Inveo Sp. z o.o.

IQIO IO

Instruction Manual

Overview

Warning

This manual applies only to the firmware version v0.37.0 onwards. Inveo does not guarantee that the information contained in this document applies to previous firmware revisions.



Purpose of the device

The IQIO IO is a versatile controller that enables monitoring, management and control of various devices and processes.
In addition to supporting various communication protocols (such as HTTP, MQTT, TCP/UDP) and time synchronisation functions, IQIO IO also offers support for one input and one output. This means that it can control one input device, for example a temperature or motion sensor, and one output device, for example a relay controlling a light or a fan.

The ability to configure notifications and the event system make the IQIO IO applicable in various fields such as industrial automation, building management, environmental monitoring and security systems. It is a reliable tool for remote control and process optimisation, meeting the needs of both individual users and businesses.

Changelog

Table of contents

- Overview
  - Purpose of the device
  - Changelog
- Device construction
  - Technical characteristics
    - Module connectors description
- Network configuration
  - Changing the device's IP address - Discoverer programme
  - Changing the computer's subnet address for configuration
  - Network settings configuration
  - Configuration mode
  - Wi-Fi connection configuration
- Device configuration
  - Module's www interface
  - Device status preview
    - Outputs window
    - Inputs window
  - Inputs/outputs configuration
    - Outputs
      - Last output state or counter retention
    - Inputs
    - Display LED
  - Notifications configuration
    - Inputs
    - Outputs
    - Configuration
  - Binding
    - Outputs
  - Logic functions
    - Conditions
    - Variables
  - System management
    - Access
      - API keys

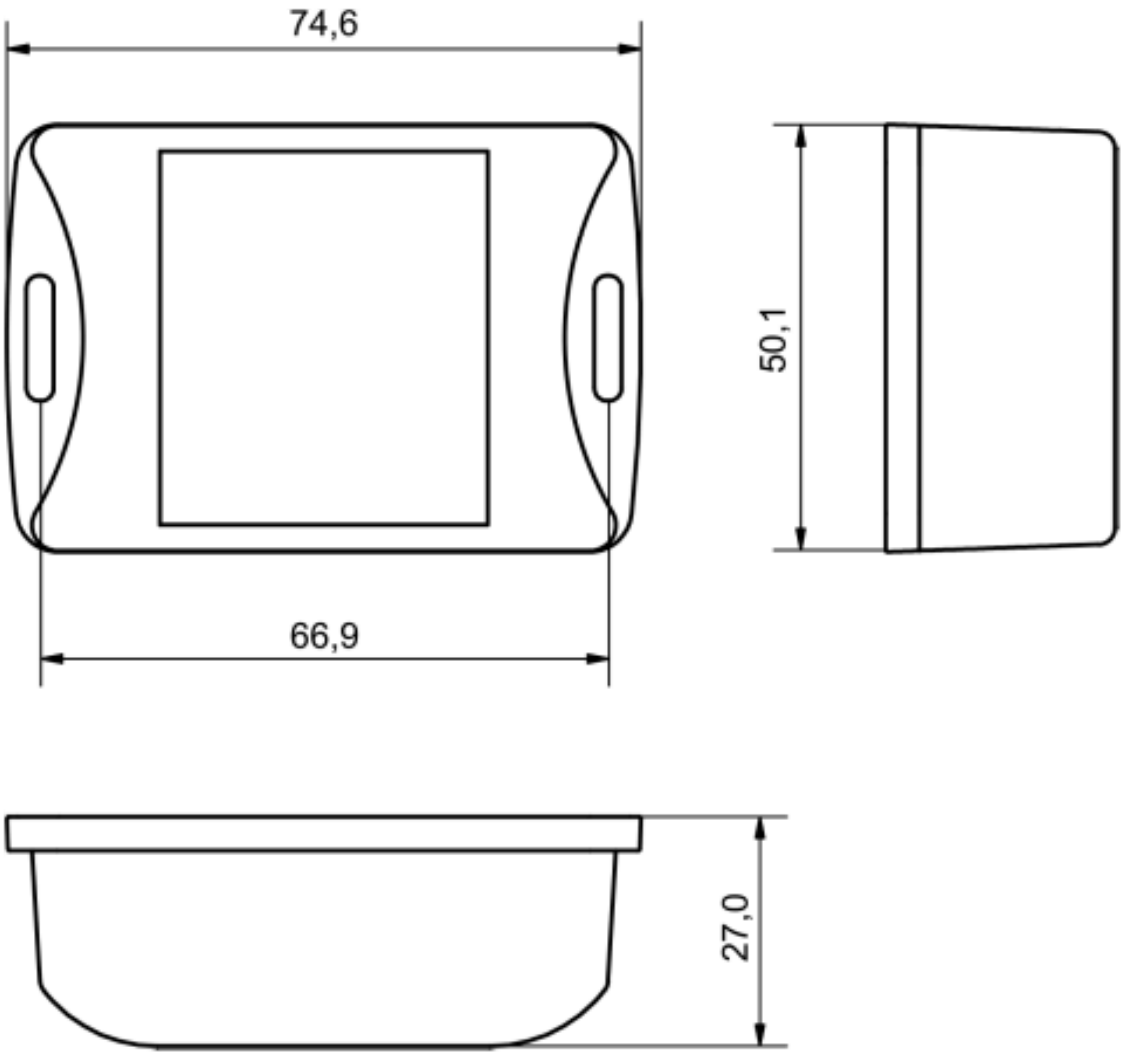
        - Network
        - System events
        - Backup
        - Restore
- Defining actions
    - All
    - Protocol configuration
        - UDP
        - TCP
        - HTTP
        - MQTT
        - IO
        - SNMP Trap
- Communication protocols
    - Web
    - HTTPc
    - MQTT
    - SNMP
        - SNMP v2c
    - TCP/UDP
    - iCluster
- Built-in variables
    - REST API
- Factory settings, backup
- Emergency firmware restore / restoring factory settings
- Warranty and manufacturer's liability
    - Storage, operation and transport conditions
    - Disposal and decommissioning

Device construction

Technical characteristics

Parameter	Description
Power supply	**10-24V DC** with a 3,5mm pluggable terminal block **PoE:** 33-57V PoE IEEE 802.3af *(IQIO IO ETH)* **USB-C** 5V
Power consumption	Max. 1,5W
Inputs	**Input type:** digital, normally open, pre-polarized
Outputs	**Output type:** potential-free NO relay output **Maximum operating voltage:** 30V DC **Maximum relay load:** 1A NO output (normally open) **Activation time:** 1ms **Deactivation time:** 5ms **Operating mode:** bistable, astable, monostable, timed
Komunikacja	Wi-Fi 2,4GHz *(IQIO IO Wi-Fi)* Ethernet 10/100Mbps Full Duplex, RJ45 port *(IQIO IO ETH)*
Enclosure ingress protection	IP30
Weight	ca. 71g (2,5oz)

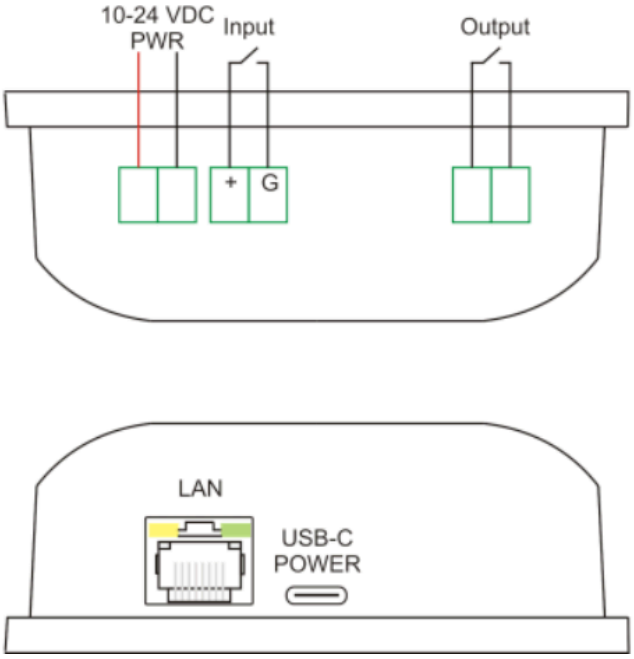
Dimensions



Information

All dimensions are in millimeters.

Module connectors description



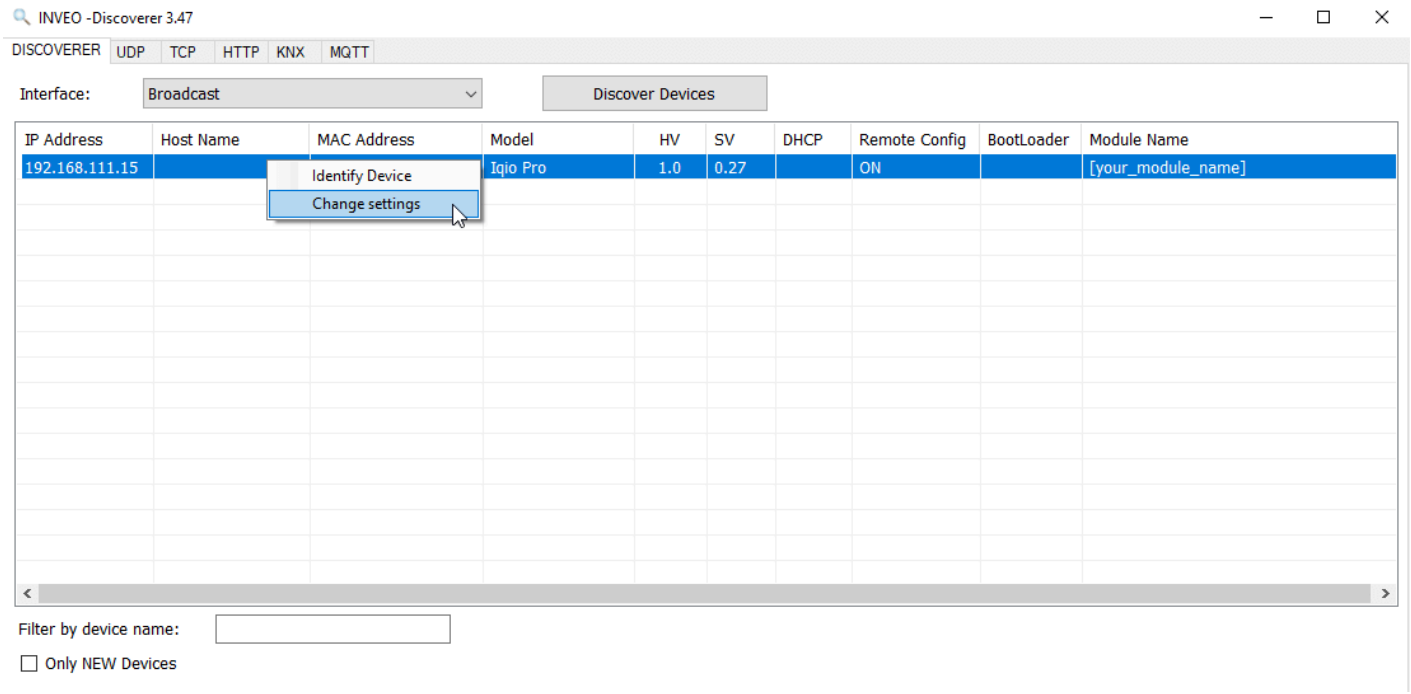
The IQIO IO module has the following connectors:

- **RELAY OUT** – Relay's pluggable terminal block,
- **IN** – Digital input,
- **LAN** – LAN and PoE IEEE 802.3af,
- **PWR 5VDC** – 5V USB-C power connector.

Network configuration

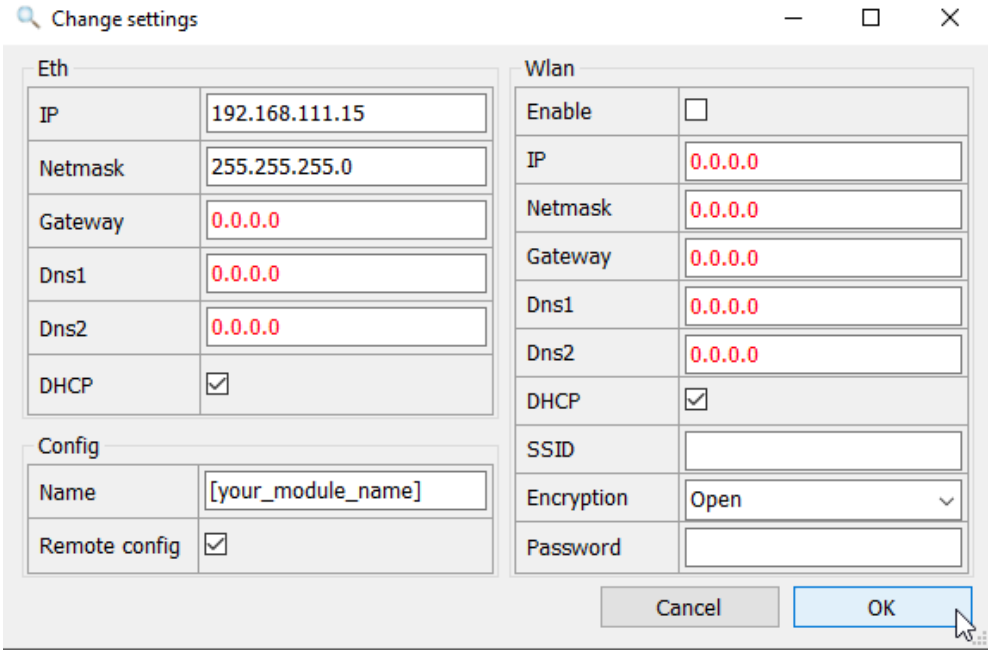
> **Warning**
>
> **After the initial power-up, it's needed to configure the device. It can be done in two ways, the simplest method is to use the Inveo Discoverer programme.**

Changing the device's IP address - Discoverer programme



After running the **Discoverer** programme and searching for the appropriate device:

1. Right-click the device,
2. Click the **Change** button.



After the dialogue window is opened you can:

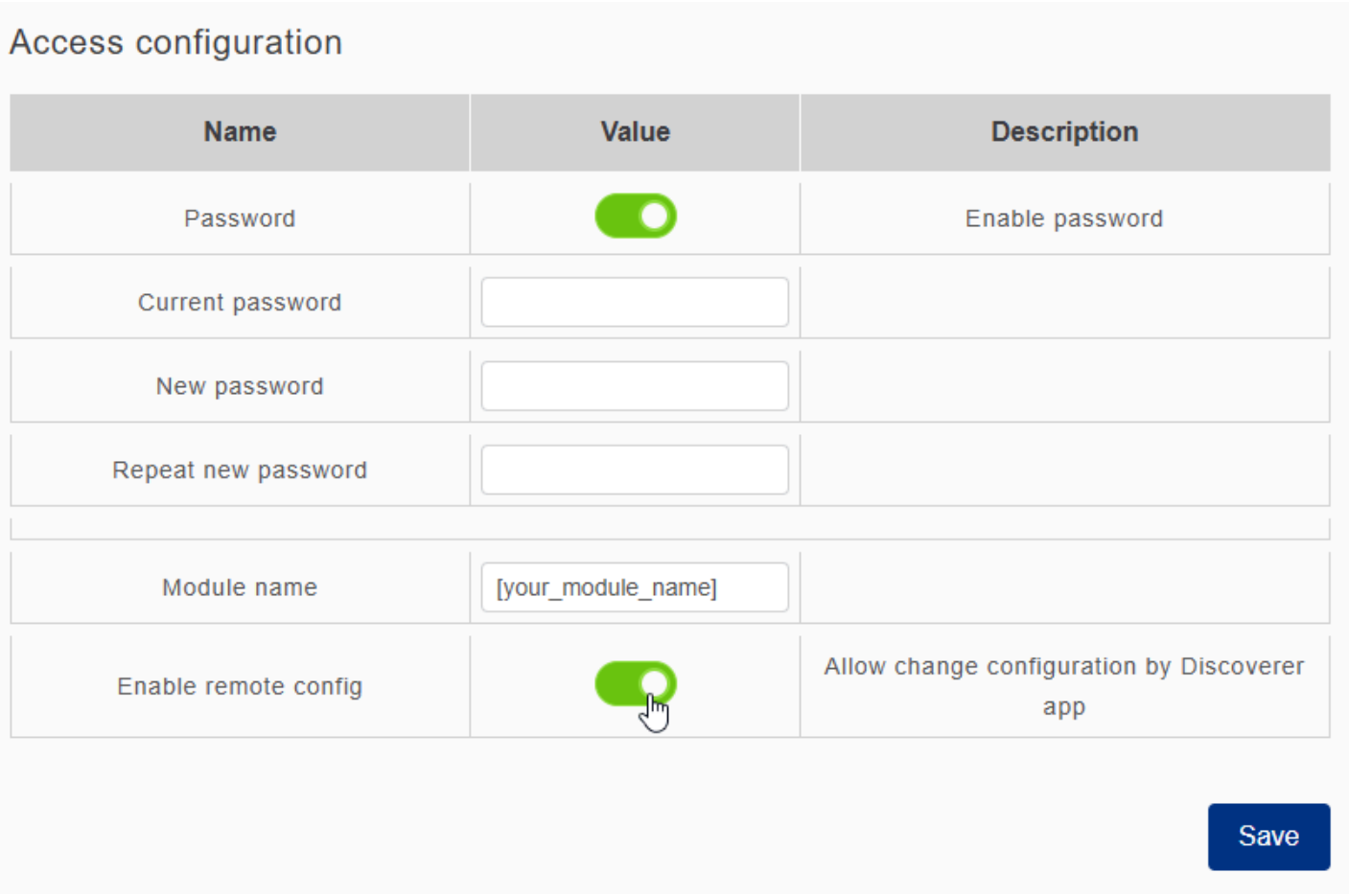
- Set the IP address, mask, gateway, DNS1/DNS2.
- Change the Host name.

> **Note**
>
> **The device will be configured after the Change button is pressed.**

If **Remote Config** is disabled (enabled by default), the device needs to be configured by changing the computer's subnet:

To enable remote configuration:

1. Go to the **Administration** tab,
2. In the **Access configuration** window, select **Enable Remote Config**.



> **Tip**
>
> **After making changes, apply them using the Save button**

Changing the computer's subnet address for configuration

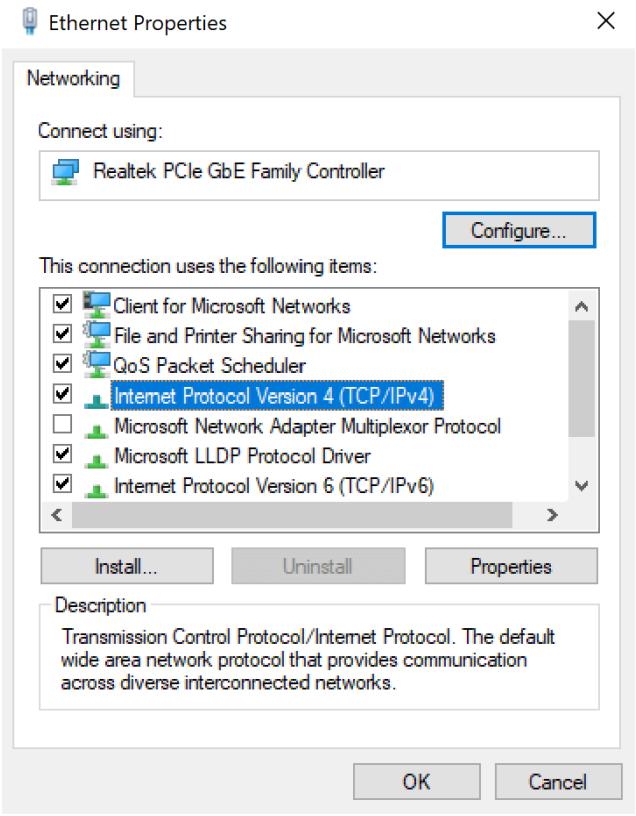
During configuration without **Discoverer** programme, it's needed to change the subnet address of the computer on the same network.

To access the computer's network configuration, do one of the following:

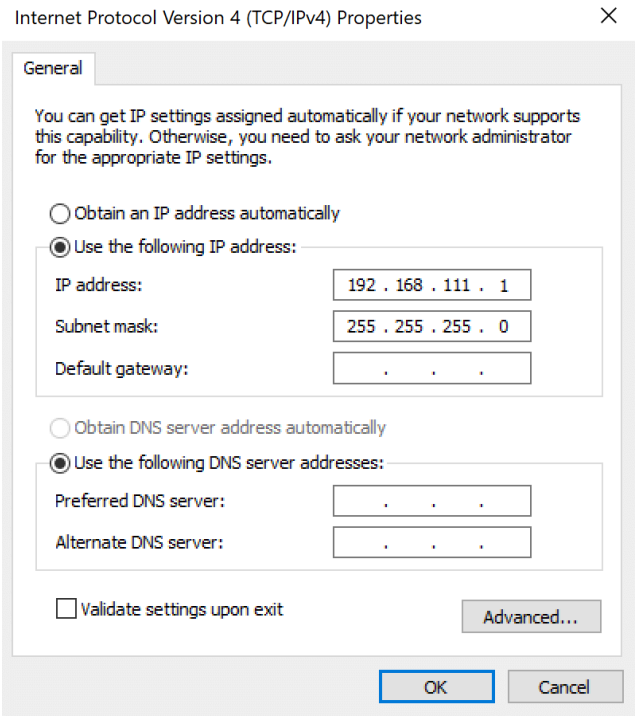
- Press **Win + R**, type in `ncpa.cpl`, and then press **Enter**.
- Alternatively, go to:
  Start → **Control Panel** → **Network and Internet** → **Network and sharing center** → **Change adapter settings**.

Then:

1. Select the network connection.
2. Right-click it and select **Properties**.
3. After selecting this option, network connection configuration screen will appear.



