities must be present.

Device should be stored in the racks.



### 9 Scope of delivery

Denomination	Quantity
Device	1 unit
Short guide	1 ex.
UTP patch cable 5e 150 mm	1 unit
Power supply terminal 2EGTK-5-02P-14	1 unit
Ethernet connector plug	1 unit

**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;
}
```