Select **Internet protocol version 4 (TCP/IPv4)** and type in the following parameters:

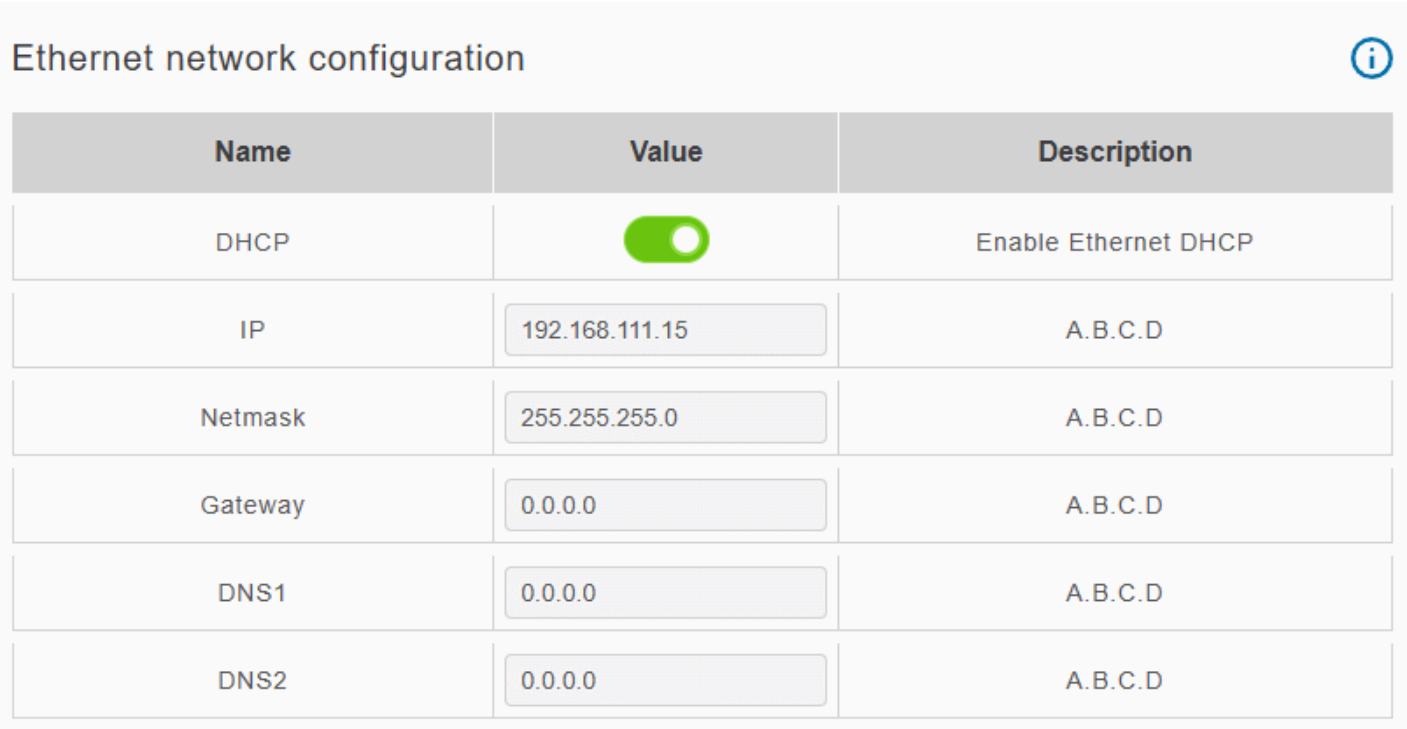


Note

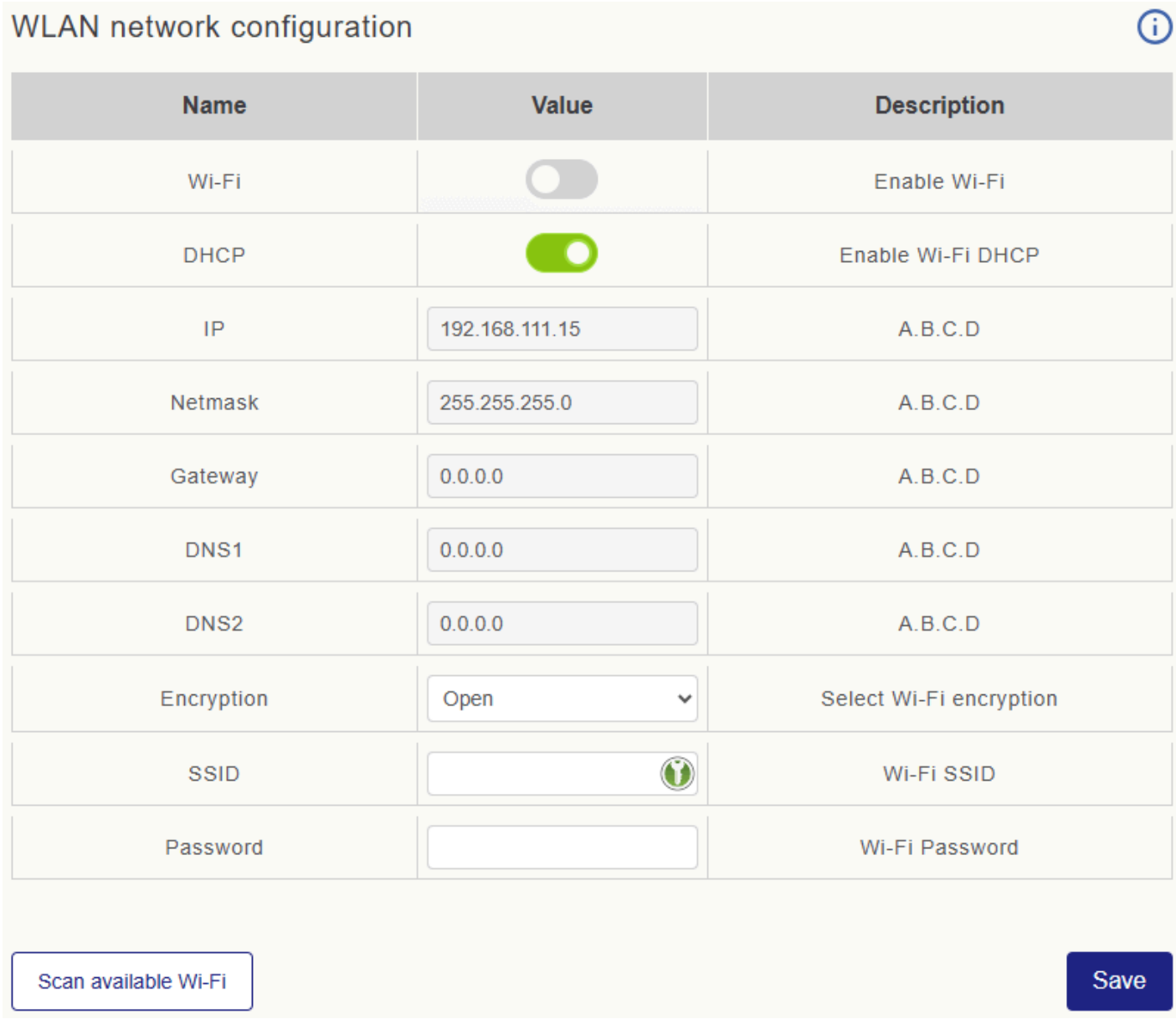
After applying the configuration with OK, start a web browser and type in the following address: 192.168.111.15. (Default user and password: admin/admin)

Network settings configuration

To customise the device's network settings, go to the **Administration / Network** tab. Parameters such as the IP address, subnet mask, gateway, DNS, and other network-specific options can be configured here. This tab allows the user to configure both the wired (**Ethernet network configuration**) and wireless (**WLAN network configuration**) connections.



- **DHCP** – Enable/disable DHCP server,
- **IP** – Device IP address,
- **Netmask** – IP subnet mask,
- **Gateway** – Network gateway,
- **DNS1, DNS2** – DNS server addresses.



- **Wi-Fi** – Enable/disable Wi-Fi service,
- **DHCP** – Enable/disable DHCP server in Wi-Fi,
- **IP** – Device IP address,
- **Netmask** – IP subnet mask,
- **Gateway** – Network gateway,
- **DNS1, DNS2** – DNS server addresses,
- **Encryption** – Wi-Fi encryption type:
  - Open,
  - WEP,
  - WPA-PSK,
  - WPA2_PSK,
  - WPA_WPA2_PSK,
  - WPA3_PSK,

- **SSID** – Network name,
- **Password** – Wi-Fi access password.

Scan available Wi-Fi allows the device to search and display a list of available Wi-Fi networks in range of the device.

Configuration mode



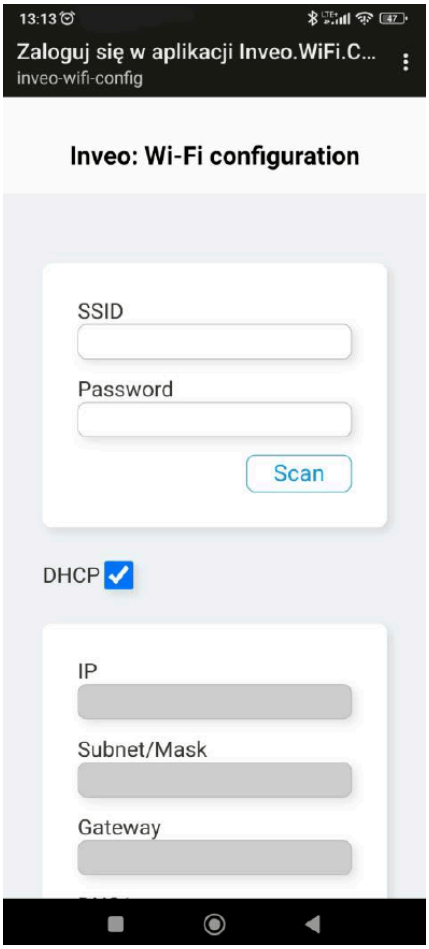
Pressing and holding the **RESET** button will display the IP address.

Configuration mode – After the power supply is connected to the device, changing and preview o certain settings is available for 3 minutes. Press and hold the **RESET** button to display the following information sequentially:

- **IP** – Device IP address,
- **dhcp eth** – If the **RESET** button is released while this prompt is displayed, DHCP will be enabled/disabled,
- **AP** – Release of **RESET** will allow the device's Wi-Fi to be configured,
- **rst def** – Release of the button will cause the device to reset to factory settings.

No changes will be made if the **RESET** button is released in the pause between prompts or after the last prompt has been displayed.

Wi-Fi connection configuration



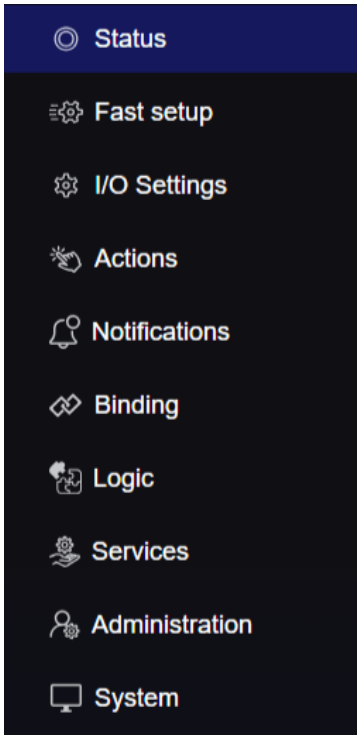
1. For 3 minutes after the power has been applied to the device (during **Configuration mode**), Wi-Fi connection can be configured. To do that, press and hold the **RESET** button until **AP** is displayed.
2. Activate Wi-Fi network search on your smartphone or other device. A network called "Inveo-wifi-config" should appear on the list.
3. Connect with the network.
4. After connection is established, click **scan** in the configuration interface or enter the Wi-Fi SSID into the correct field.
5. Select the network the device should connect to from the list of available options.
6. Enter the correct password to the selected network.
7. If the DHCP server is unavailable, manual configuration can be conducted after de-selecting **DHCP**.
8. **SUCCESS** will appear if the settings are saved successfully. The device will restart and resume normal operation.

Device configuration

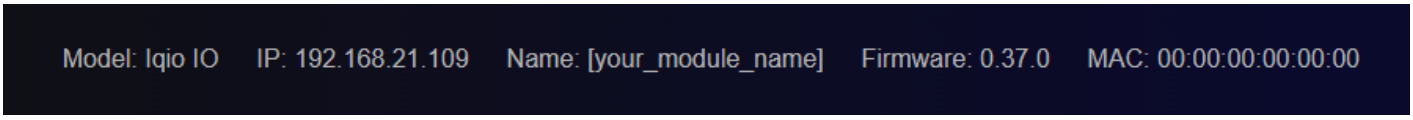
Module's www interface

The web interface allows for intuitive and advanced device management. After typing in the device's IP address into the browser's search bar, a webpage opens allowing full configuration and customisation of the device's operating parameters according to the individual user needs. On the left-hand side of the screen, a list of tabs is located, allowing for quick access to various functions and settings. Available tabs:

> **Note**
>
> **Some tabs are only visible after activationg the Advanced view.**



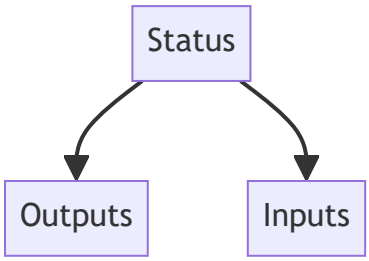
In the upper portion of the website, an information ribbon is located, informing the user about key device data, such as: model, IP address, unique user-assigned name, firmware version, and MAC address.



Thanks to the website, the user can modify settings, configure parameters, and monitor device operation in real-time. The webpage is the central control point, allowing for effective management and customization according to the changing user needs.

Device status preview

In the **Status** tab, all information about currently used outputs, inputs and sensor readings can be found.



Outputs window

In this window, current output state is displayed. Outputs can be defined in the **I/O Settings** tab.

Outputs

Name	Off/On	Coil state
DO 0		
VO 1		
VO 2		

Individual table columns contain the following data:

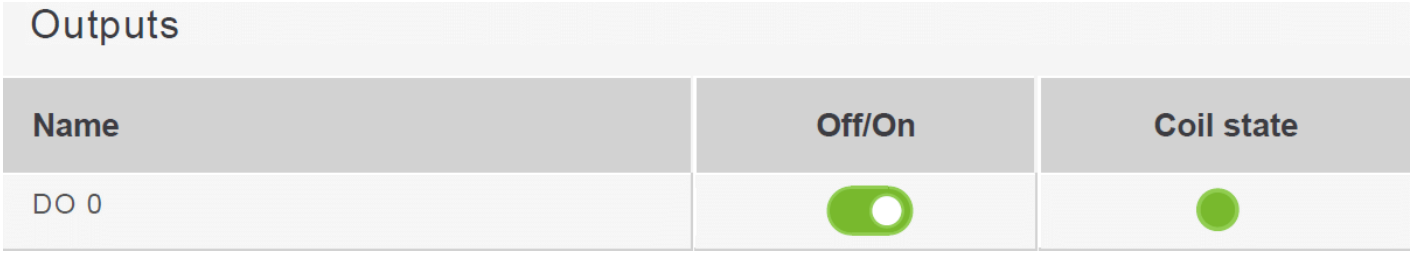
- **Name** – Output name defined in the **I/O Settings** tab.

The following input types are available:

- **Digital – DO** – Physical module outputs,
- **Virtual – VO** – Virtual outputs used for binding and relations configuration.

If the output state is dependent upon different factors, appropriate information is shown under its name:

- **output unavailable – assigned to the shutter**,
- **output unavailable – output is routed** – See section **Binding**.
- **Off/On** – Current output state, clicking the left mouse button in this area will change the output state - this option enables manual output control,
- **Coil state** – Current relay coil state - green colour means that the relay is activated.



Output activation (**Off/On** column in the table) is not always identical as the **Coil state**.

Example

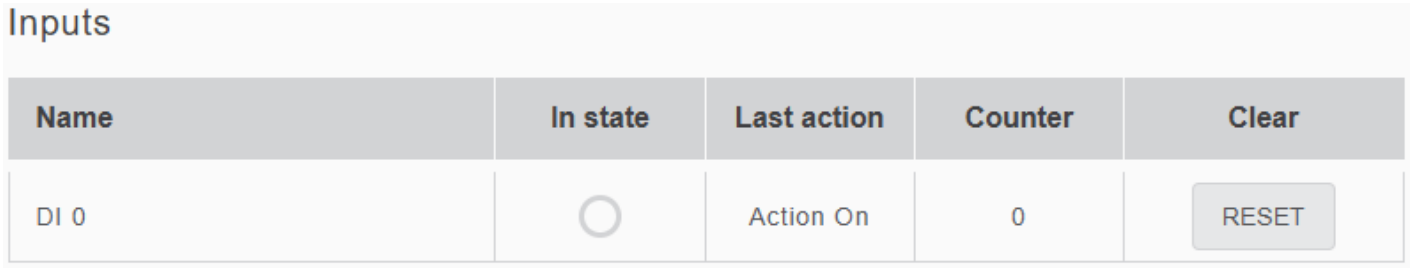
If the output is configured in astable mode, with Time on and Time off parameters defined, enabling the output in the Status tab changes the state in the Off/On column. In this case we can observe the coil state is alternately activated/deactivated, according to the parameters defined in Time on / Time off.

Tip

If the output is configured for shutter control or other input/output state has been bound to it, testing its operation in the Off/On column will not be possible.

Inputs window

In this window, current input state is displayed. Inputs can be defined in the **I/O Settings** tab.

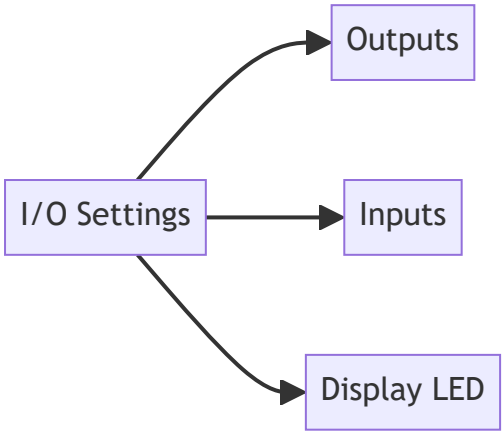


Individual table columns contain the following data:

- **Name** – Input name defined in the **I/O Settings** tab,
- **In state** – Input state,
- **Last action** – Last registered action (eg. Action On),
- **Counter** – Displays the amount of times the input has been activated since last reset,
- **Clear** – Resets the counter.

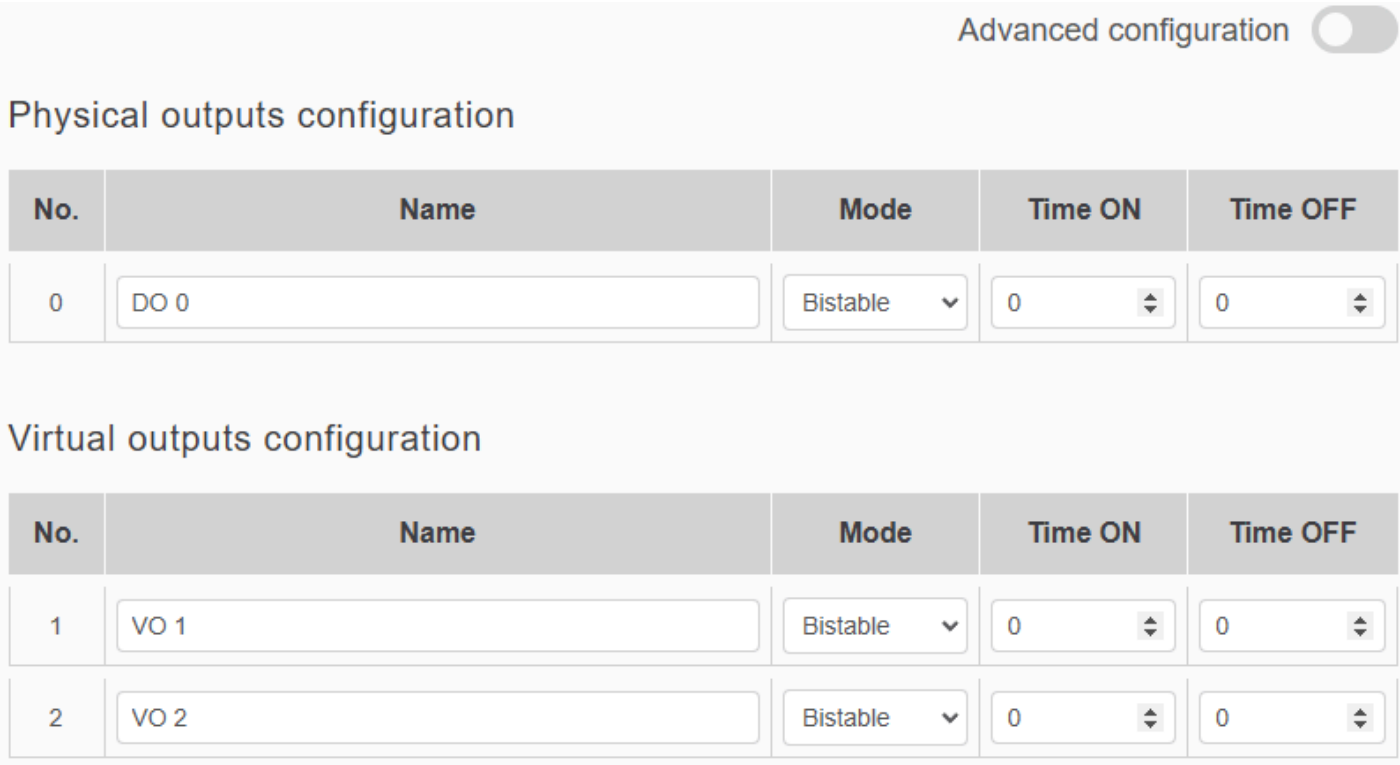
Inputs/outputs configuration

In the **I/O Settings** tab, advanced configuration options can be accessed, allowing the user to define device operation. Here, behaviour of inputs and outputs can be configured. Additionally, this tab gives users the option to configure the display.

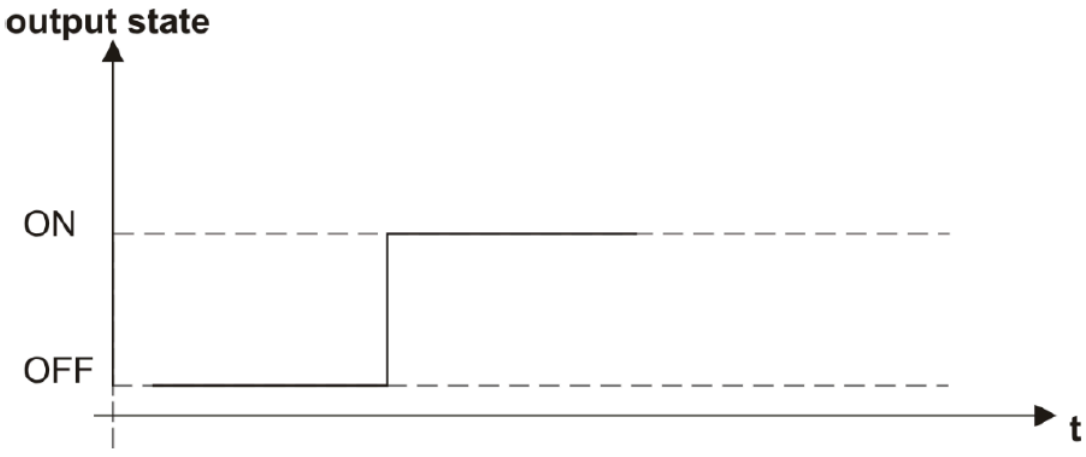


Outputs

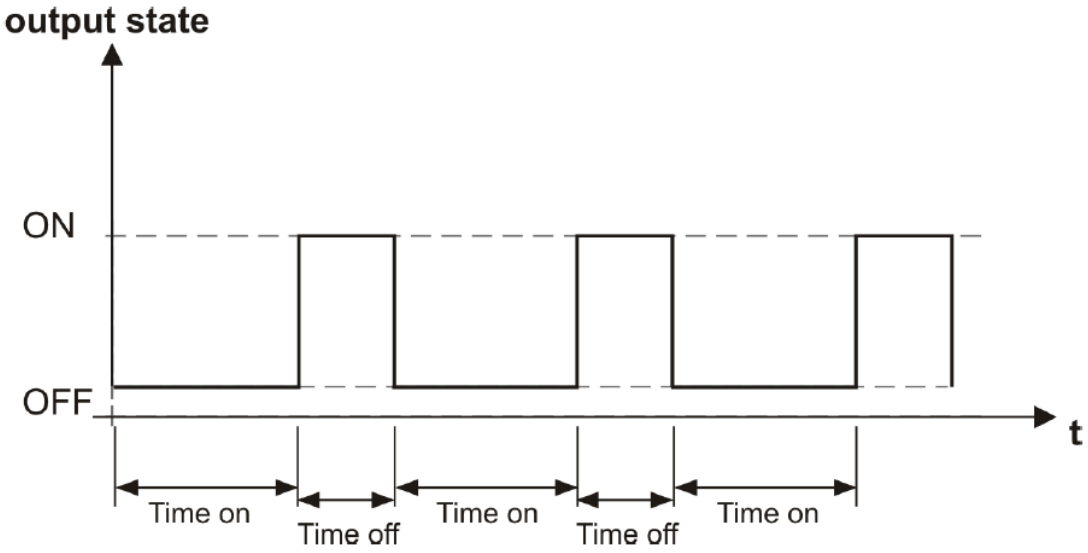
This tab allows configuration of outputs – both physical and virtual *(option)*. The following settings can be changed in individual columns:



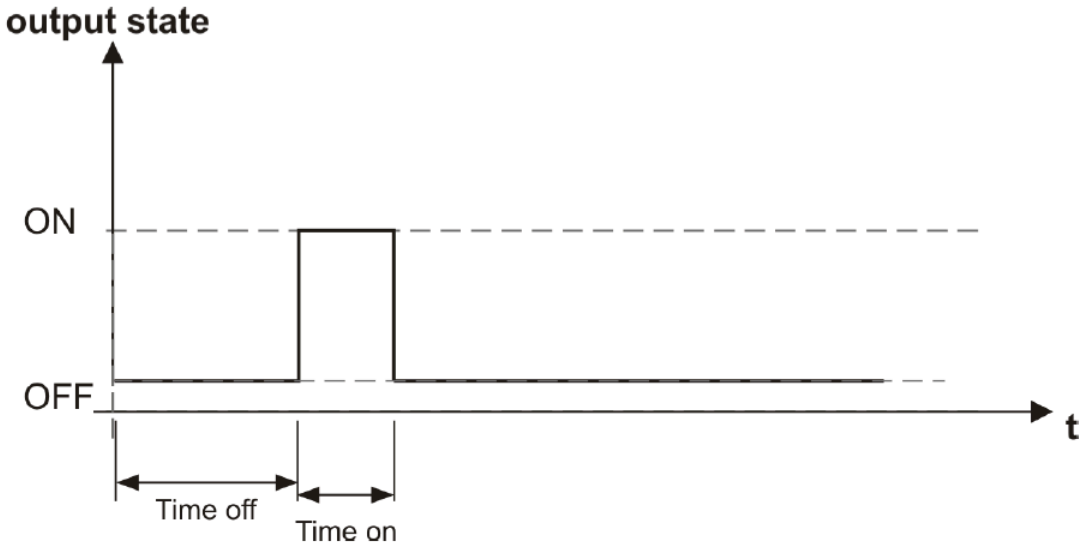
- **Name** – This field allows the output name to be changed,
- **Mode** – Output operation mode:
  - **Disable**,
  - **Bistable**,



  - **Astable** – Periodic mode - output is activated and deactivated periodically. The activation and deactivation times are defined by the parameters **Time on** and **Time off**,

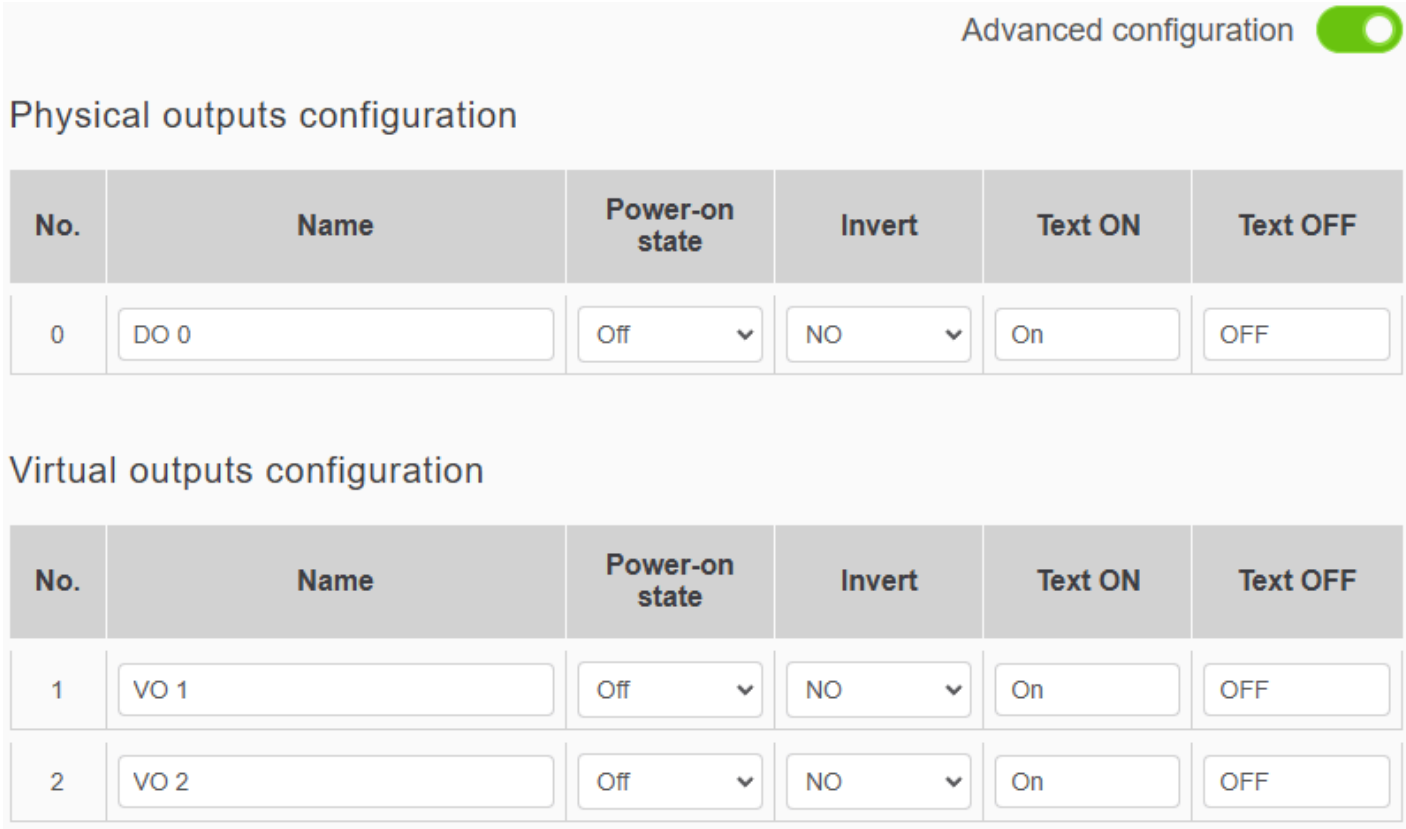


- **One-pulse** – Single output trigger mode - output is activated for a time defined in **Time on** after time defined in **Time off** passes,



- **Time on** – (parameter used in **Astable** and **One-pulse** mode),
- **Time off** – (parameter used in **Astable** and **One-pulse** mode).

Clicking **Advanced configuration** enables changing of advanced output settings:

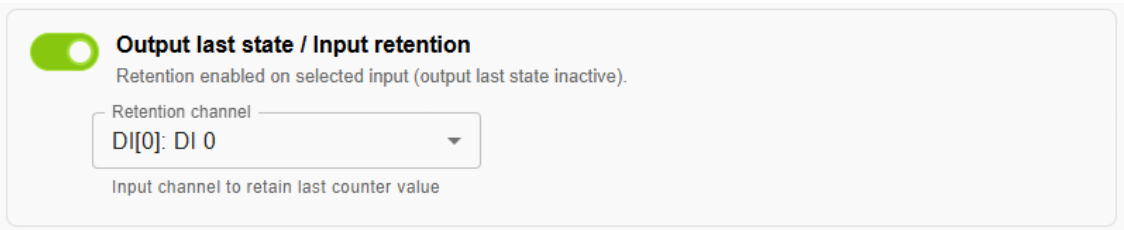


- **Name** – This field allows the output name to be changed,
- **Power-on state** – State that the device should reach after power is connected to the module:
  - **Off**,
  - **On**,
  - **Last** – State before the power was disconnected.
- **Invert** – Output channel inversion:
  - **NO** – Normally open,
  - **NC** – Normally closed.
- **Text ON** – Text dispayed on the LED display when the output is activated,
- **Text OFF** – Text dispayed on the LED display when the output is deactivated.

LAST OUTPUT STATE OR COUNTER RETENTION

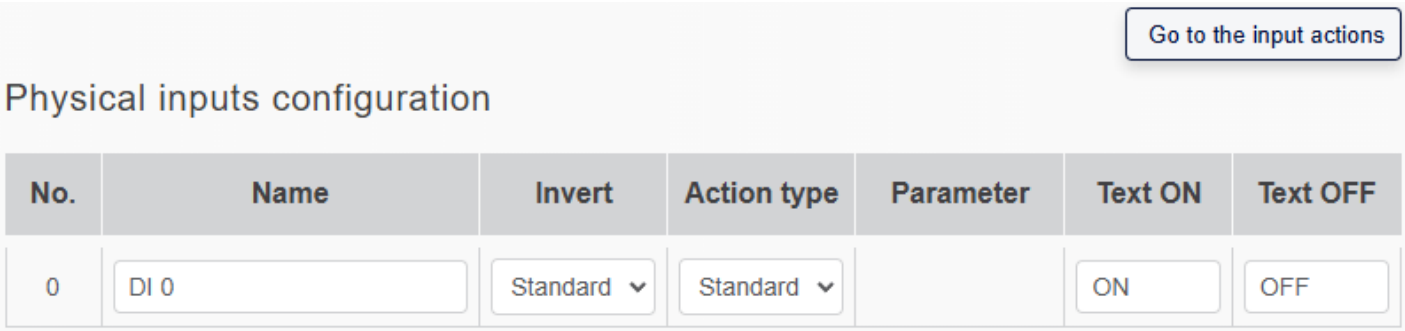
The module offers the possibility to store the last output or specified input counter state in non-volatile memory.

The **Ouptut last state / Input retention** slider enables selection of the desired setting.

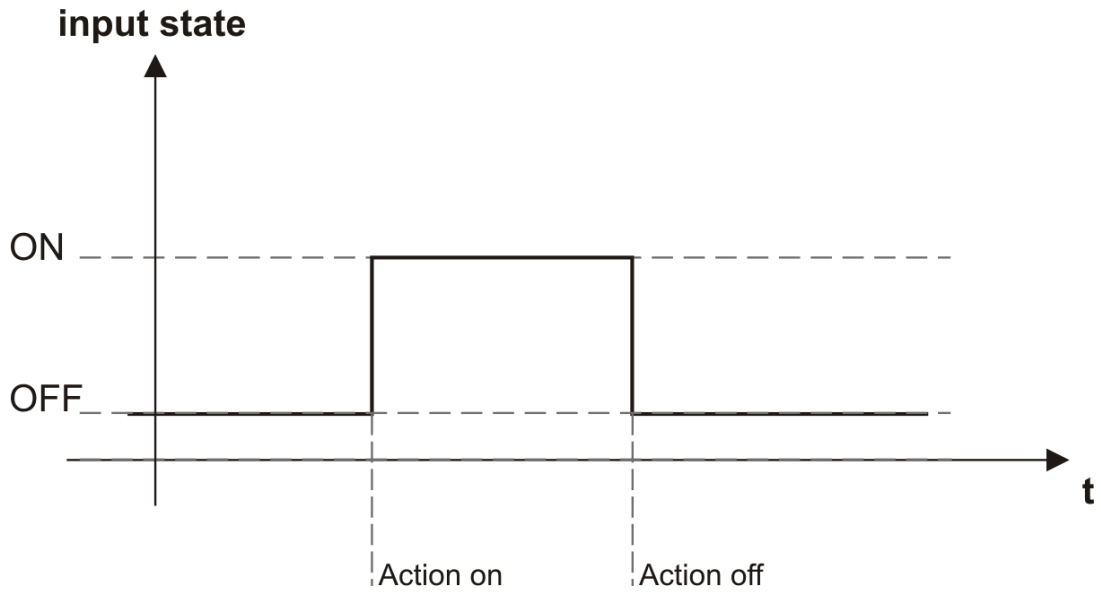


Inputs

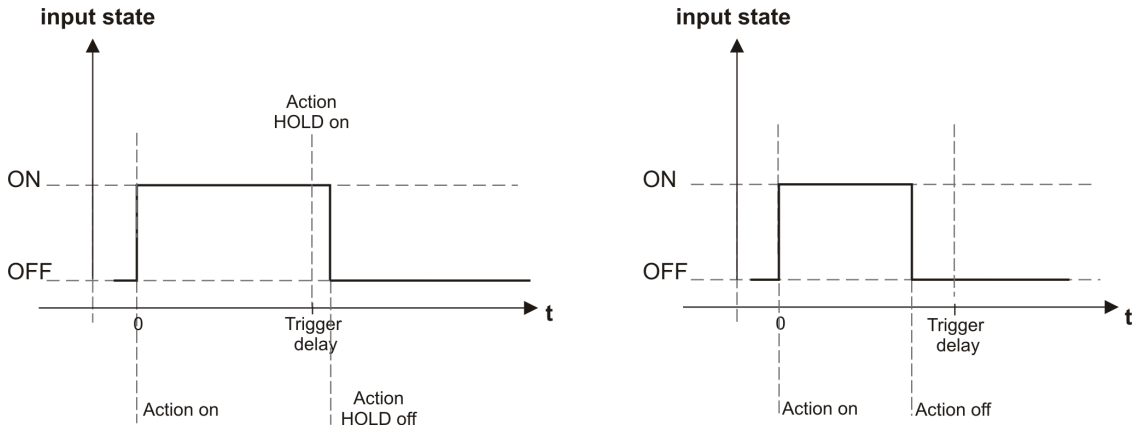
This tab allows configuration of inputs supported by the device. The following settings can be changed in individual columns:



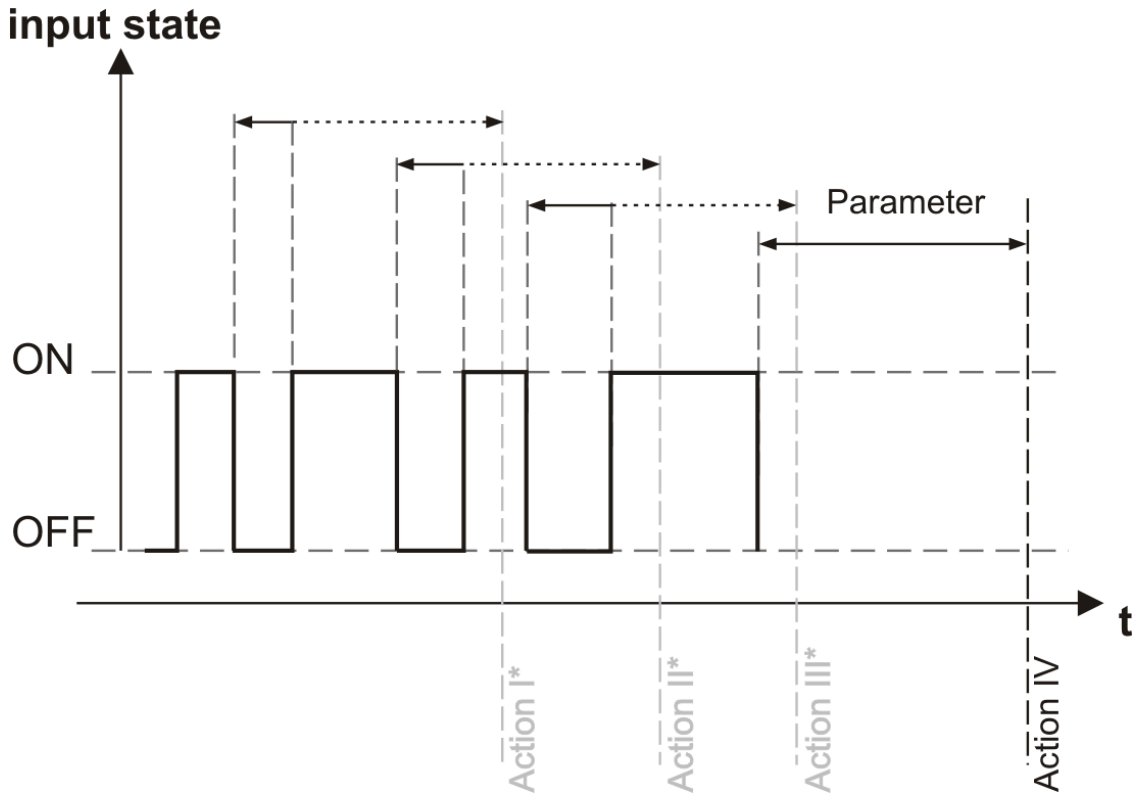
- **Name** – This field allows the input name to be changed,
- **Invert** – Input channel inversion,
- **Action type**:
  - **Standard** – Action is triggered by input activation/deactivation,



- **Hold** – Triggering of a defined action is dependent upon the pulse length on the input. The pulse recalls the action defined as **Action on**. If the pulse persists and exceeds the time in the **Parameter** field - action defined as **Action hold on** is recalled. If the pulse ceases before the time defined in **Parameter** - **Action off** is recalled. If the pulse ceases after the time defined in **Parameter** field - **Action hold off** is recalled,

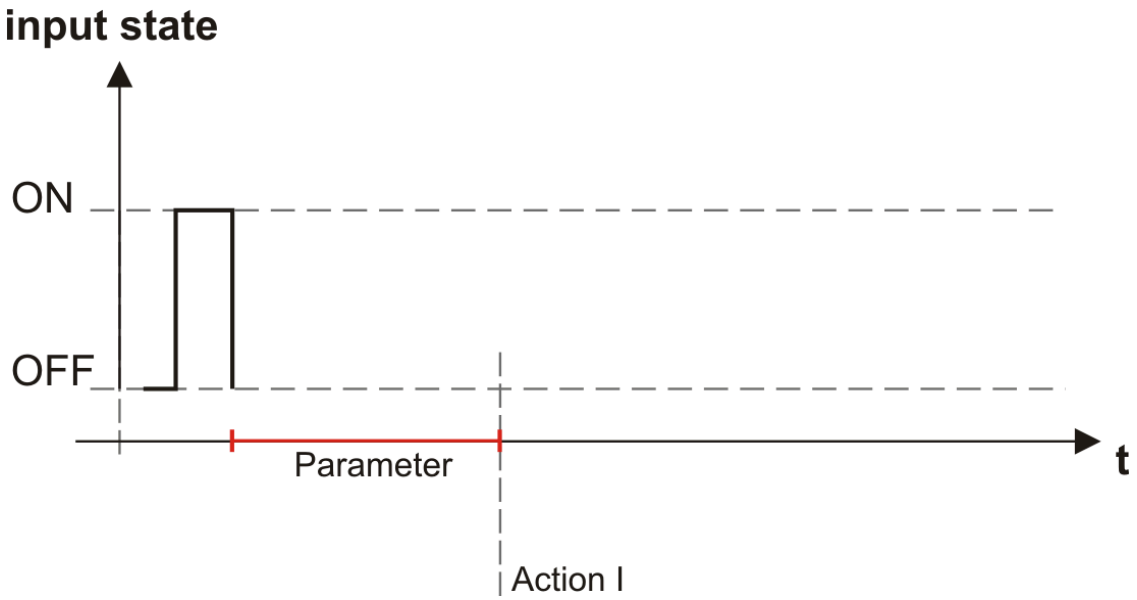


- **Cnt** – Pulse counter during specified time period - the action is recalled by the defined amount of activations in the time period specified by **Parameter**,

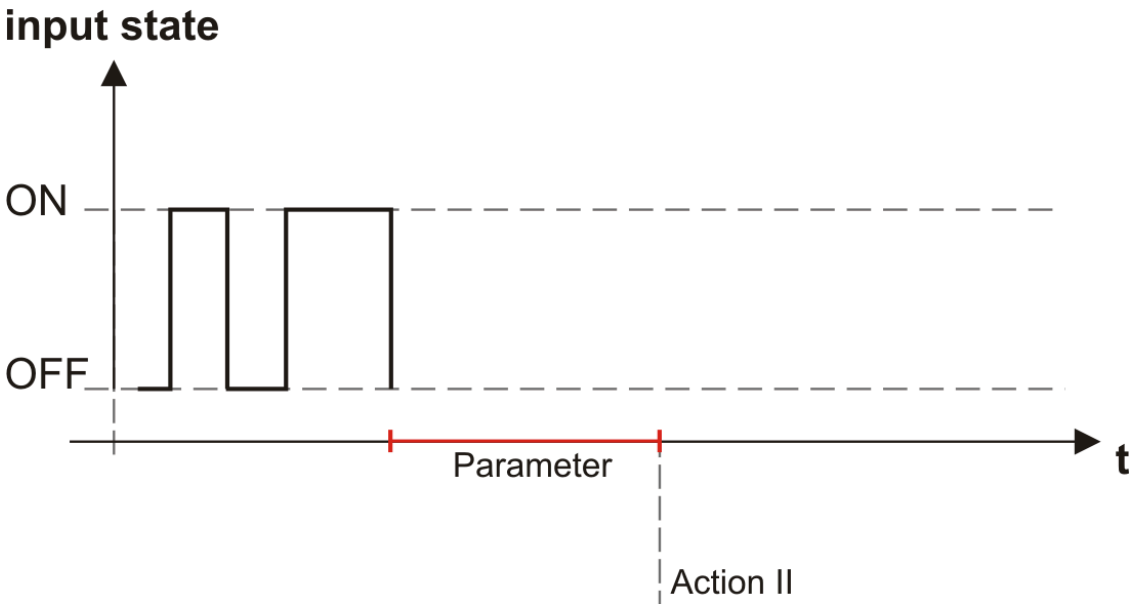


Action will be recalled only when no subsequent pulse occurs (input is not activated again). The defined action is recalled after the pulse absence delay time passes (i.e. a button is released). If another pulse occurs during this delay time, the countdown is stopped.

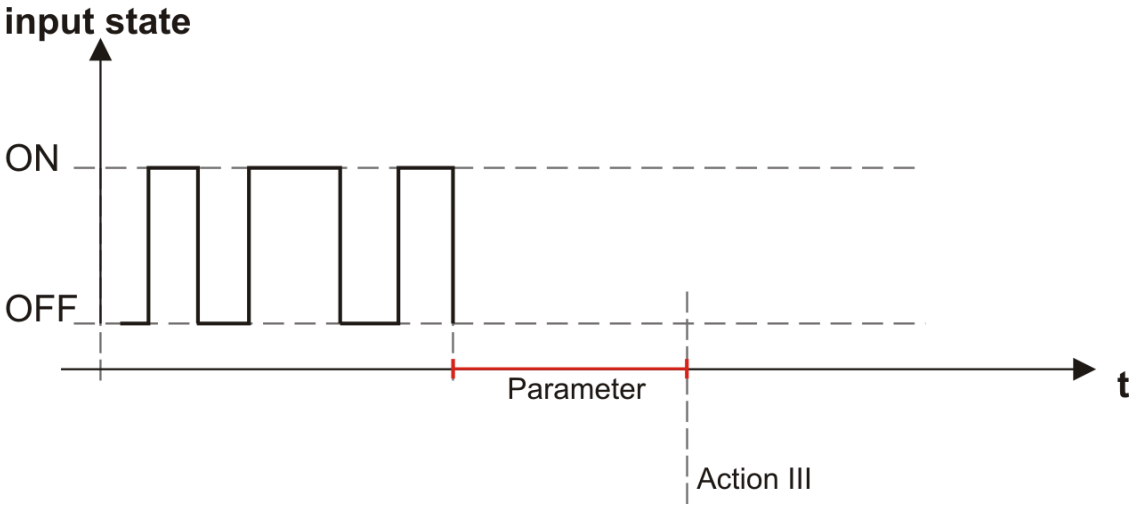
Recalling action I:



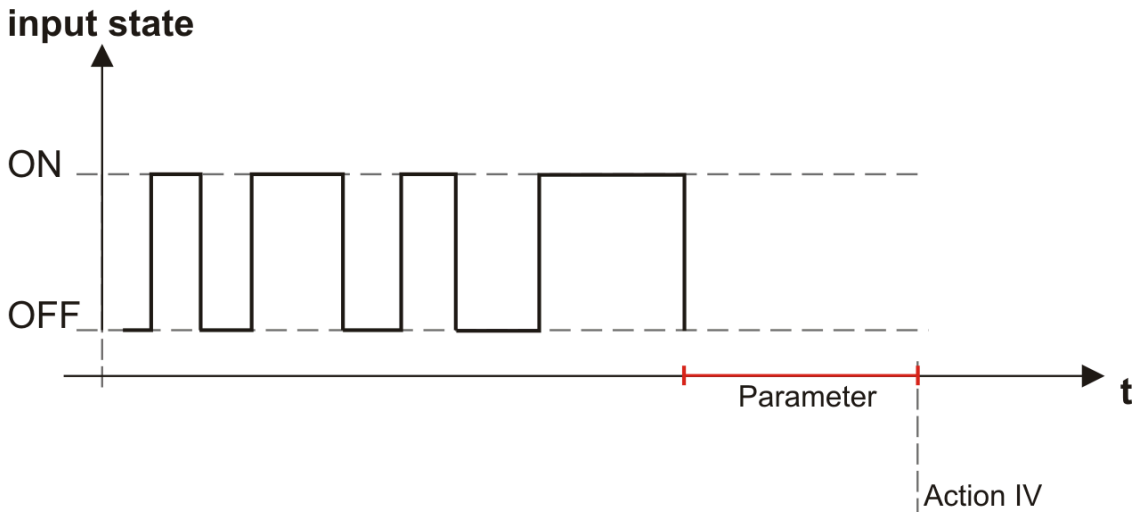
Recalling action II:



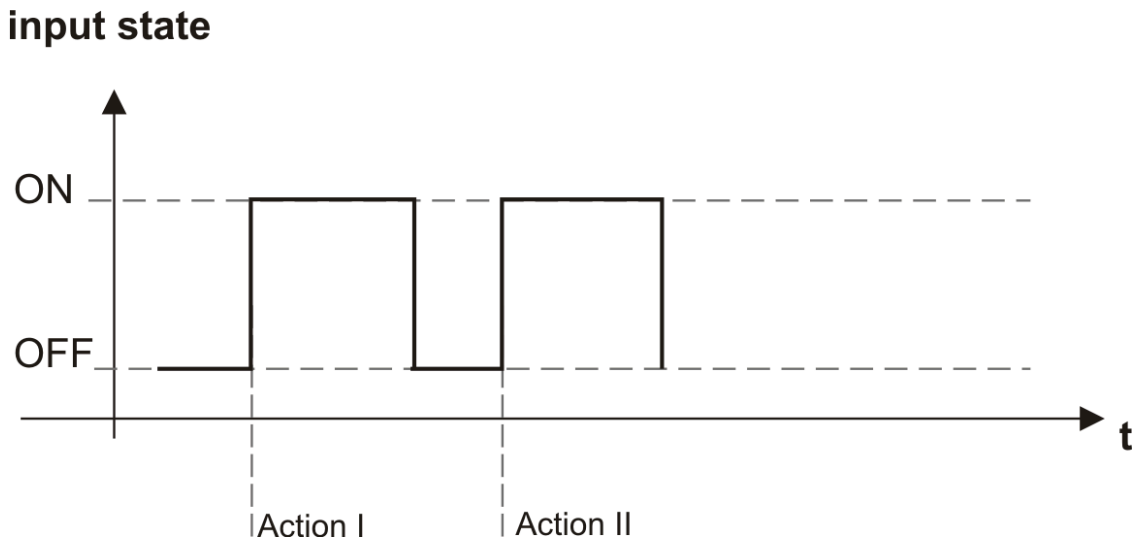
Recalling action III:



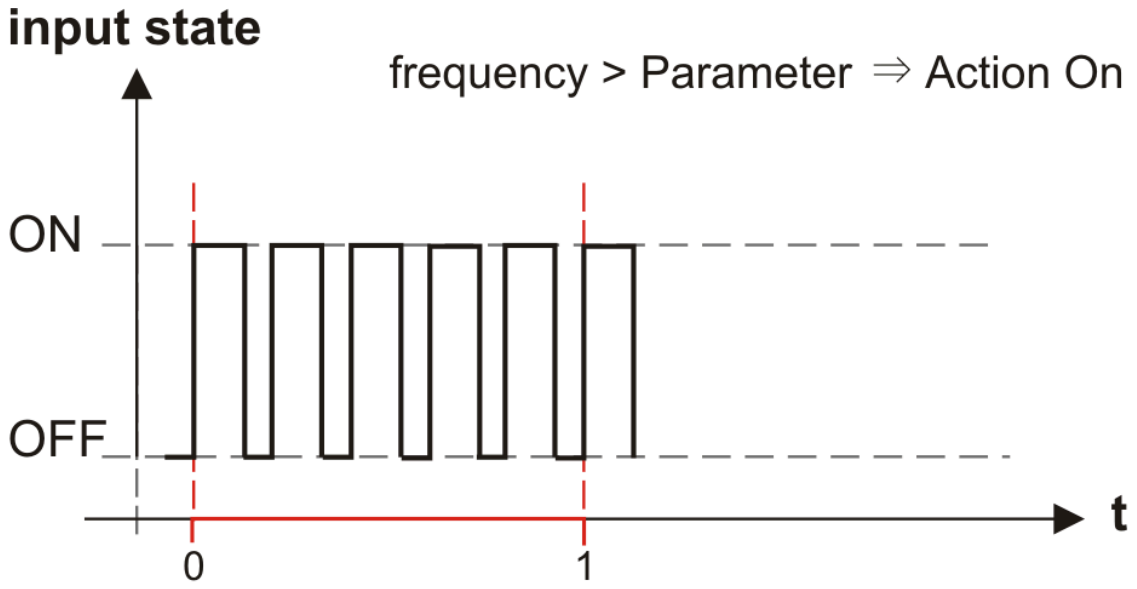
Recalling action IV:

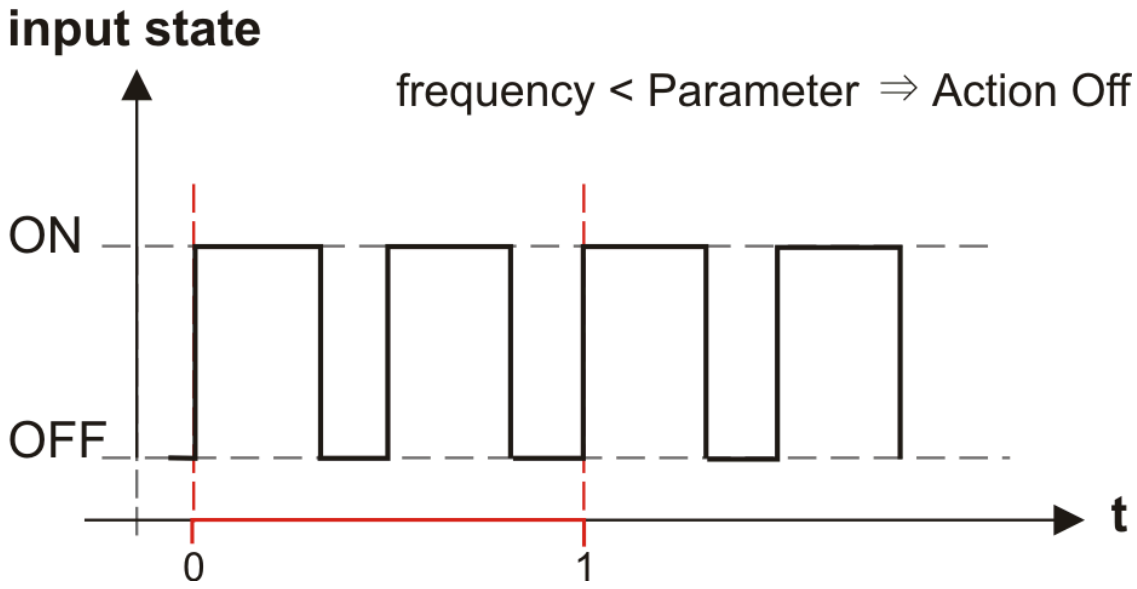


- **Toggle** – Subsequent pulses on the input recall **Action I** / **Action II** alternately,



- **Freq** – Action recall is dependent upon the frequency of pulses on the input. The desired frequency is inputted into the **Parameter** field in **Hz**,



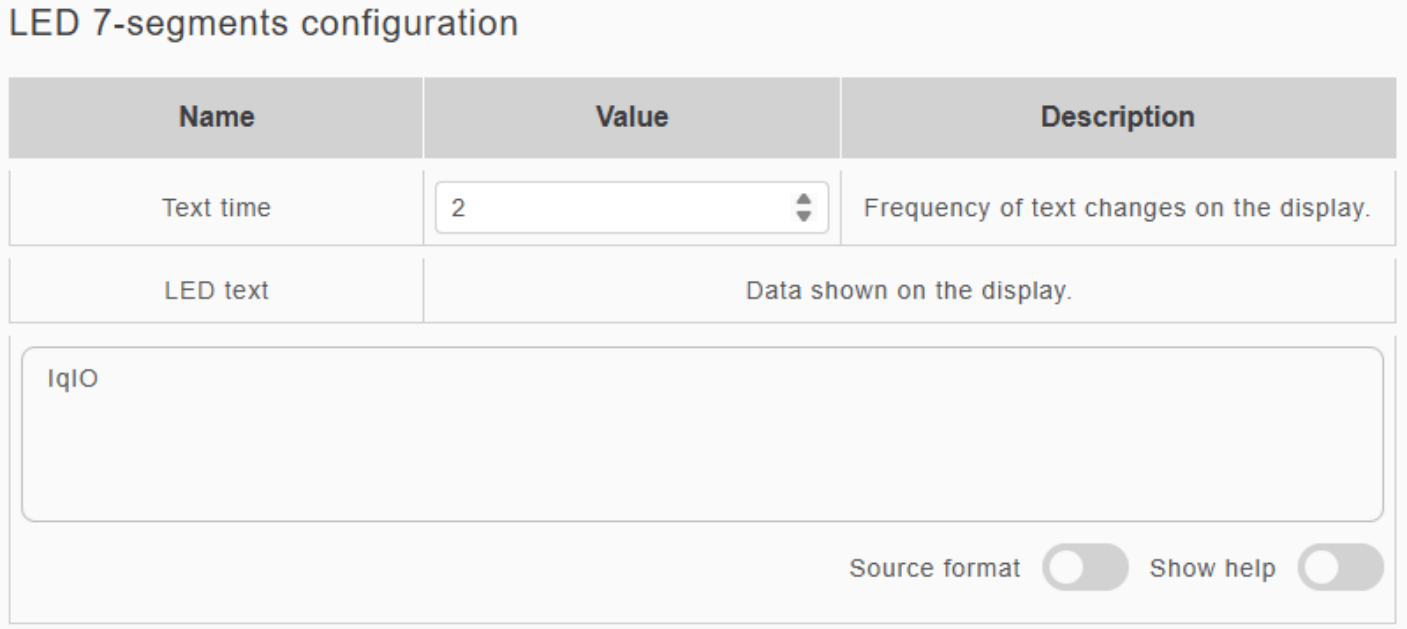


- **Parameter** – Value used in various action activation methods. When the cursor is placed over the input field, a parameter unit is displayed - eg. **Hz** for **Freq**,
- **Text ON** – Text dispayed on the LED display when the input is activated,
- **Text OFF** – Text dispayed on the LED display when the input is deactivated.

The **Go to the input actions** button located in the upper-right corner allows quick access to the **Action/Inputs** tab.

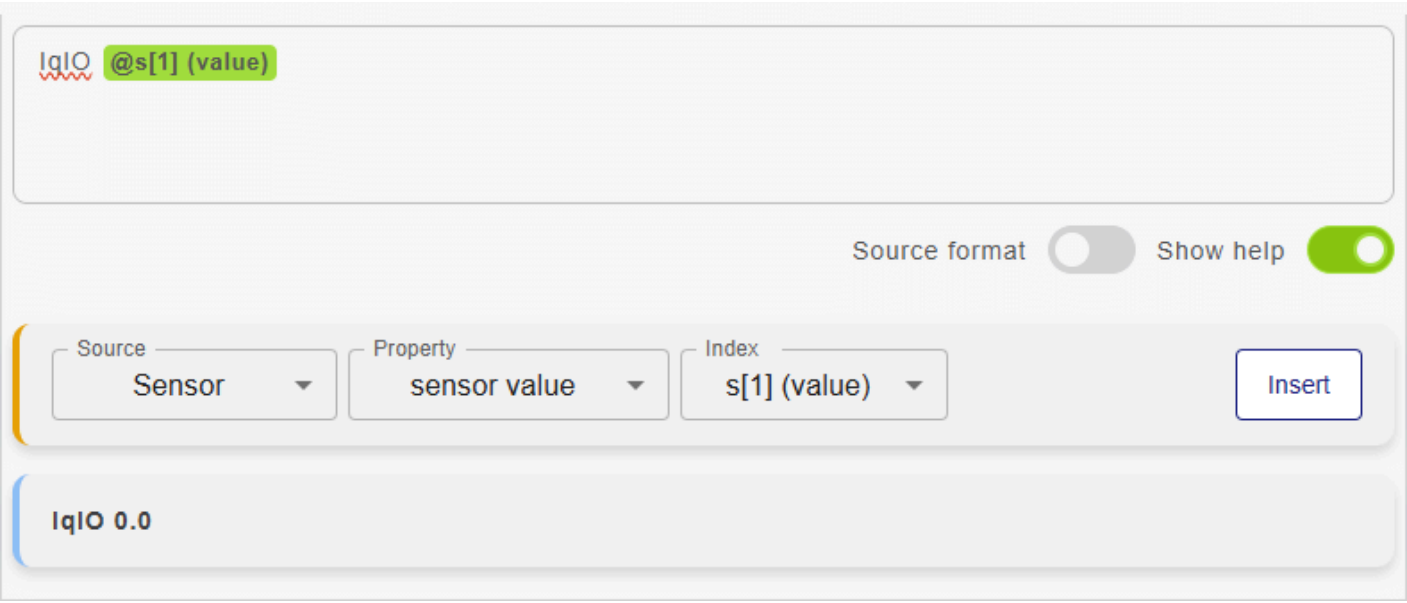
Display LED

The display settings can be configured in this tab:



- **Text time** – Frequency of display text changes – in seconds,
- **LED text** – Data displayed on the main screen. Predefined variables can be used.

Clicking **Show help** opens a tool that allows easy input of variables into the **LED text** field. Displayed text preview is available in the bottom part.



Notifications configuration

This tab enables configuration of various notifications – activation, deactivation and notifications assignment involving E-mail, SMS, SNMP Trap, MQTT, related to sensors, inputs and outputs *(version dependent)*.

For the notifications to be effective:

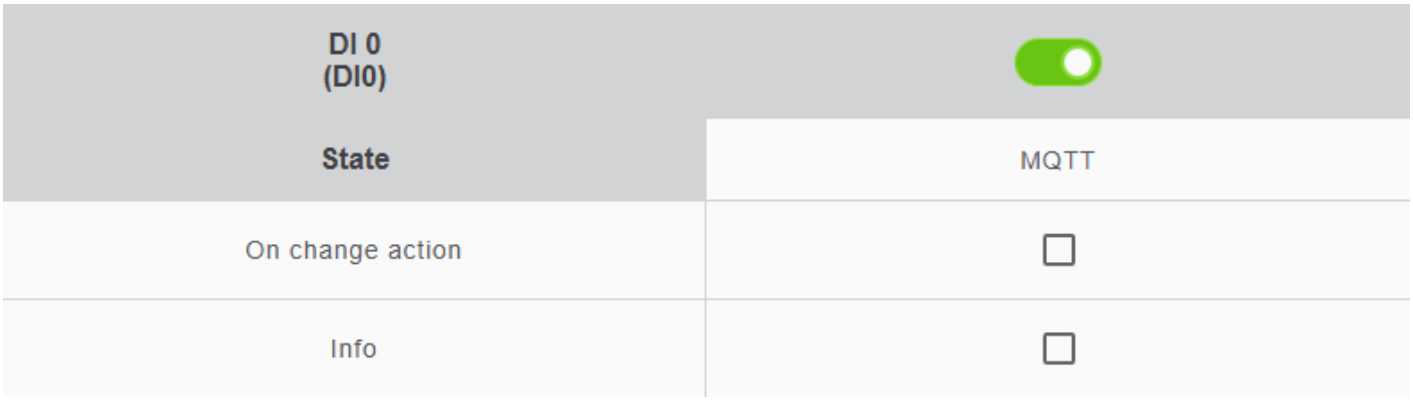
Step 1: Enable notifications in the selected system elements tab: sensors, inputs or outputs, define the type of notifications,

Step 2: Depending on the notification type selected - SMS, e-mail, SNMP Trap, MQTT – configure the settings in **Services** – consult **Network services**

Step 3: Enable notifications in the **Configuration** tab.

Inputs

In this tab, notifications settings regarding operation of device inputs can be configured. After clicking , notifications for the selected input will be activated, leaving only their details to be configured. The configuration window that appears after the notifications have been activated, enables full customization.



It's possible to activate MQTT and SMS, SNMP Trap and notifications.

> **Tip**
>
> **In order for the E-mail, SMS, SNMP Trap and MQTT notifications to operate correctly, configure them in the Services tab – consult Network services.**

In the table, the user can select the type of notifications that are to be sent in response to defined events:

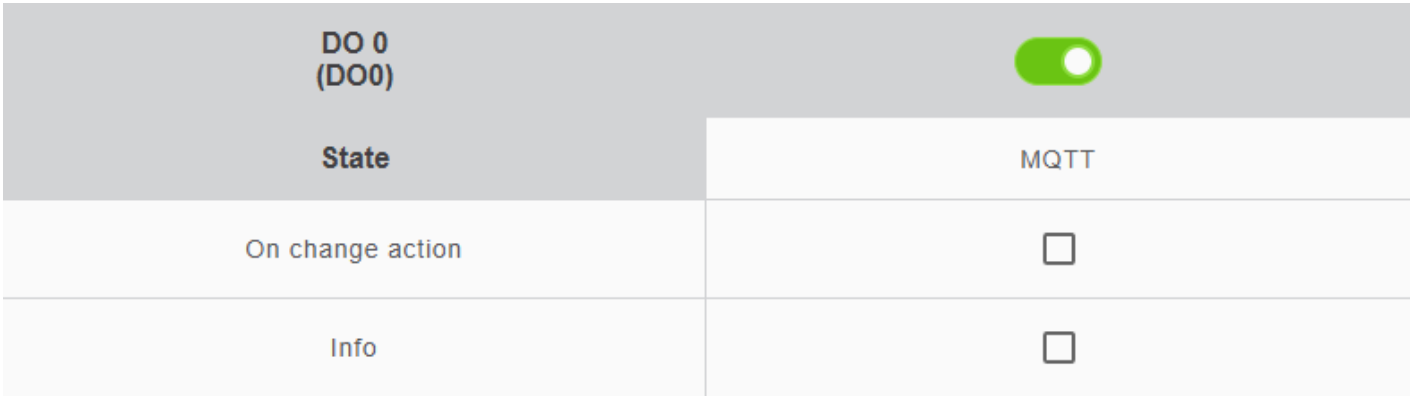
- **On change action** – Input state change (activated – deactivated),
- **Info** – Input state information.

> **Tip**
>
> **To activate the notifications function, it is important to enable it in the Configuration tab, in addition to the settings configured here.**

Outputs

In this tab, notifications settings regarding operation of devices connected to the device's outputs can be configured.

After clicking , notifications for the selected output will be activated, leaving only their details to be configured. The configuration window that appears after the notifications have been activated, enables full customization.



It's possible to activate E-mail, SMS, SNMP Trap and MQTT notifications.

> **Tip**
>
> **In order for the E-mail, SMS, SNMP Trap and MQTT notifications to operate correctly, configure them in the** Services **tab - consult** Network services**.**

In the table, the user has the ability to select the type of notifications that are to be sent in response to defined events:

- **On change action** – Output state change (activated – deactivated),
- **Info** – Output state information.

> **Tip**
>
> **To activate the notifications function, it is important to enable it in the** Configuration **tab, in addition to the settings configured here.**

Configuration

In this section, there is an option to activate the notification functions required to send notifications. Additionally, the user has the ability to adjust general parameters related to notifications.

Protocol	Value	Description
Notifcation		Enable notification
MQTT info	60	MQTT info time [s]
MQTT Retain		Set MQTT Retain flag
IO time	100	Minimum time before sending another state change for inputs and outputs [ms]

- **Notification** – Enable/disable notifications,
- **E-mail info** – Frequency of e-mail messages containing the sensor/input/output state sending *(only modules with SMTP support)*,
- **SMS info** – Frequency of SMS messages containing the sensor/input/output state sending *(only modules with SMS API support)*,
- **MQTT info** – Frequency of MQTT messages containing the sensor/input/output state sending,
- **MQTT Retain** – Activated option causes the broker to retain last messages for topics the device sends data to,
- **SNMP Trap** – Selected SNMP Trap,
- **IO time** – Minimum time that must pass between subsequent input/output state changes to prevent multiple notifications to be sent, especially during testing or experiments with device's inputs/outputs.

The tables **Sensor**, **Input** and **Output** are located in this tab.
Each table contains predefined commands that allow sending of e-mail and SMS messages containing the current device states. Additionally, the user has the ability to edit those commands, which makes customization to own preferences possible, eg. by assigning a name to the device. Each table contains a topic, use of which is required if using MQTT to send notifications.

Input	
MQTT	MQTT Topic iqio/c09bf4a00464 Input notification MQTT Topic

Output	
MQTT	MQTT Topic iqio/c09bf4a00464 Output notification MQTT Topic

Binding

This tab enables configuration of relations between inputs and outputs.

Outputs

This tab allows routing of state from other components:



- **Name**,
- **Route from** – Source that will have influence over the output state – change of state of the channel in this field will cause the output state to change accordingly.

Available output state sources:

- `i[x]` – Input channel state,
- `i[x].act` – Last input action,
- `o[x]` – Output channel coil state,
- `o[x].on` – Output channel state,
- `v[x]` – Virtual variable state,
- `s[x].aHi` – Sensor high level alarm,
- `s[x].aLo` – Sensor low level alarm,
- `s[x].wHi` – Sensor high level warning,
- `s[x].wLo` – Sensor low level warning,
- `s[x].err` – Sensor error,
- `s[x].ok` – Sensor operates correctly,
- `ping[x]` – Ping status: `0` - error, `1` - success,
- `mb[x].y` – mbAgent value,
- External device:
  - `dev[x].err` – Connection lost,
  - `dev[x].o[y]` – External output,
  - `dev[x].i[y]` – External input,
  - `dev[x].s[y].err` – External sensor error.
- Negation and logic operations (|/&/^) e.g.:
  - `!s[0].err`
  - `!o[0]&v[1]`
- **Route to** – Variable, the output state will have influence on. Available variables:
- `v[x]` – Internal variable,
- `mb[x].y` – mbAgent value,
- External device:

- `dev[x].o[y]` – External output.

- Negation and logic operations (|/&/^) e.g.:
  - `!s[0].err`
  - `!o[0]&v[1]`

> **Tip**
>
> **If the source is preceeded by "!i", the coil state will be opposite to the state of source channel. For example: !io[3] means that the output will have the state opposite to input 3.**

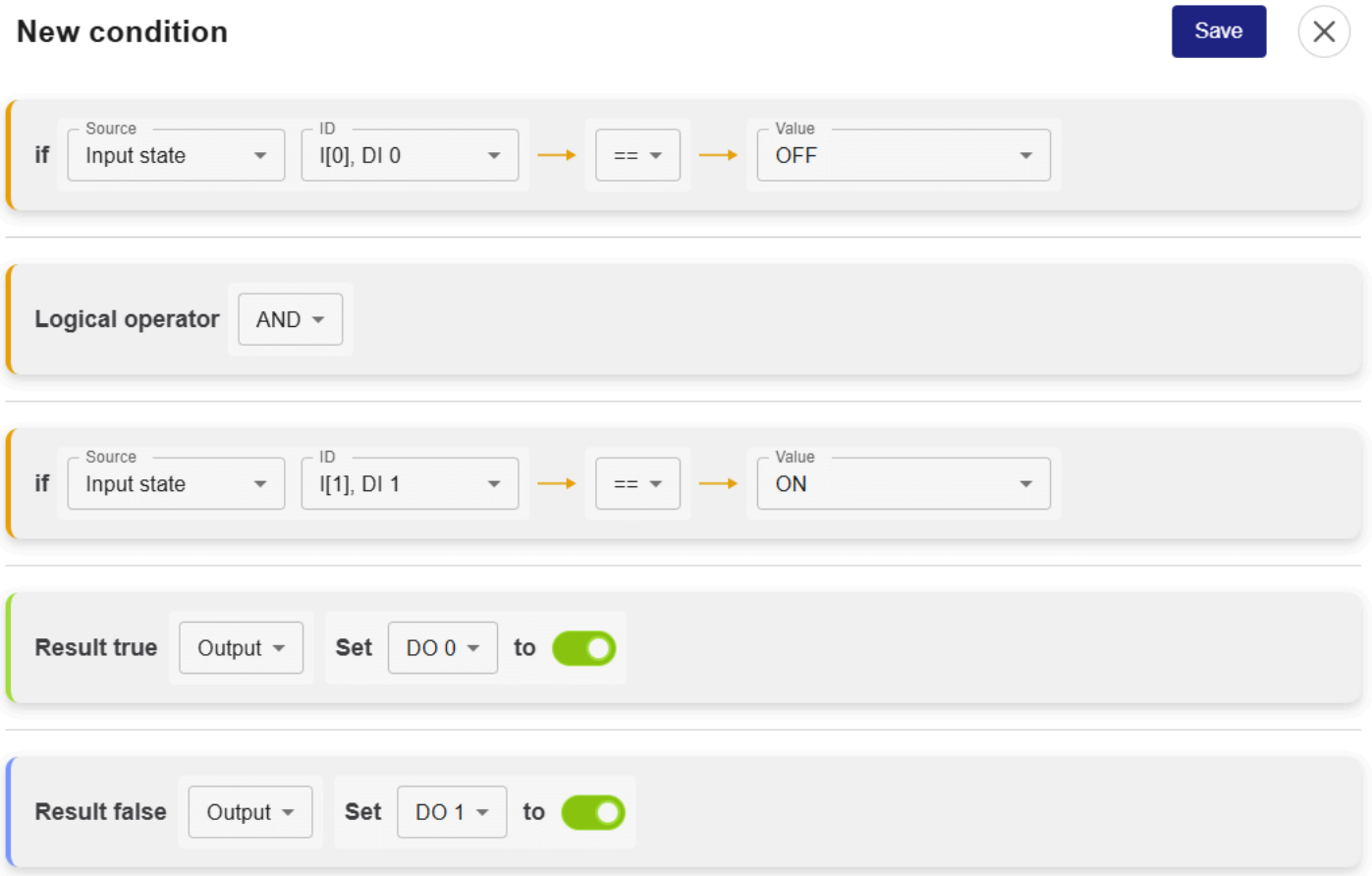
The  icon opens the step-by-step configuration window.

Logic functions

This tab allows configuration of logic functions – afer certain conditions are met, a defined action is executed.

Conditions

To configure a new logical funtion, click **Add a condition**.



Available logical variables *(availability depends on the device version)*:

- **input** – Physical or virtual input state,
- **input cnt** – Input counter,

- **output** – Physical or virtual output state,
- **sensor** – Sensor reading,
- **sensor state**:
  - **0** – Sensor error,
  - **1** – Normal state,
  - **2** – Sensor low level warning,
  - **3** – Sensor high level warning,
  - **4** – Sensor low level alarm,
  - **5** – Sensor high level alarm.
- **constant**,
- **variable**,
- **active input action** – In **cnt** input mode,
- **schedule**.

Available logic operators:

- **AND** – Conjunction,
- **OR** – Disjunction (logical sum),
- **XOR** – Exclusive disjunction,
- **NAND** – Negation of conjunction,
- **NOR** – Negation of disjunction,
- **XNOR** – Negation of exclusive disjunction (logical equivalence).

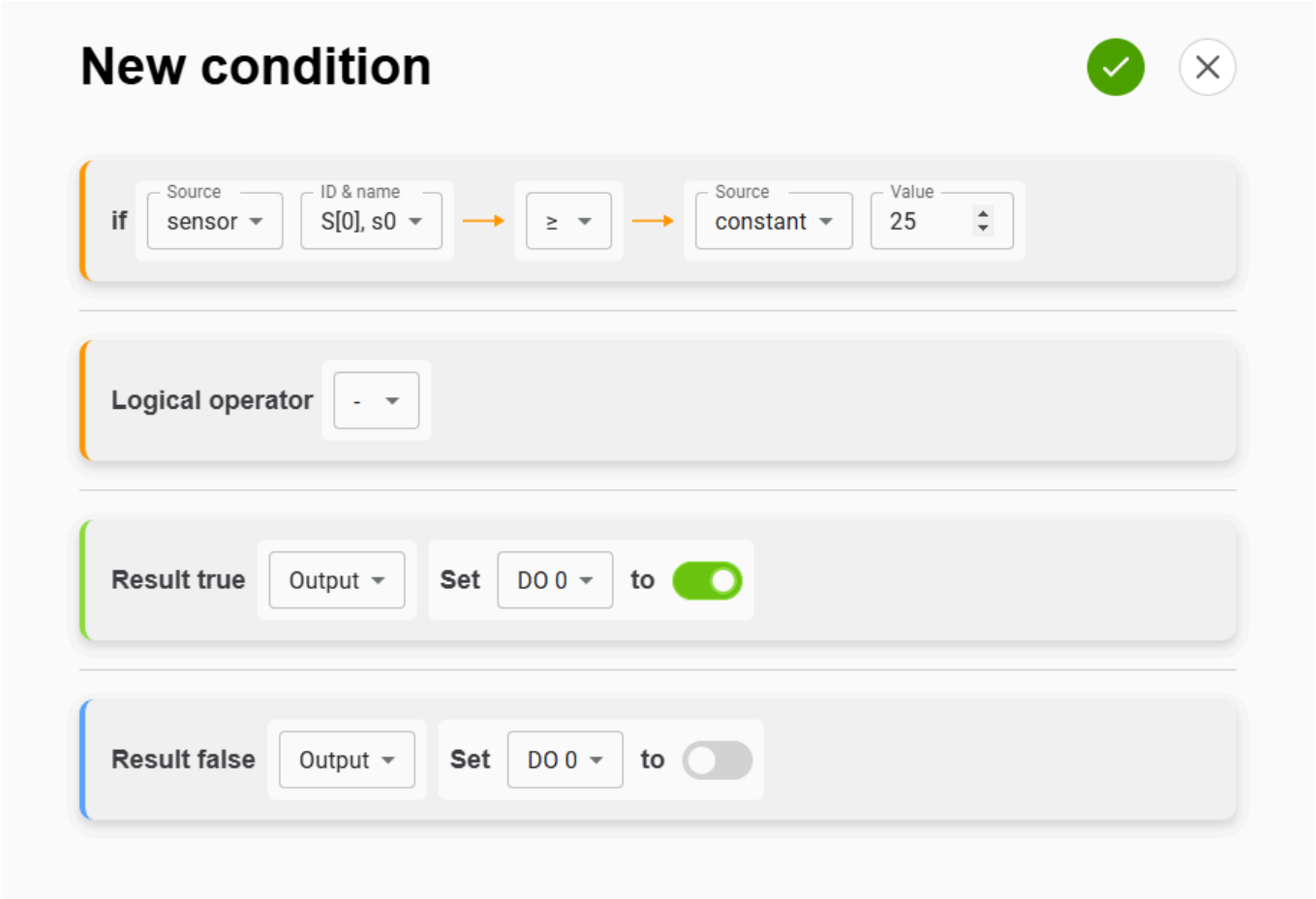
Result True / Result False – Reaction for meeting/not meeting the condition:

- **None**,
- **Output** – Output control,
- **Action** – Recall defined action,
- **Variable** – Virtual input/output control.

Tip

All changes should be saved with 

Example



Output 1 will be activated as long as the sensor 0 temperature is greater than or equal to 25. If the temperature drops below this value, output 1 will be deactivated.

Variables

The **Variables** tab allows the user to define the values of internal variables.

Manual modification and preview of virtual variable values

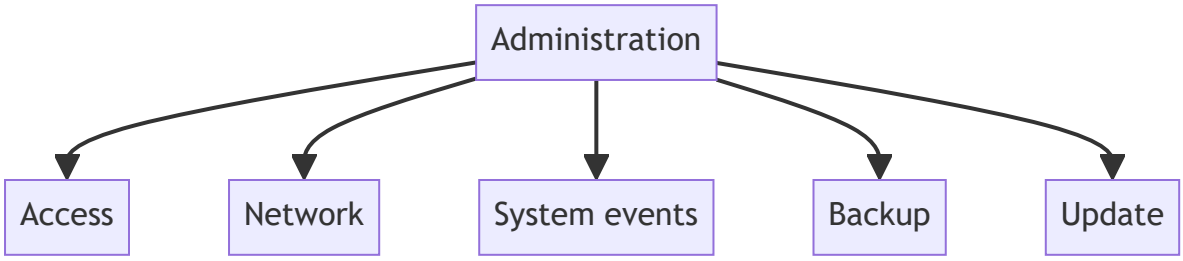
Set V[0] as Value 0

Virtual variables

Column 1	Column 2	Column 3	Column 4
V[0] = 0	V[1] = 0	V[2] = 0	V[3] = 0
V[4] = 0	V[5] = 0	V[6] = 0	V[7] = 0
V[8] = 0	V[9] = 0	V[10] = 0	V[11] = 0
V[12] = 0	V[13] = 0	V[14] = 0	V[15] = 0
V[16] = 0	V[17] = 0	V[18] = 0	V[19] = 0
V[20] = 0	V[21] = 0	V[22] = 0	V[23] = 0
V[24] = 0	V[25] = 0	V[26] = 0	V[27] = 0
V[28] = 0	V[29] = 0	V[30] = 0	V[31] = 0

System management

The **Administration** tab allows control over device aspects that influence the operation, security and system configuration.



Access

The user can manage webserver access. This involves authentication, device name and access from the Discoverer programme.

Access configuration

Name	Value	Description
Password		Enable password
Current password		
New password		
Repeat new password		
Module name	[your_module_name]	Your custom module name
Enable remote config		Allow change configuration by Discoverer app

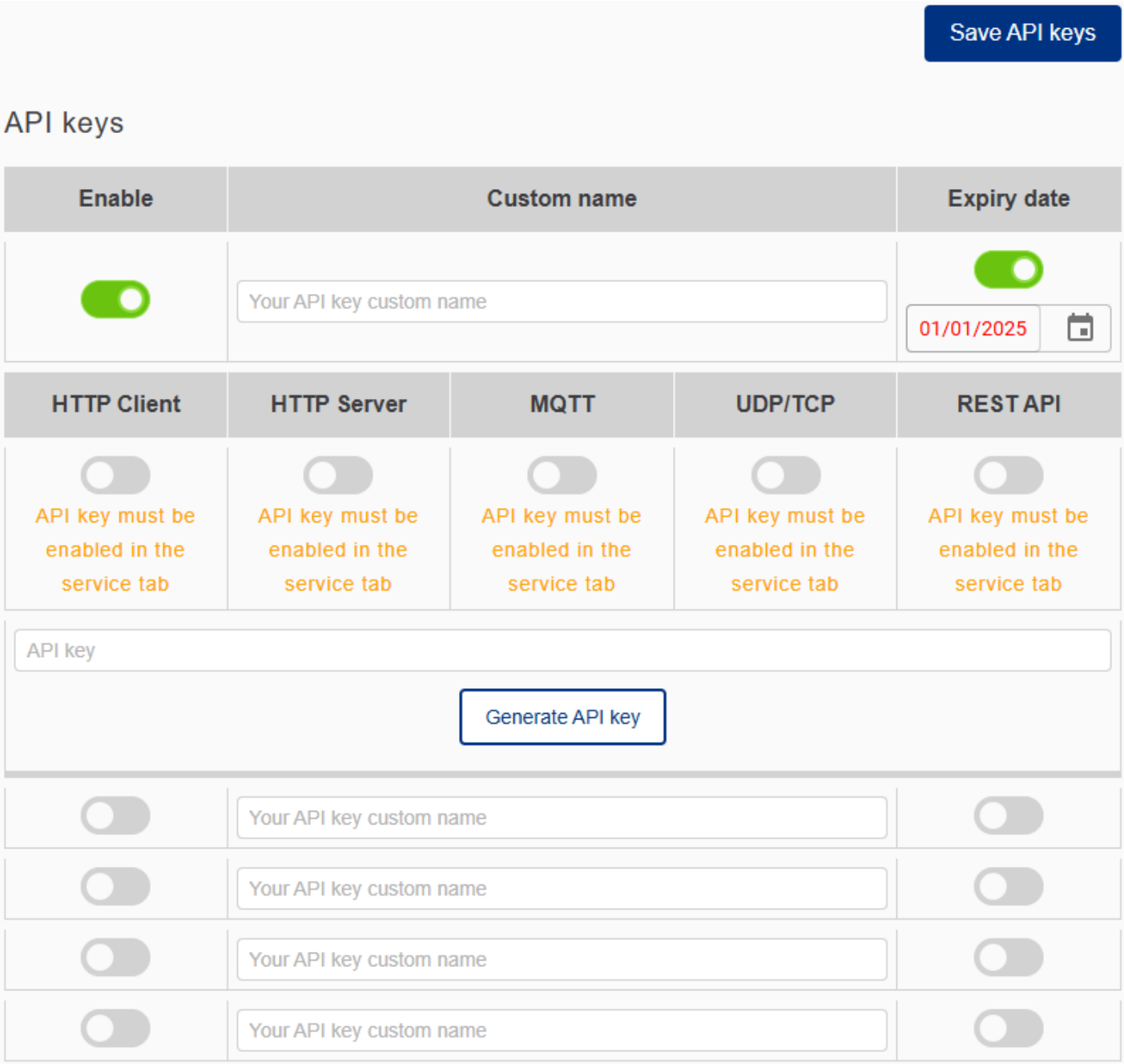
- **Password** – Enable/disable configuration password,
- **Current password**,
- **New password**,
- **Repeat new password**,
- **Module name** – Assigning an individual name to the module makes module identification in the system easier,
- **Enable remote config** – Enable/disable configuration via the Discoverer programme.

Information

Default device settings:
- username: admin
- password: admin

API KEYS

The device allows the user to configure a set of five unique API keys, used for authentication and identification of users or processes that initiate communications with the device.



- **Enable** – Enable/disable the selected API key,
- **Custom name** – Key name assigned by the user,
- **Expiry date** – API key expiry date,

- **HTTP Client, HTTP Server, MQTT, UDP/TCP, REST API** – Selection of the communication protocol that will use the specified API key,
- **API key** – Key value entry field,
- **Generate API key** – This button generates a random API key that will automatically be inputted into the **API key** field.

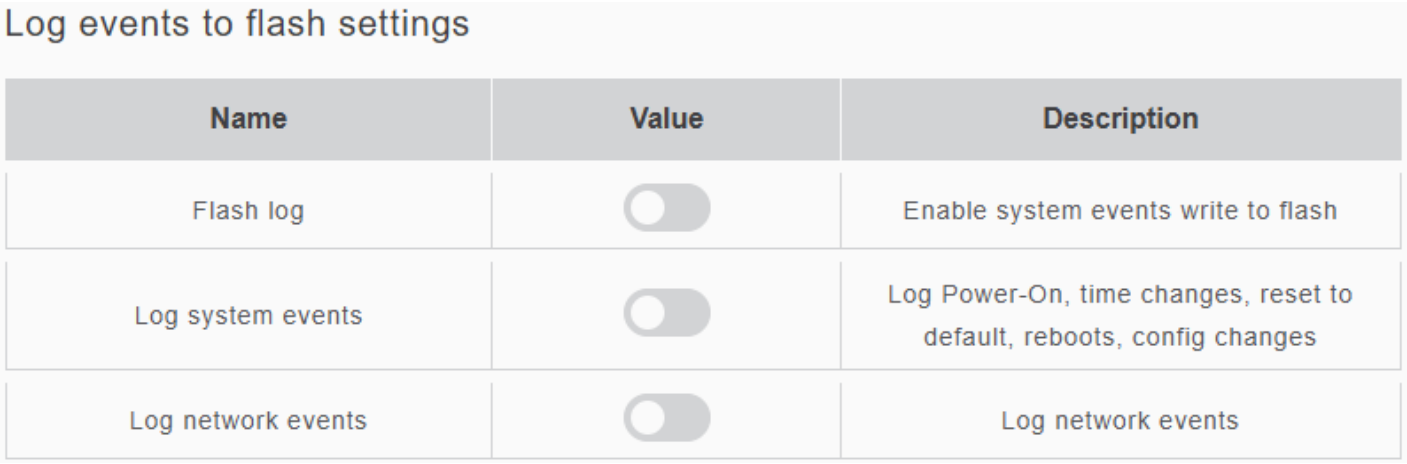
> **Tip**
>
> **Changes made in the API keys tab should be saved with Save API keys.**

Network

In this tab, the device's network settings can be changed. Configuration is described in the **Network configuration** chapter.

System events

This tab enables logging of system events to the flash memory, giving users the option to preview and analyze different system events. This process helps with system operation monitoring and diagnosing possible problems.

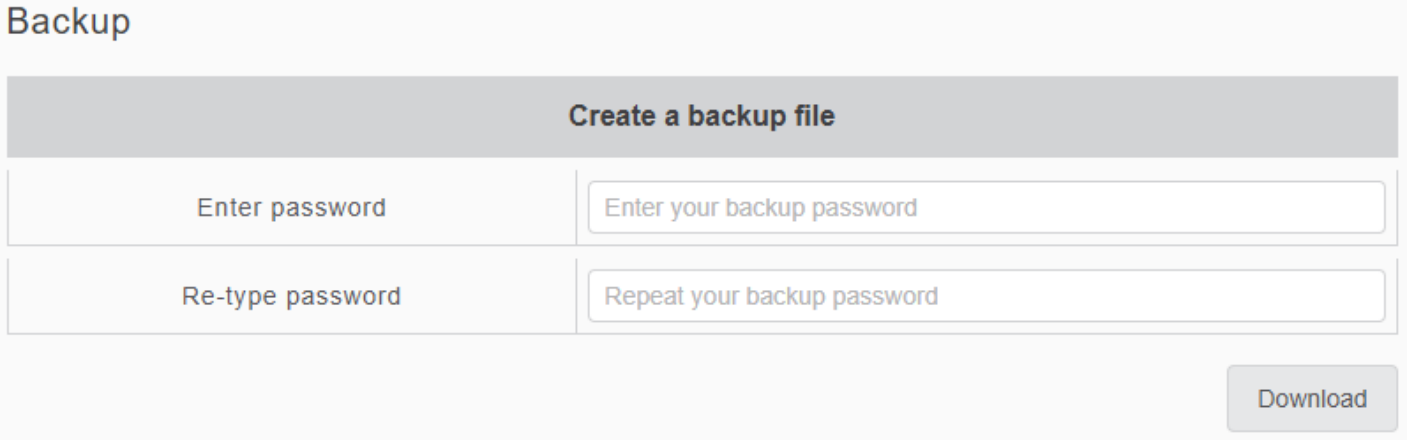


- **Flash log** – Enable/disable logging of events to flash memory,
- **Log system events** – Enable/disable logging of activations, time changes, factory settings restore, restarts, changes to configuration,
- **Log network events** – Enable/disable logging network events.

Backup

Users can create configuration backups and restore the system from previous backups in this tab.

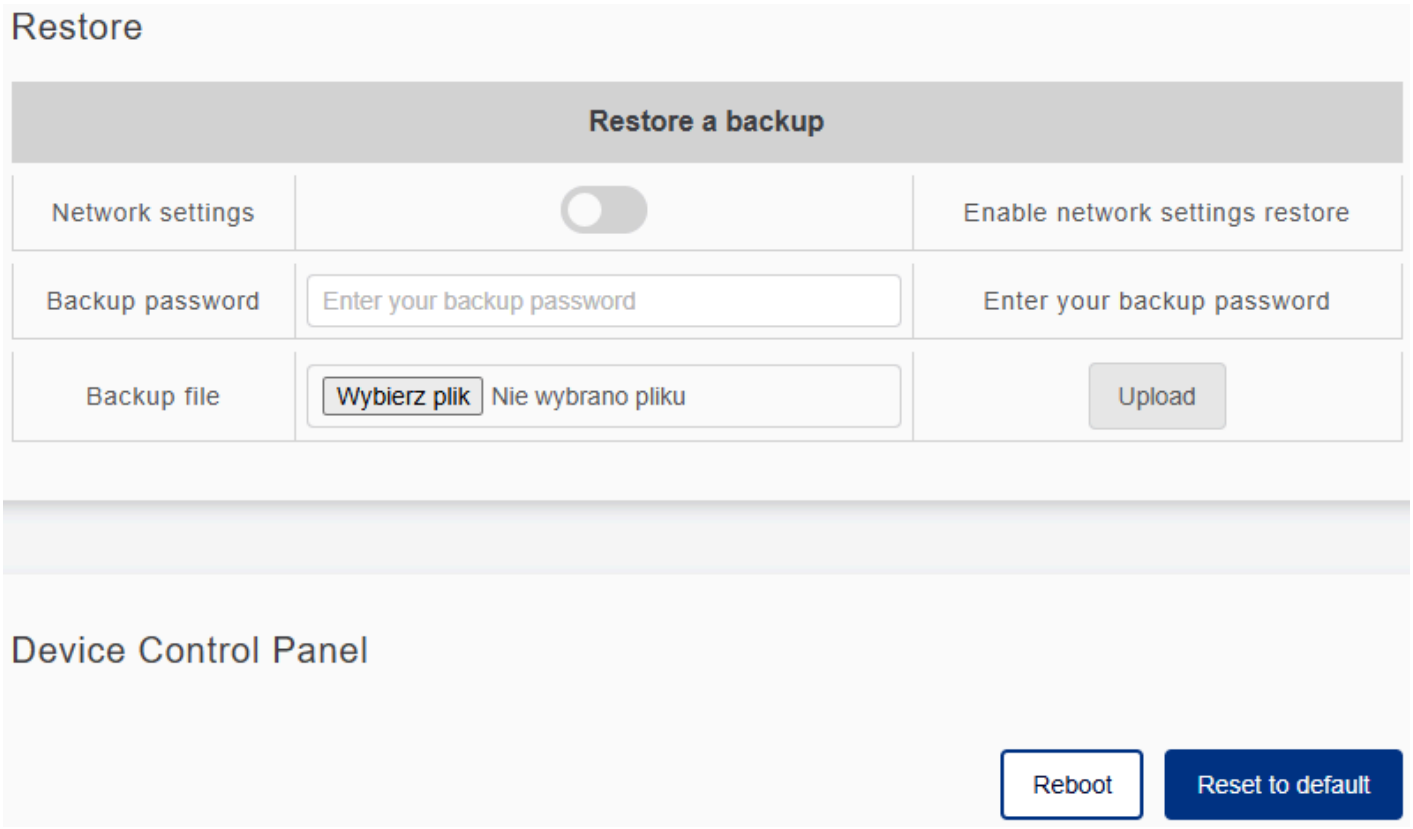
Create a backup file



- **Enter password** – Password protecting the backup,
- **Re-type password**.

Download enables download of configuration data to the PC.

Restore



- **Backup password** – Backup access password,
- **Backup file** – Browse the backup file.

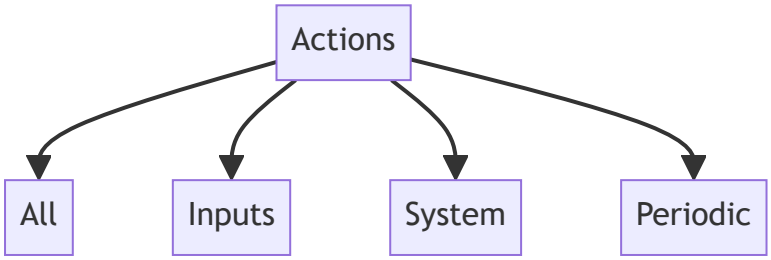
Upload – Uploads the selected backup to the device. **Reboot** – Allows the device to reboot. **Reset to default** – Allows the user to restore default settings.

Defining actions

The device undertakes user-defined actions in response to specific signals or sensor readings. Those may include:

- **Output control:** Activation or deactivation of a selected output based on sensor readings. (eg. turning a fan on after temperature exceeds a certain threshold),
- **Sending notifications in the form of SMS, e-mail, MQTT frame, HTTP, TCP, UDP, SNMP trap, and other:** Automatic sending of alert/notification to the user or another system in response to defined conditions,
- **Other actions defined by the user:** Actions specific to a particular system or needs such as saving data to a database, alarm activation, change to other device settings etc.

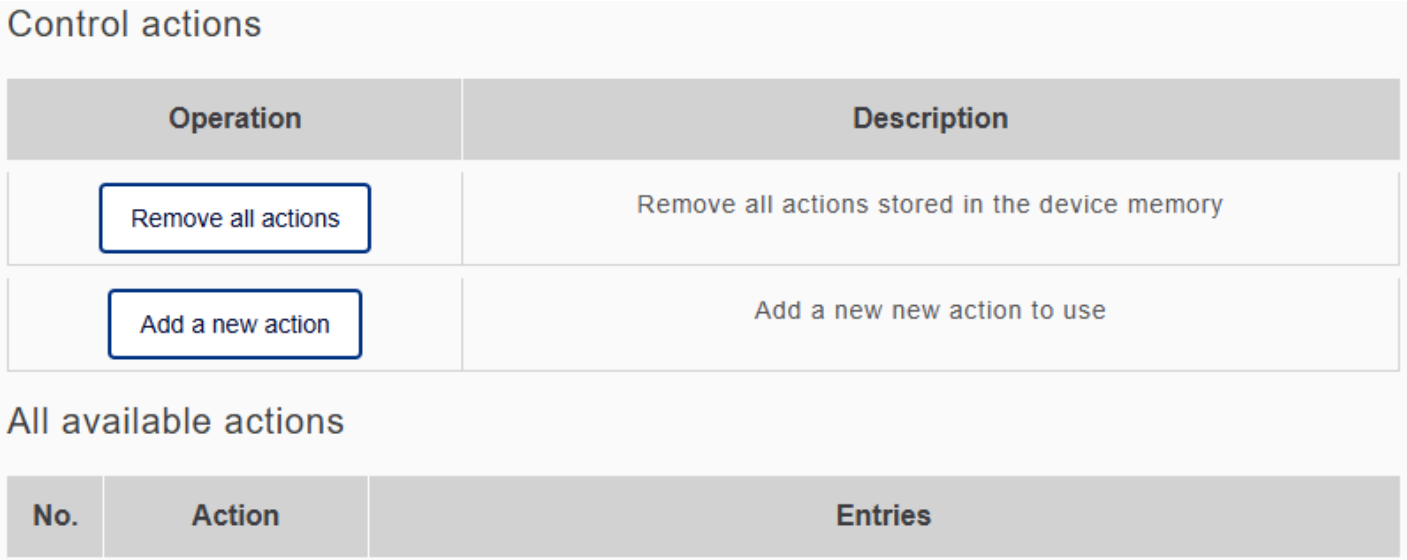
The actions are specific reactions of the module to received signals and input data, in accordance to the instructions set by the user. Many functions can be conducted by multiple methods depending on preferences and needs.



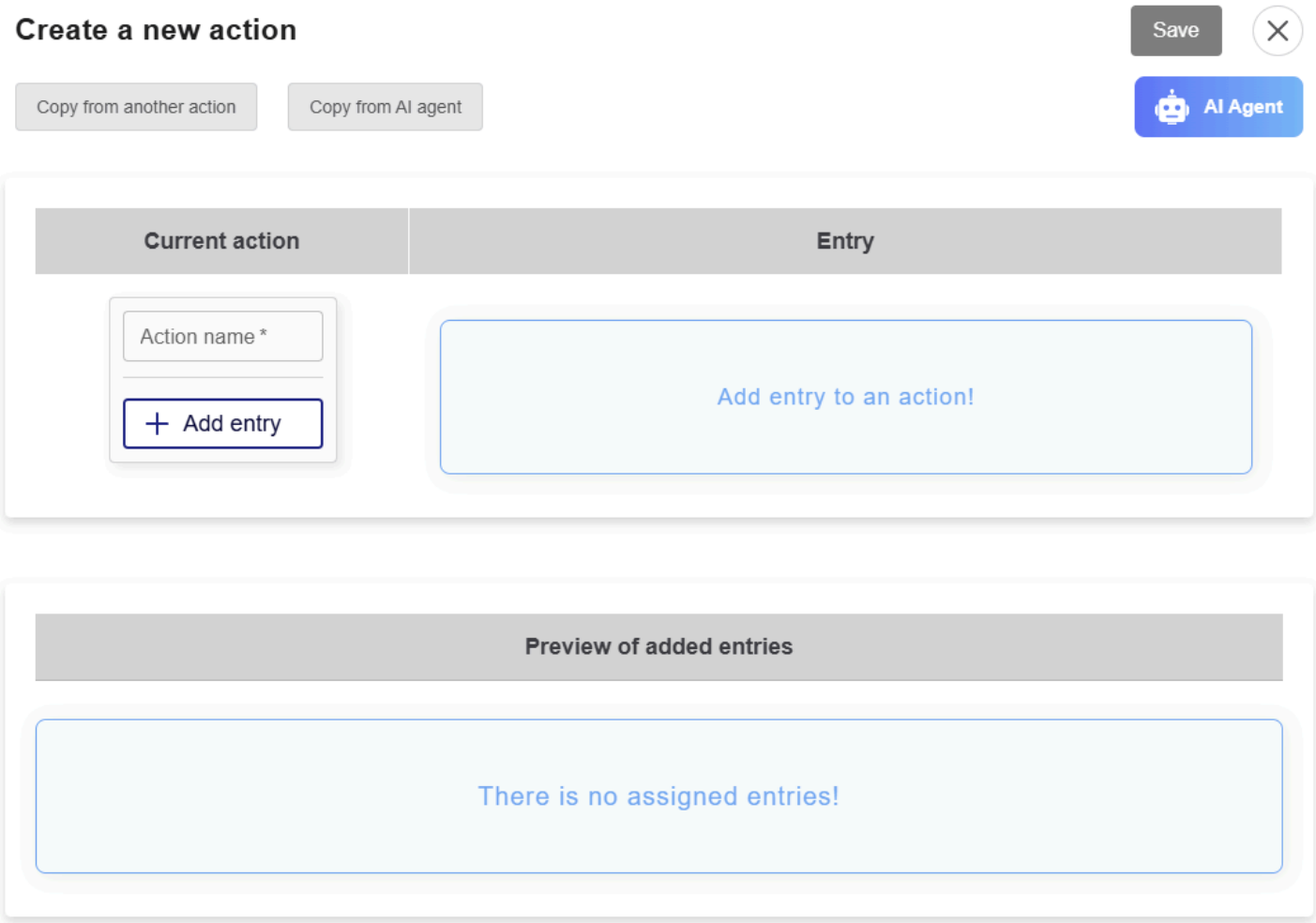
All

This tab allows preview and management of defined actions supported by the device.

Control Actions window



- **Remove all actions** – Removes all actions defined in the device,
- **Add a new action** – A window displayed after clicking this button allows the user to define each parameter of the added action:



- **Copy from another action**,
- **Copy from AI agent** – Opens a new window, to which the Inveo AI - generated string can be entered,
- **Action name**.

Clicking **Add entry** allows for communication protocol selection and further configuration.

After the programmed action's details are configured, click **Add**. It's possible to configure multiple actions for one event. After defining all required settings, save the changes using .

All available actions window

All defined and system actions are visible in this window. Each of them can be:

- Amended –
- Tested – Try
- Deleted –

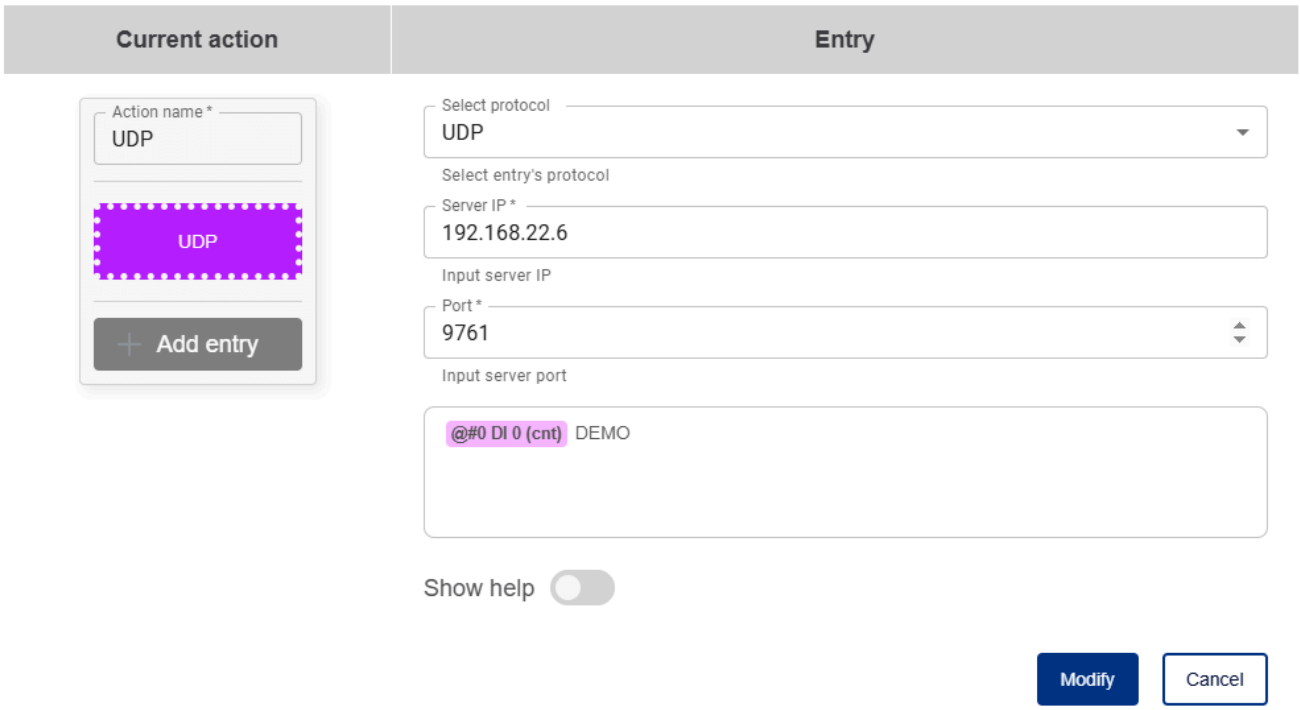
Protocol configuration

UDP

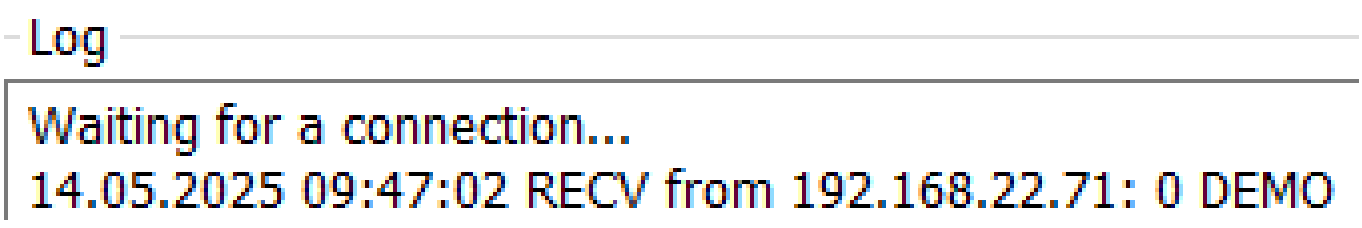
- **Server IP** – Target IP address,
- **Port** – Port, on which the target device is listening,
- **Format** – Data format: **TEXT** or **HEX**,
- **Input data** – Command sent to the target device.

Example

- **Server IP: 192.168.22.6**
- **Port: 9761**
- **Format: TEXT**
- **Input data:** `%i[0].cnt% DEMO`



Sent data:

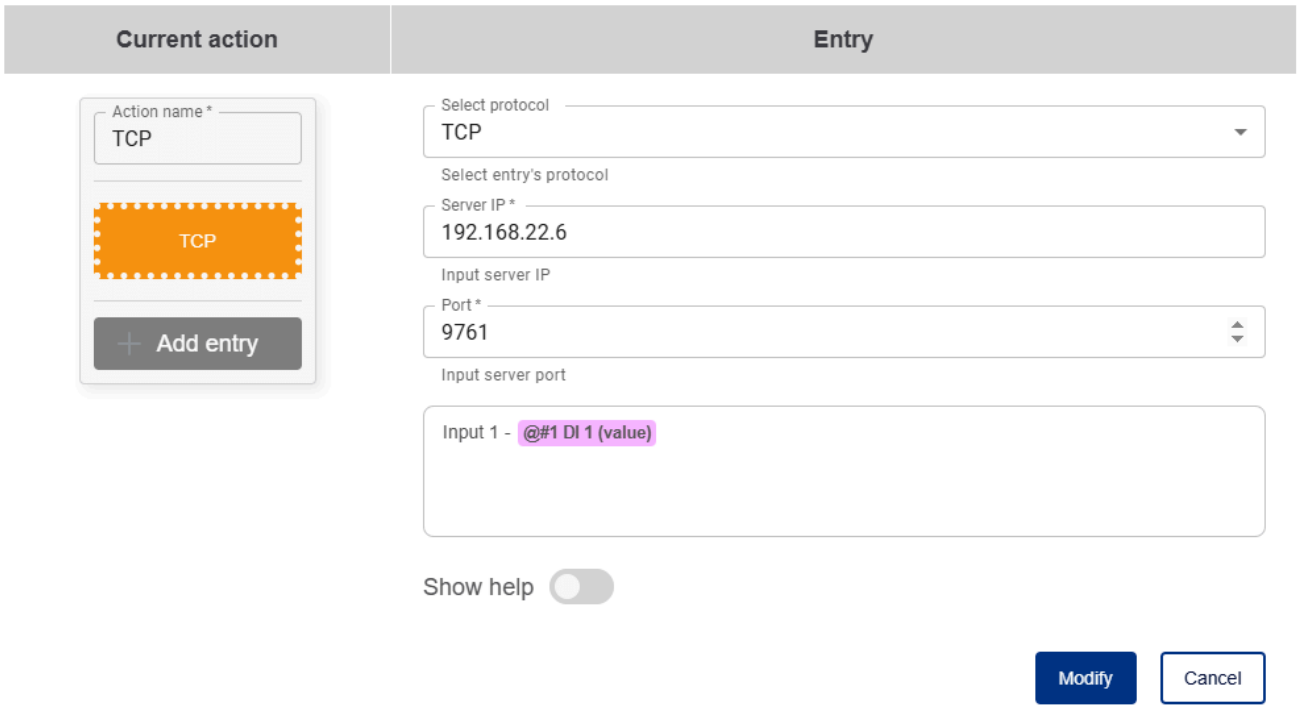


TCP

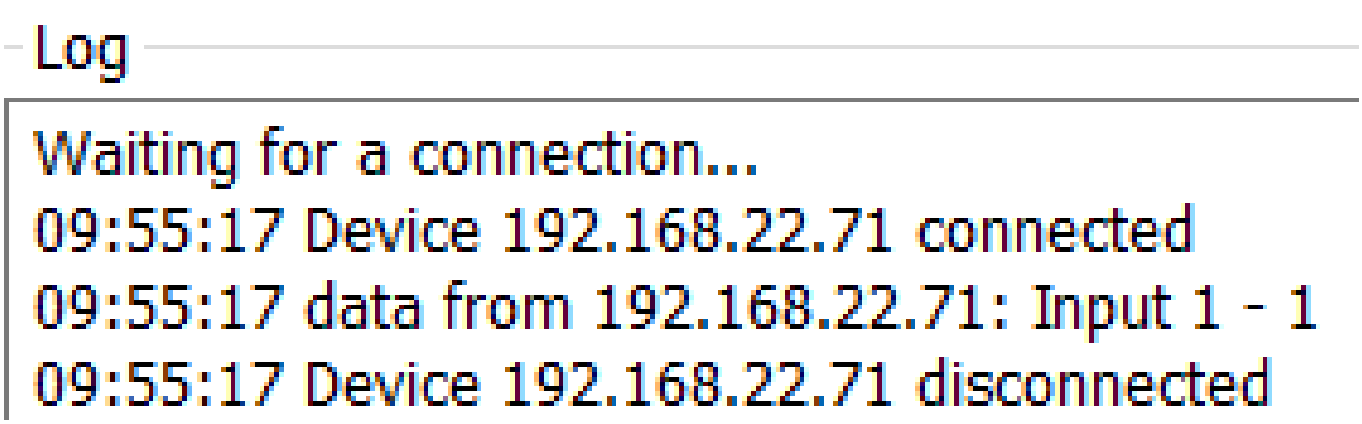
- **Server IP** – Target IP address,
- **Port** – Port, on which the target device is listening,
- **Format** – Data format: **TEXT** or **HEX**,
- **Input data** – Command sent to the target device.

Example

- **Server IP: 192.168.22.6**
- **Port: 9761**
- **Format: TEXT**
- **Input data:** `Input 1 - %i[1]%`



Sent data:



HTTP

Detailed protocol configuration is located in the **Services** tab.

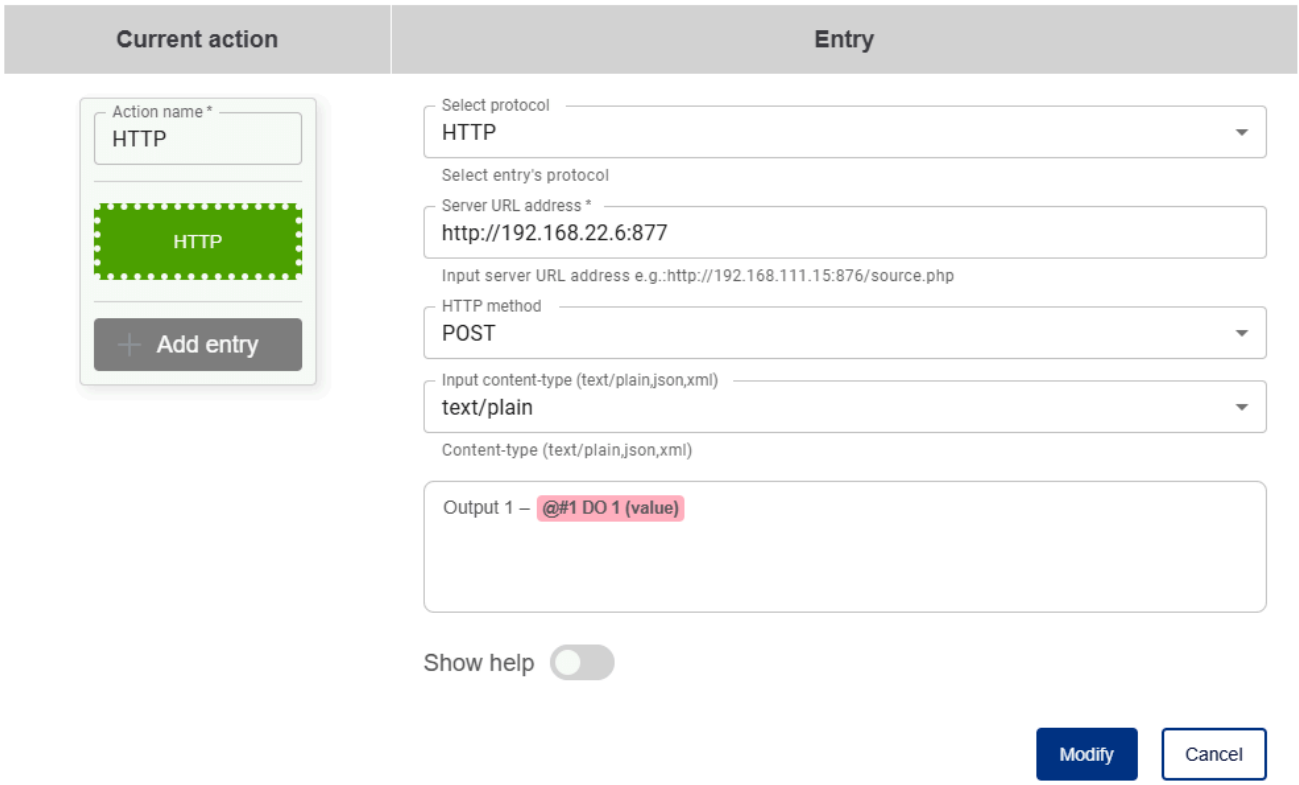
- **Server URL address** – Target URL address,

> **Warning**
>
> **The target URL must match the following convention:** `http://<IP_address>:<port>/<resource>`

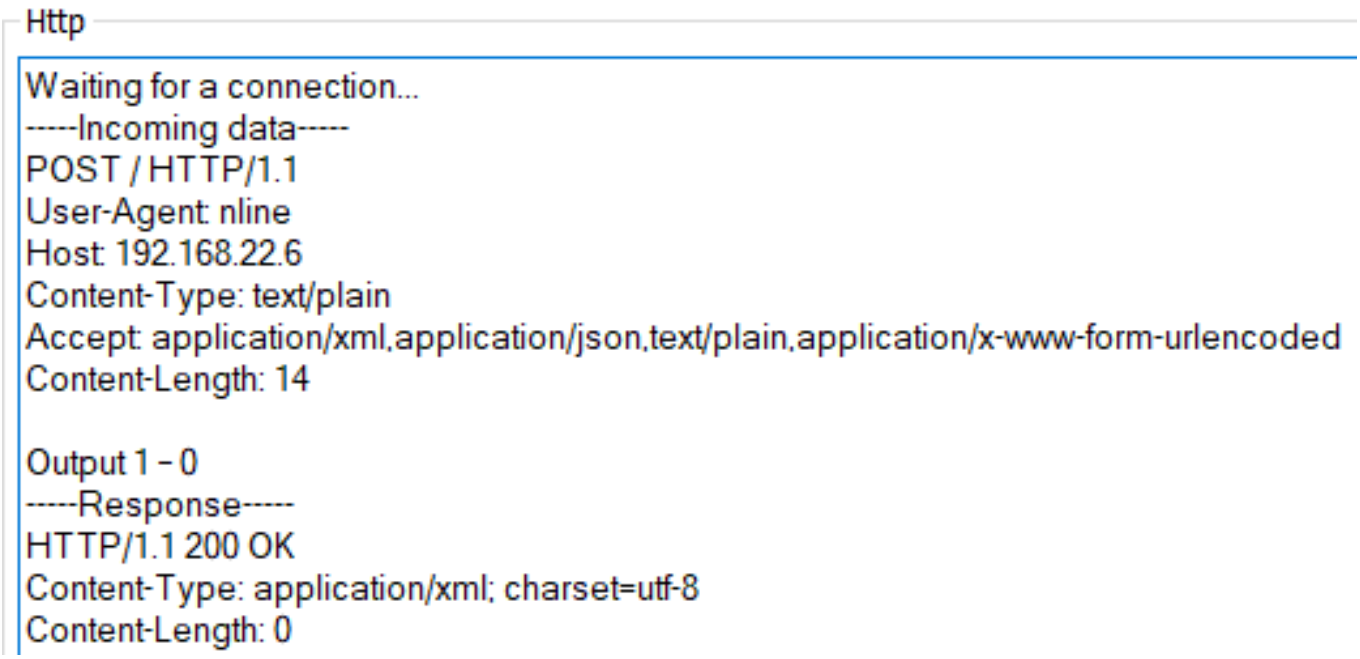
- **HTTP method:**
  - GET
  - POST
  - PUT
  - DELETE
- **Content-type:**
  - text/plain
  - application/x-www-form-urlencoded
  - application/json
  - application/xml
- **Input data** – Command sent to the target device.

Example 1

- **Server URL address:** `http://192.168.22.6:877`
- **HTTP method: POST**
- **Content-type: text/plain**
- **Input data:** `Output 1 – %o[1]%`



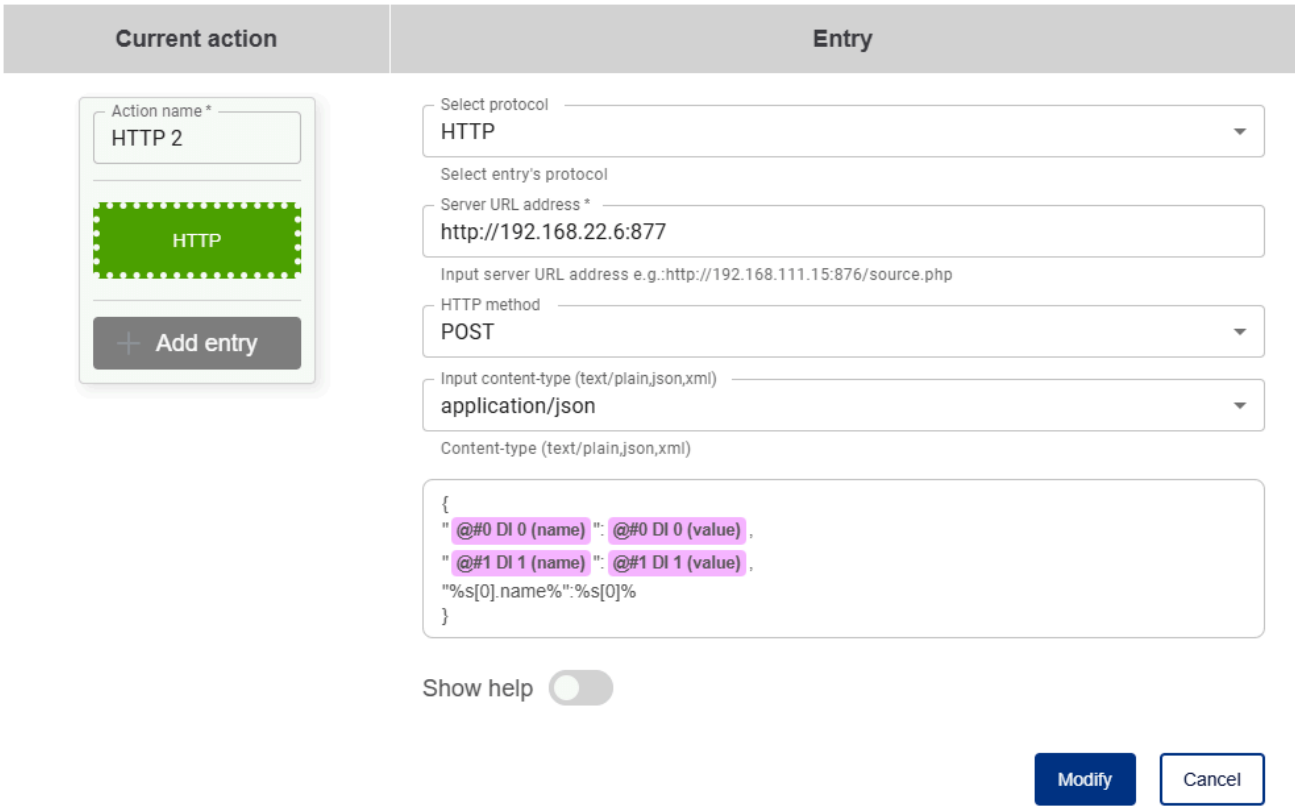
Sent data:



Example 2

- **Server URL address:** `http://192.168.22.6:877`
- **HTTP method: POST**
- **Content-type: application/json**
- **Input data:**

```
{
"%i[0].name%":%i[0]%,
"%i[1].name%":%i[1]%,
"%s[0].name%":%s[0]%
}
```



Sent data:

Http

```
Waiting for a connection...
-----Incoming data-----
POST / HTTP/1.1
User-Agent: nline
Host: 192.168.22.6
Content-Type: application/json
Accept: application/xml,application/json,text/plain,application/x-www-form-urlencoded
Content-Length: 30

{"DI 0":0,"DI 1":1,"s0":0}
-----Response-----
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
Content-Length: 0
```

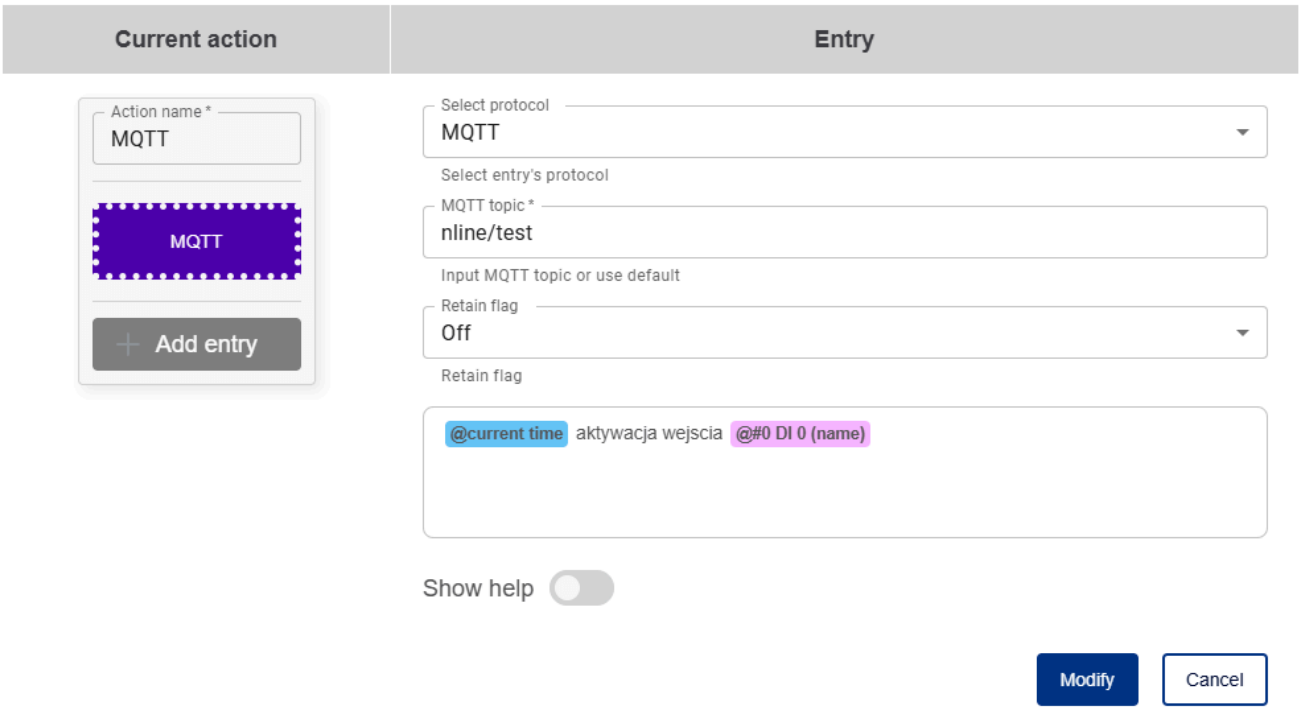
MQTT

Detailed protocol configuration is located in the **Services** tab.

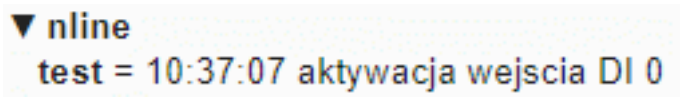
- **MQTT topic** – Topic, the device sends data to,
- **Retain flag** – Activated option causes the broker to retain last messages for topics the device sends data to,
- **Input data** – Message payload.

Example

- **MQTT topic:** `nline/test`
- **Retain flag: Off**
- **Input data:** `%time% input %i[0].name% activation`



Sent data:

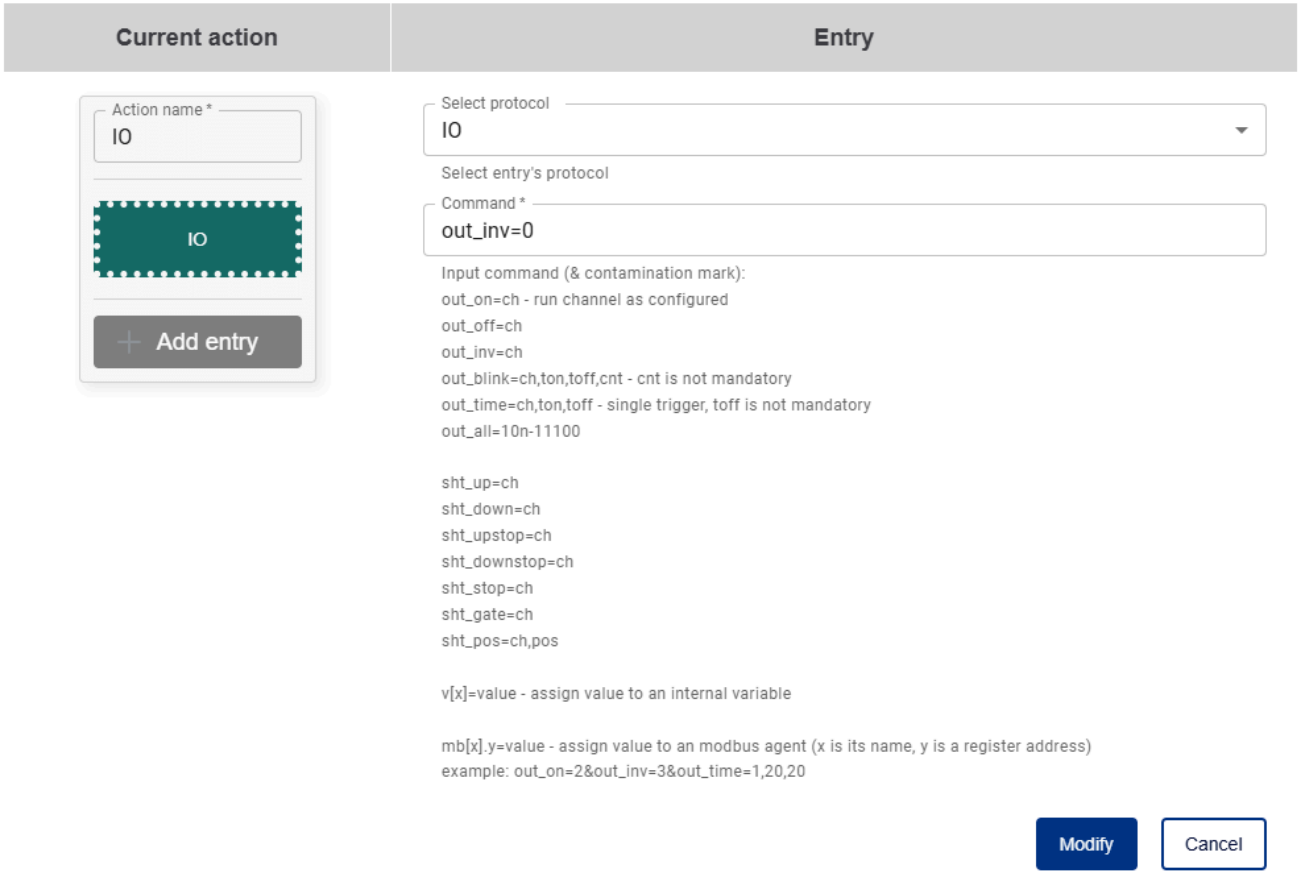


IO

- **Input command** – Command field:
  - `in_cnt=ch,val` – Set the `ch` channel counter value to `val`,
  - `out_on=ch` – Activate output channel `ch`,
  - `out_off=ch` – Deactivate output `ch`,
  - `out_inv=ch` – Output `ch` state invert,
  - `out_blink=ch,ton,toff,cnt` – Periodic output `ch` control. Parameters:
    - `ton` – Activation time (in seconds),
    - `toff` – Deactivation time (in seconds),
    - `cnt` – Number of activation cycles (not required).
  - `out_time=ch,ton,toff` – Activate output `ch` for the time defined in `ton`, after `toff` time passes. The `toff` parameter is not required – omitting it will activate the output without a delay,
  - `out_all=10n-11100` – Command defining the state of all available outputs. Each digit represents a subsequent output:
    - `1` – Activated,
    - `0` – Deactivated,
    - `n` – State invert,
    - `-` – No state change.
  - `sht_up=ch` – Open the shutter,
  - `sht_down=ch` – Close the shutter,
  - `sht_upstop=ch` – Open the shutter and stop,
  - `sht_downstop=ch` – Close the shutter and stop,
  - `sht_stop=ch` – Shutter stop,
  - `sht_gate=ch` – ,
  - `sht_pos=ch,pos` – Move shutter to the specified position,
  - `v[x]=value` – Assign a `value` to the variable `x`,
  - `mb[x].y=value` – Write the `value` to the Modbus register `y`, of the agent `x`,
  - `dev[x].o[y]=0|1` – Assign 0 or 1 to the external output `y` in device `x`.

Example

- **Input command:** `out_inv=0`



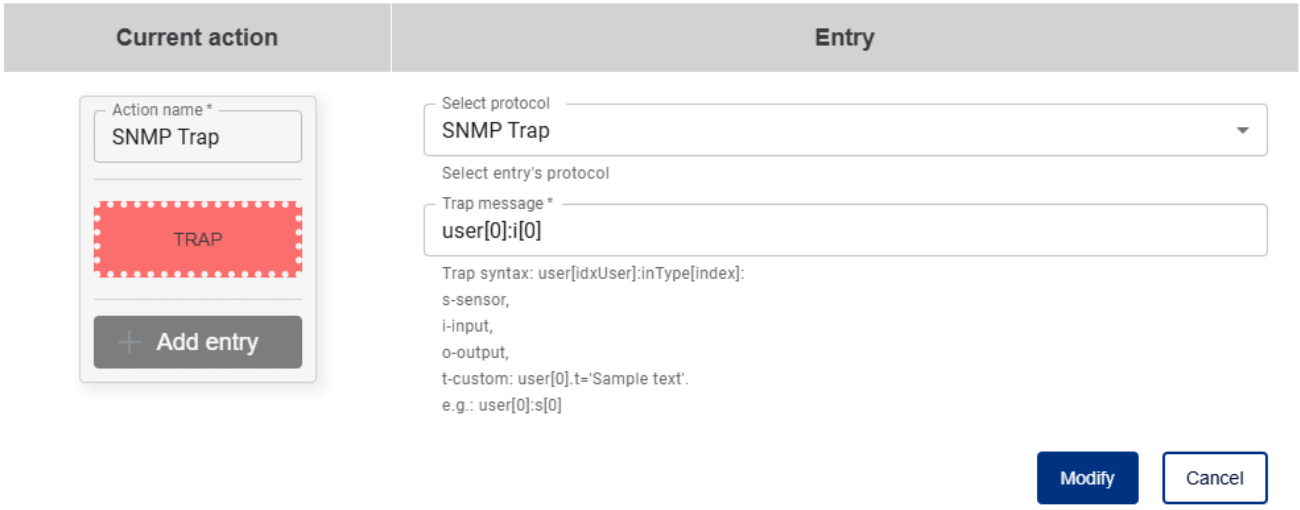
This setting will invert the state of output 0.

SNMP Trap

- **Trap message** – Message syntax.

Example

- **Trap message:** `user[0]:i[0]`

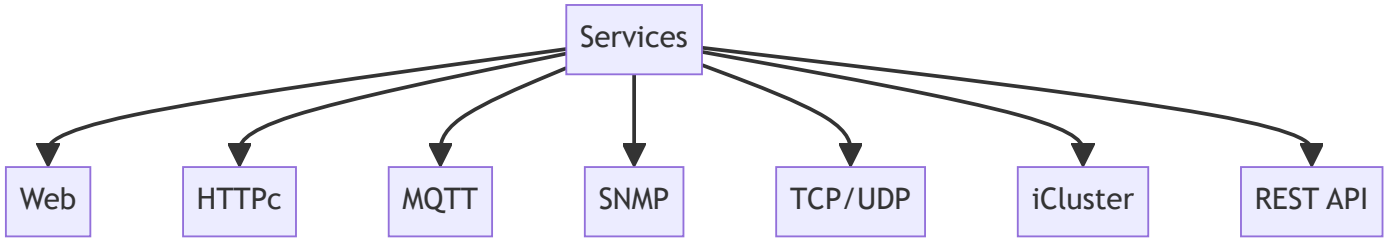


Sent data:

Description	Source	Time	Severity
.1.3.6.1.4.1.42814.10.1.1.100.0.5	192.168.22.71	2025-05-14 10:41:34	

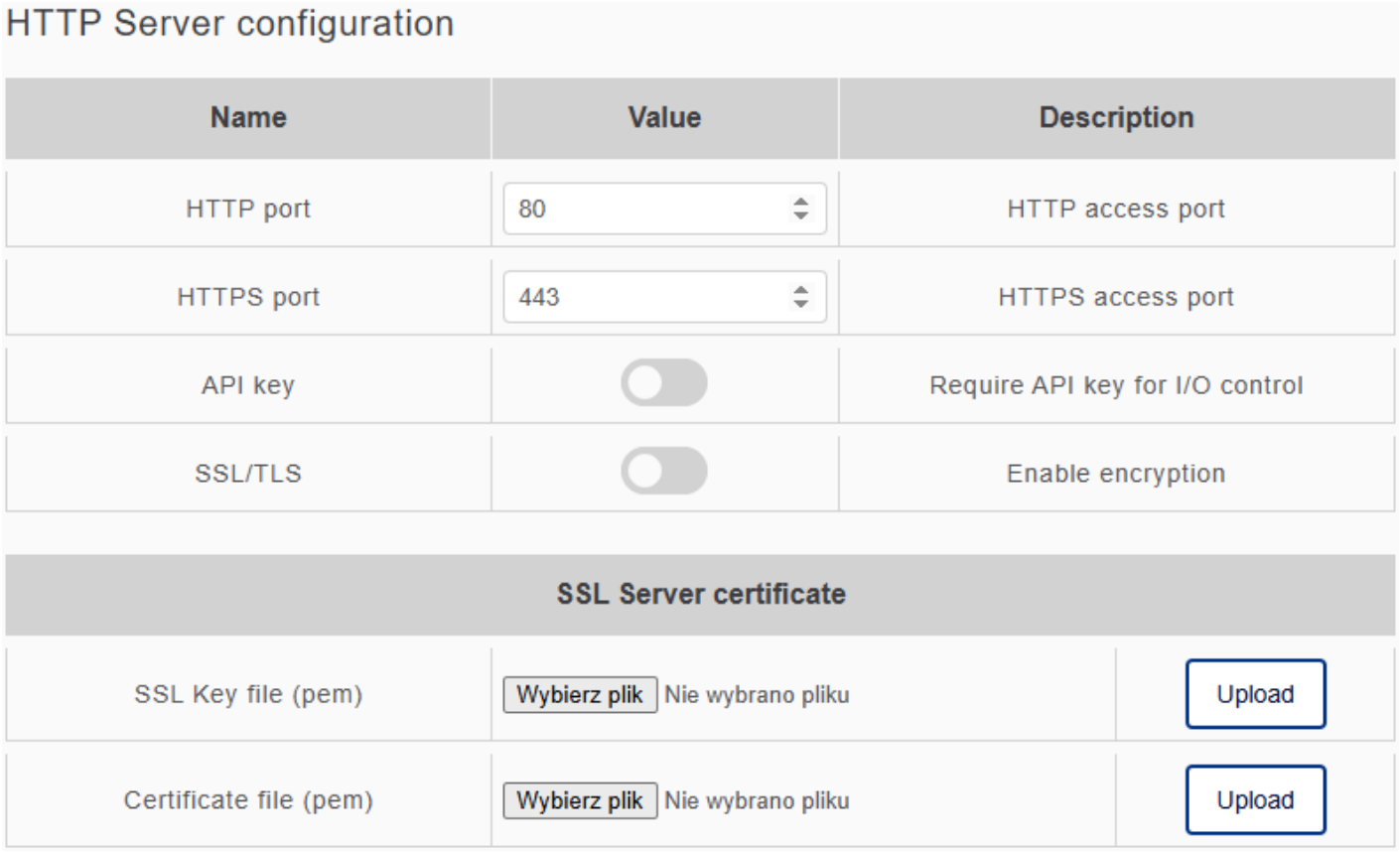
Communication protocols

In the **Services** tab, options enabling detailed communication protocol configuration options are located:



Web

In this section, the user can customize the device's network interface settings, manage resource access or modify network connection parameters.



- **HTTP Port**,
- **HTTPS port**,
- **API key** – Enable/disable API key requirement for I/O control,
- **SSL/TLS** – Enable/disable encryption,
- **Select Key file (pem)** – Allows upload of SSL server key (pem format),
- **Select CSR file (pem)** – Allows upload of CSR server key (pem format).

HTTPc

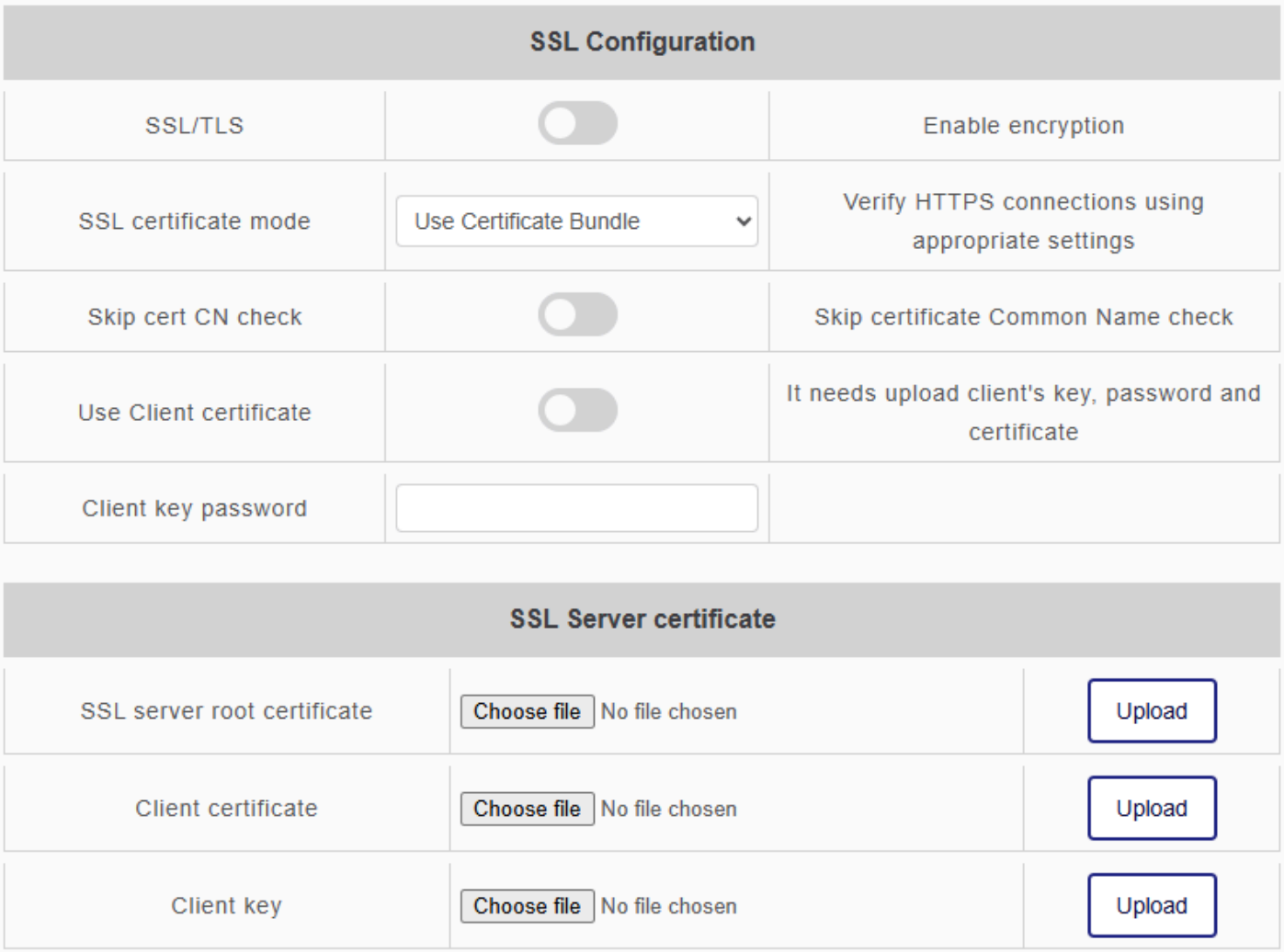
In this section, the device can be configured so it initiates HTTP connections with defined servers or services. URL addresses, request parameters and other connection details can be defined here. The device can send event data via HTTP/HTTPS, using GET or POST.

HTTP Client configuration

Name	Value	Description
HTTP Client		Enable HTTP Client
Server		Remote server address: IP or DNS-registered name
HTTP port	0	HTTP access port
HTTP Method	GET	HTTP request default method
Content type	text/plain	Content-type header
Resource	/	HTTP resource i.e. /resource.php
API key		Require API key for I/O control
Authentication to Server		
Authentication type	None	Select the authentication method for server communication
HTTP ping request		
HTTP ping request interval	0	Ping time in secs. 0-disable
HTTP ping request	HTTP ping requests to test the availability of web servers and APIs. These can be used in health testing or monitoring systems.	

Type @ and select variable...
You can mention variables like current time or module name

Source format Show help



- **HTTP Client** – Enable **HTTP Client** service,
- **Server** – HTTP server address, to which information will be sent,
- **HTTP Port** – Port, on which the HTTP server listens,
- **HTTP Method** – Message sending method: GET / POST / PUT / DELETE,
- **Content type**:
  - text/plain,
  - json,
  - xml,
  - form.
- **Resource** – Resource the module will refer to,
- **API key** – Enable/disable API key requirement for I/O control,
- **Authentication type** – Server authentication type,
  - **None**,
  - **Basic Authentication** – Username and password authorization:
    - **User**,
    - **Password**.

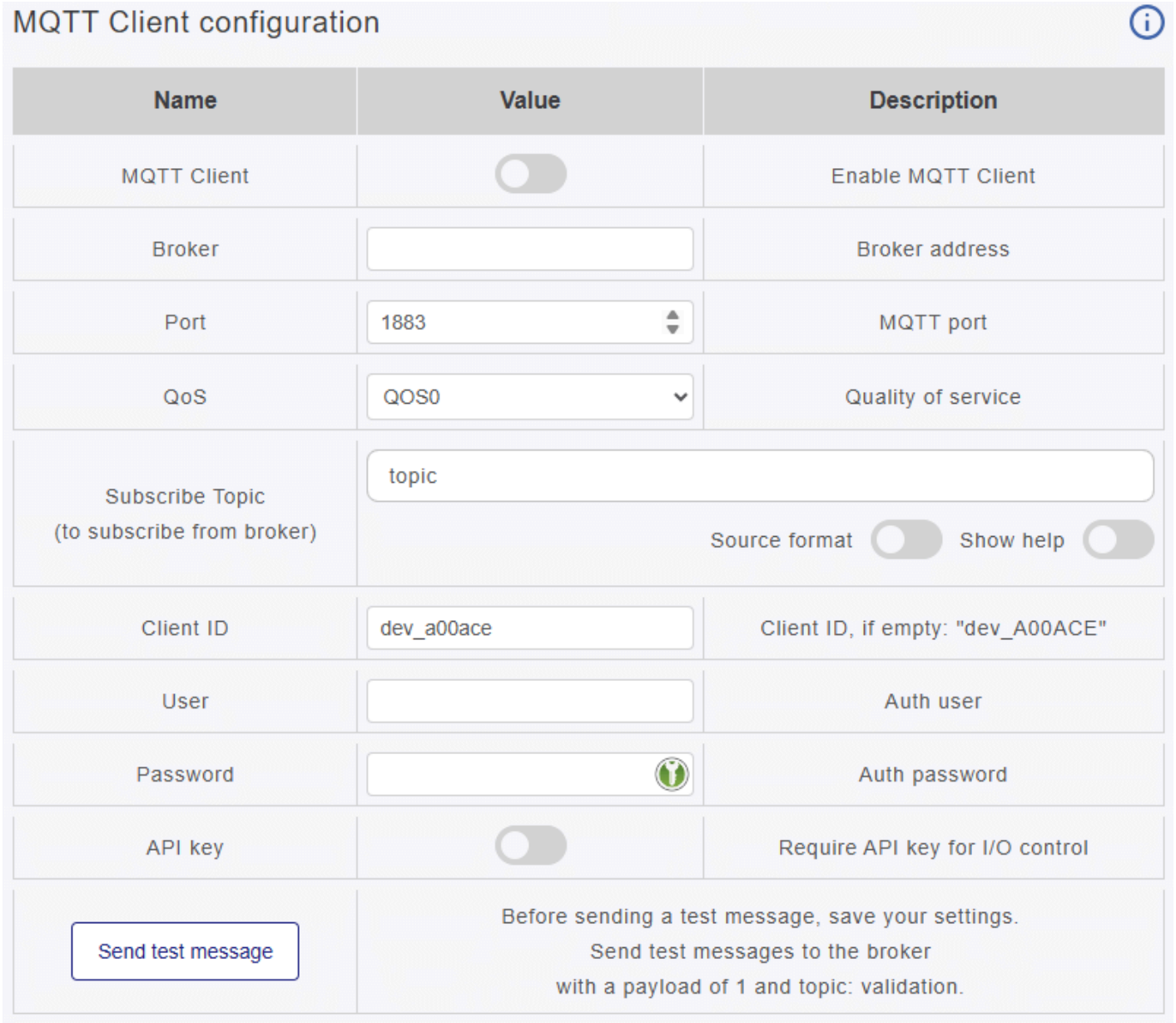
    - **X-API-KEY** – API key authorization:
        - **API Key**.
    - **Bearer Token**
        - **API Key**.
- **HTTP ping request interval**,
- **HTTP ping request** – Ping request syntax,
- **SSL/TLS** – Enable/disable encryption,
- **SSL certificate mode**:
    - **Use Certificate Bundle**,
    - **Use Uploaded Certificate**,
    - **Insecure! Disable SSL verification**,
- **Skip cert CN check** – Skip certificate common name check,
- **Use Client certificate**,
- **Client key password**,
- **SSL server root certificate**,
- **Client certificate**,
- **Client key**.

MQTT

This tab is used to configure MQTT broker communication parameters, allowing for data exchange with the publish-subscribe model. This allows the user to define key aspects such as topics, server address, port, and other connection parameters.
The device sends data to the server every minute and every time a value change occurs.

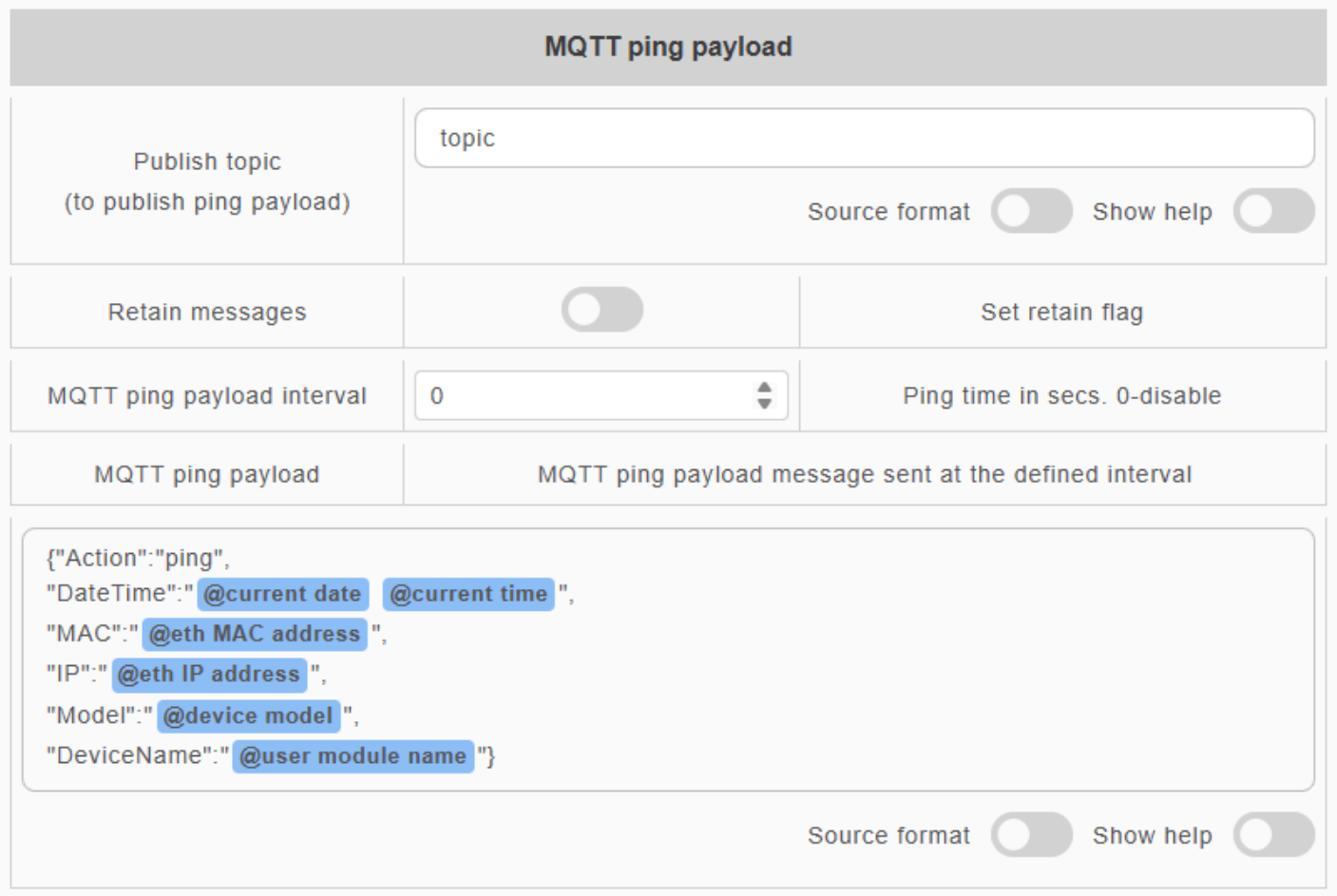
The data stream can be encrypted. After a connection with the MQTT broker is established, the users can subscribe to the data coming from the device.

There is no limit in the amount of subscribers that can receive data from a single device.



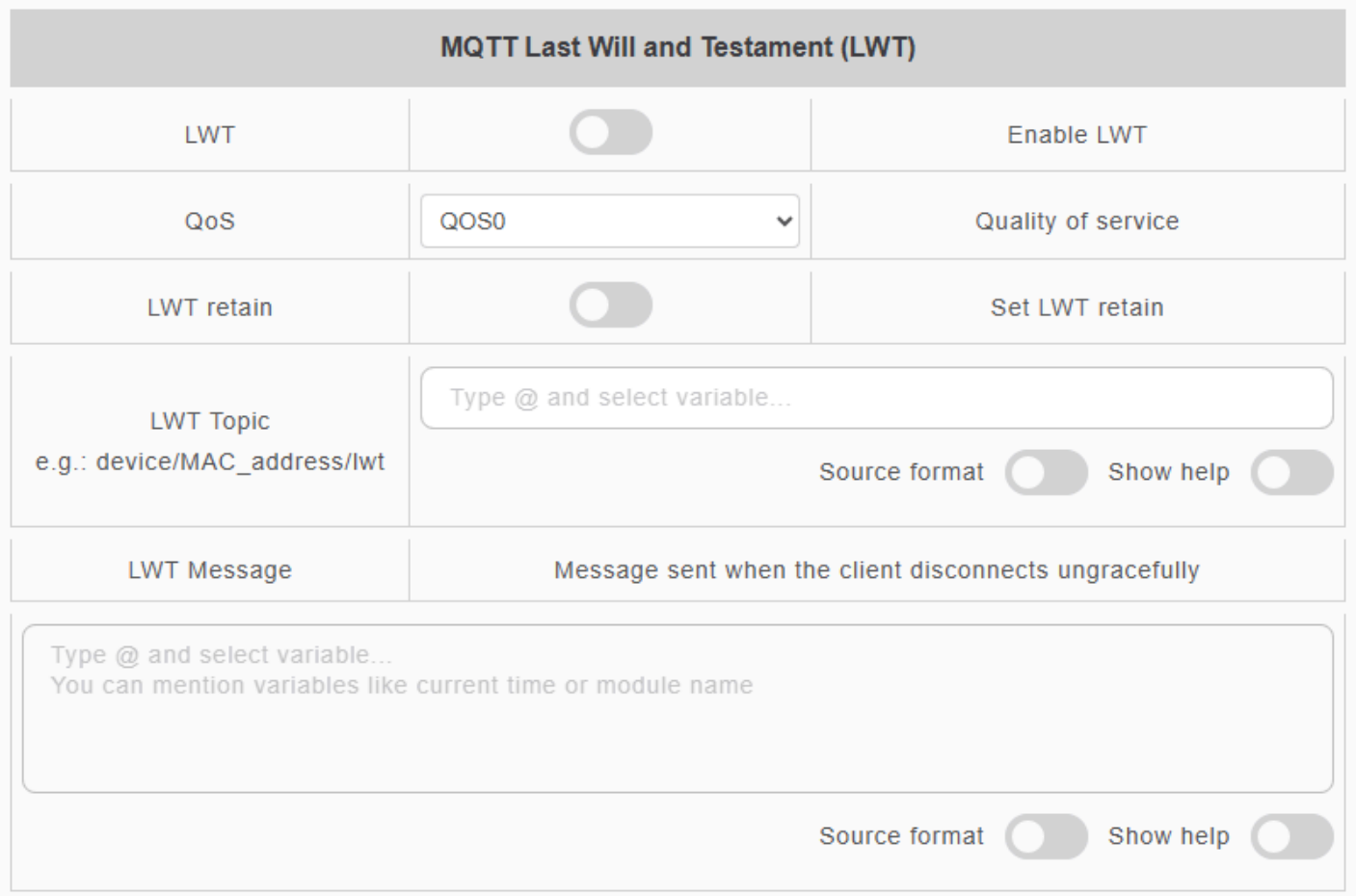
- **MQTT Client** – Enable MQTT service,
- **Broker** – MQTT broker address,
- **Port** – Port, on which the server listens (commonly 1883),
- **QoS** – Quality of Service level. Can assume one of three values: 0 (At most once), 1 (At least once), 2 (Exactly once),
- **Subsribe topic** – The topic must be in the format (eg. /sensor/home – without „/” at the end),
- **Client ID**,
- **User** – (optional) MQTT user name,
- **Password** – (optional),
- **API key** – Enable/disable API key requirement for I/O control,
- **Send test message** - Send a test message to the broker - topic **validation**, payload **1**.

The module can periodically send its data to the Broker. This functionality can be used to verify the integrity of notifications or for periodic data reading.

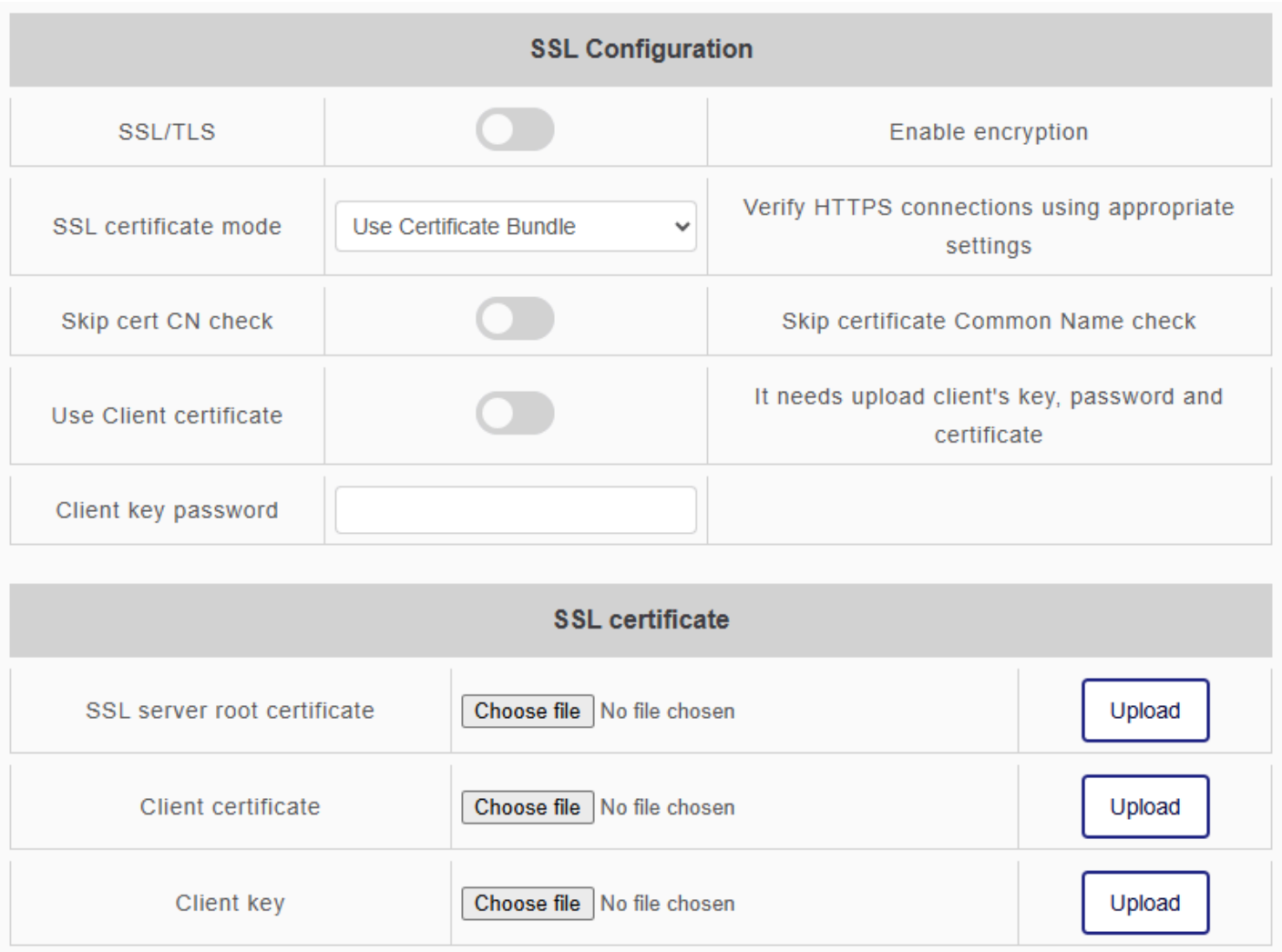


- **Publish topic** – Topic, to which data will be sent periodically,
- **Retain messages** – If enabled, the message will be saved on the broker an will be available for new topic subscribers,
- **MQTT ping payload interval** – Time interval between the data packets. Setting `0` disables the function,
- **MQTT ping paylaod**.

The device is equipped with the LWT mechanism, meaning **Last Will and Testament**. LWT is a mechanism that allows the MQTT client to send a message automatically in the event of a client error or MQTT broker connection loss.
The LWT mechanism allows the user to define a topic and message that will be published if the client looses connection.



- **LWT** – Enable/disable LWT,
- **QoS** – Quality of Service level. Can assume one of three values: 0 (At most once), 1 (At least once), 2 (Exactly once),
- **LWT retain** – If enabled, the LWT message will be saved on the broker an will be available for new topic subscribers,
- **LWT Topic** – Topic that the LWT message will be published at,
- **LWT Message** – LWT message contents.



- **SSL/TLS** – Enable/disable encryption,
- **SSL certificate mode**:
  - **Use Certificate Bundle**,
  - **Use Uploaded Certificate**,
  - **Insecure! Disable SSL verification**,
- **Skip cert CN check** – Skip certificate common name check,
- **Use Client certificate**,
- **Client key password**.
- **SSL server root certificate**,
- **Client certificate**,
- **Client key**.

Tip

After making changes, apply them using the Save button.

> **Tip**
>
> **For the Inveo broker, the settings should be as follows:**
> - MQTT Address**: mqtt.inveo.com.pl**
> - MQTT Port**: 1883**
> **A computer with the Inveo Monitoring application installed can be used as a broker. To do that, type the PC's IP address into the MQTT Address field.**

> **Tip**
>
> **Make sure that the selected Topic is unique, eg. /daxi/MAC address.**

SNMP

This section allows configuration of SNMP protocol settings, used for monitoring and network device management. The module is equipped with the v2c or v2c and v3 *(version dependent)* SNMP servers. Depending on the SNMP version selected, different parameters are available.

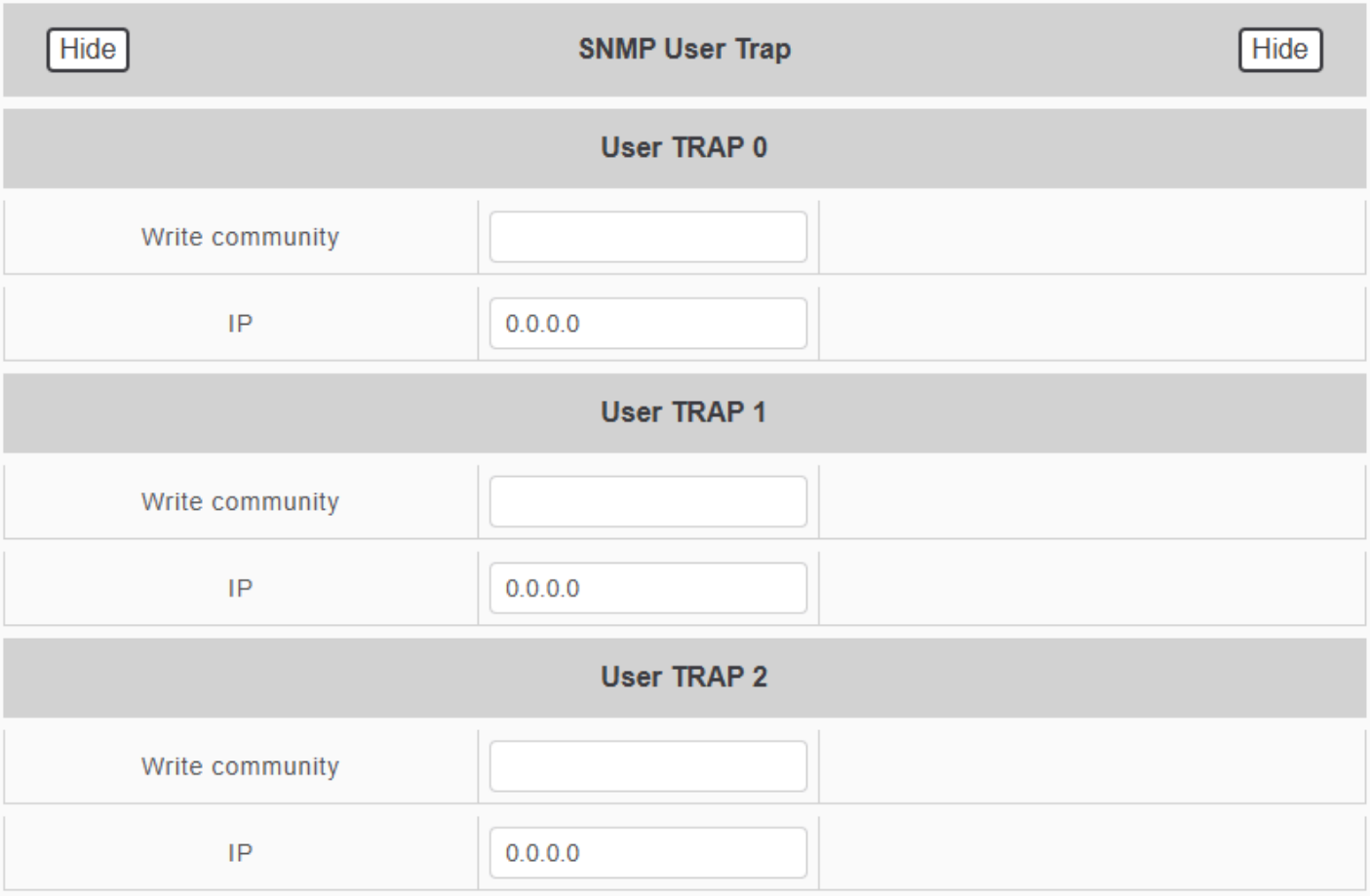
SNMP v2c

SNMP configuration

Name	Value	Description
SNMP		Enable SNMP
SNMP version	2c	Select SNMP version
sysDescr		
sysContact		
sysName		
sysLocation		
Read community		SNMP v2c only
Write community		SNMP v2c only

- **SNMP** – Enable/disable SNMP service,
- **SNMP version** – **v2c** or **v3**,
- **sysDescr**, **sysContact**, **sysName**, **sysLocation** – Optional system identification fields,

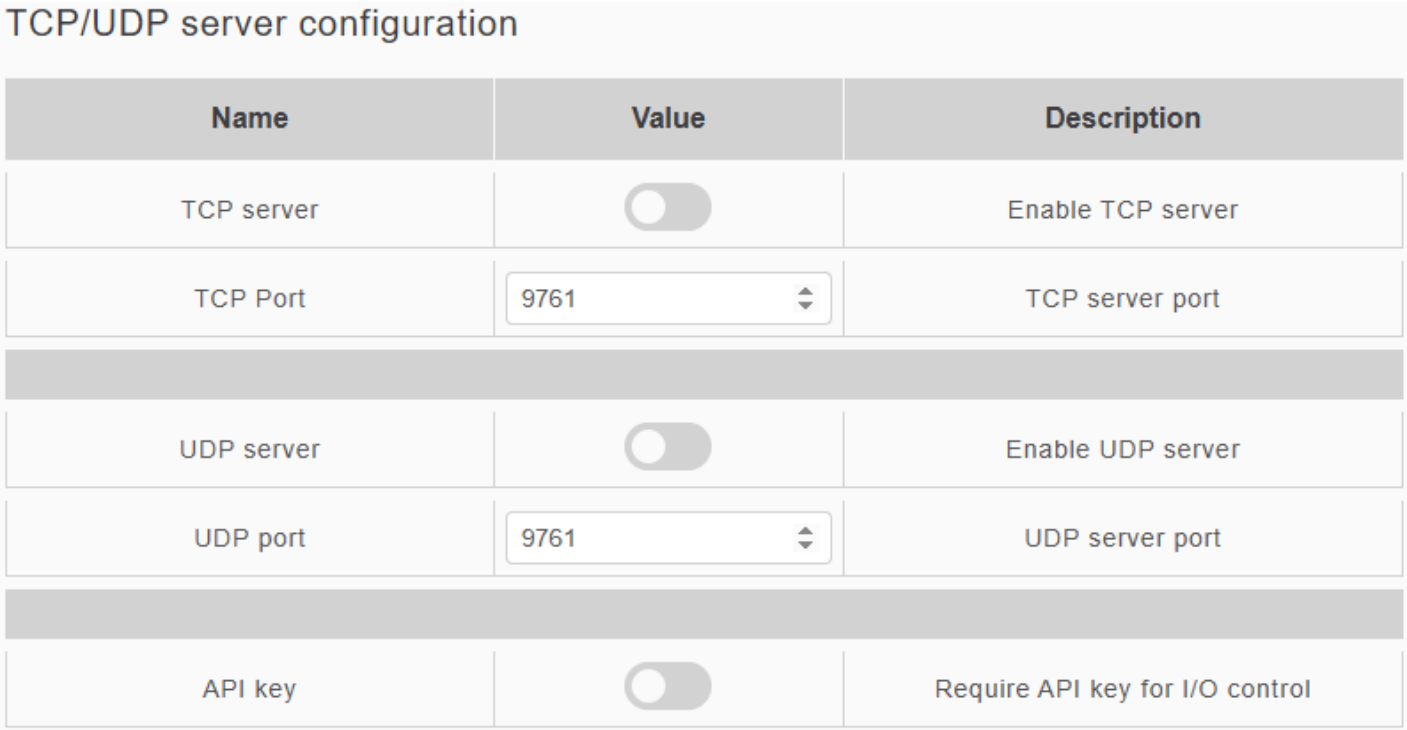
- **Read community** – Data read password,
- **Write community** – Data write password.



- **Write community** – Data write password,
- **Trap IP** – Address, to which trap messages will be sent.

TCP/UDP

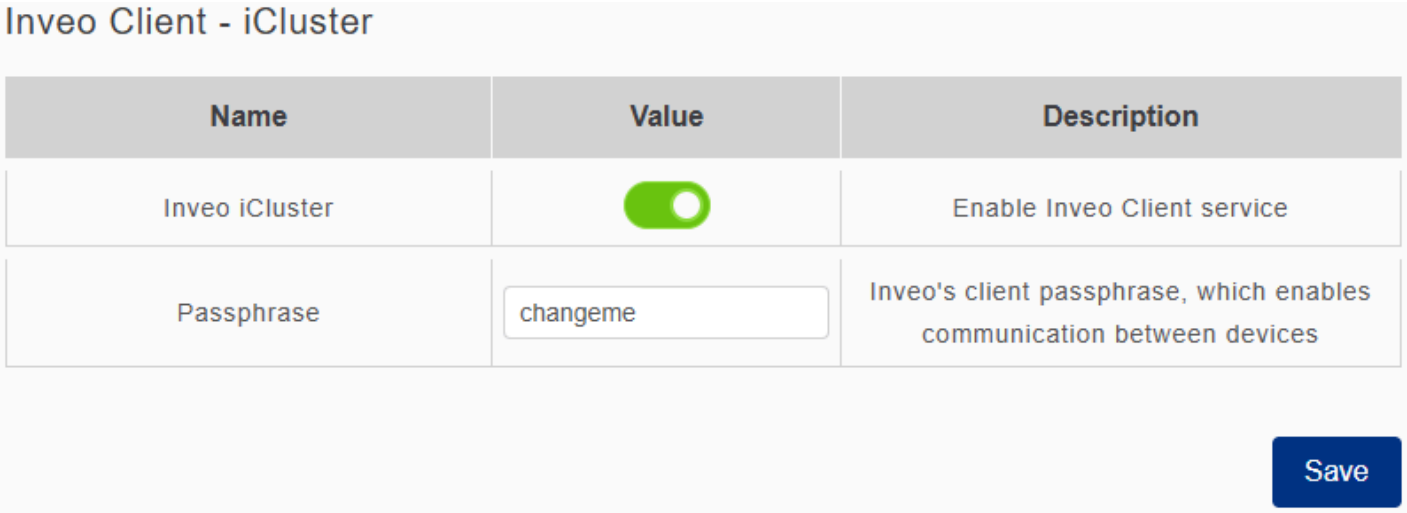
This tab allows the user to enable and configure TCP and UDP protocols support. The user can customize settings such as ports and communication parameters, providing flexibility in device configuration in compliance with network requirements. By sending commands to the device's IP address and port using an appropriate protocol, the user can control the outputs state. Available commands – see **IO commands**.



- **TCP server**,
- **TCP port**,
- **UDP server**,
- **UDP port**,
- **API key** – Enable/disable API key requirement for I/O control.

iCluster

iCluster allows multiple Inveo devices to interconnect within a single LAN. IQIO can operate as an iCluster client, allowing the device to oversee its parameters such as: sensor readings, input/output state, output control etc.



- **Inveo iCluster** – Enable/disable **iCluster** service,
- **Passphrase** – Password used by the Inveo client that allows communication between devices. The password set here will be required in the device that would like to establish a connection with a Daxi device.

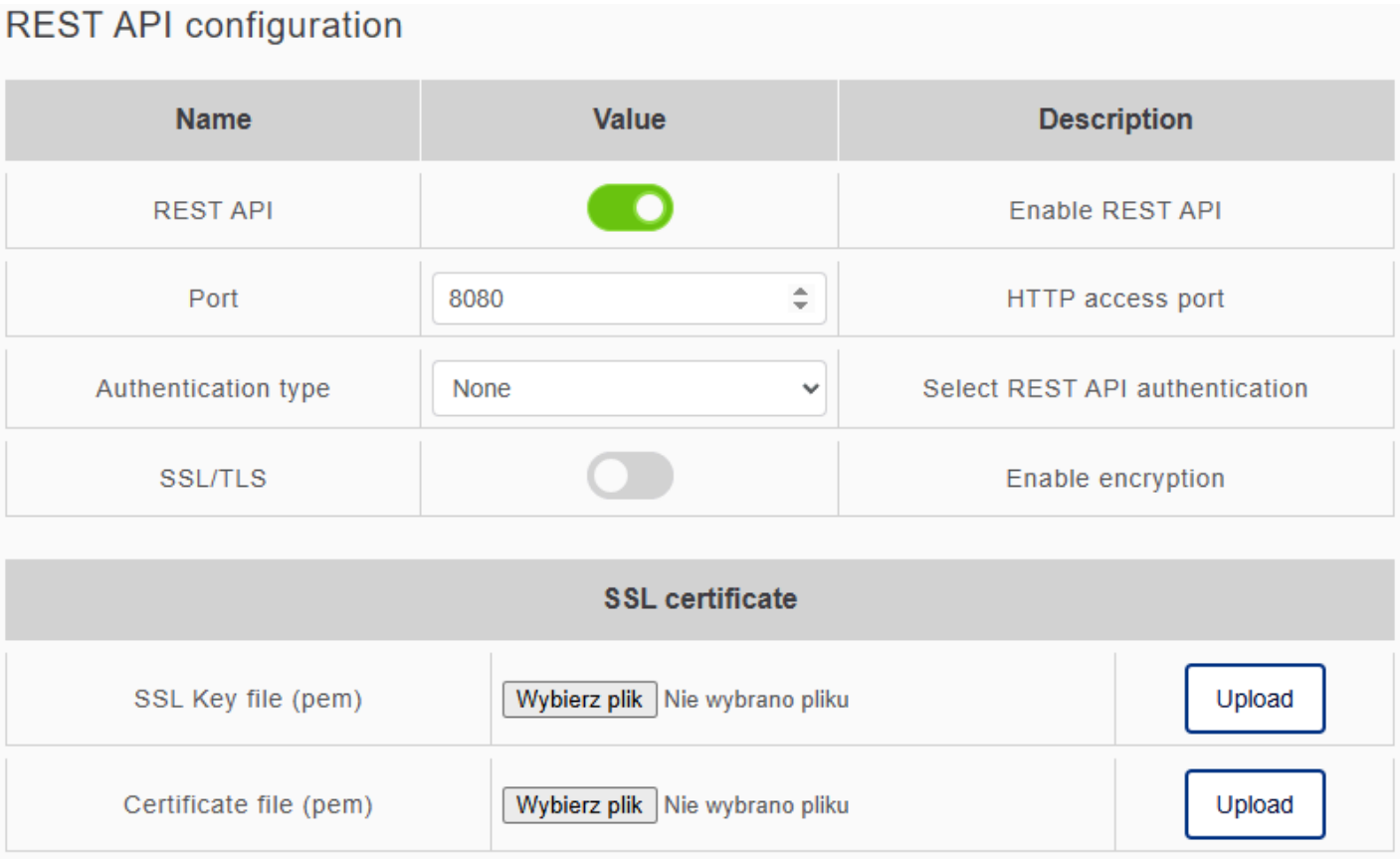
Built-in variables

The following table contains example internal variables that allow for precise transfer of module operation data. Those variables are key elements in configuration, HTTP Client, MQTT, TCP/UDP notifications etc.

Syntax	Example	Description
`%i[x]%`	`%i[0]%`	Input `x` state Example: returns the state of input No. 0
`%i[x].cnt%`	`%i[0].cnt%`	Input `x` counter value
`%i[x].name%`	`%i[0].name%`	Input `x` name
`%i[x].idx%`	`%i[0].idx%`	Input `x` index
`%i[x].valTxt%`	`%i[0].valTxt%`	Input `x` state (text)
`%o[x].on%`	`%o[0].on%`	Output `x` state
`%o[x]%`	`%o[0]%`	Output `x` coil state

`%o[x].name%`	`%o[0].name%`	Output `x` name
`%o[x].idx%`	`%o[0].idx%`	Output `x` index
`%o[x].valTxt%`	`%o[0].valTxt%`	Output `x` state (text)
`%v[x]%`	`%v[1]%`	Virtual variable state
`%time%`		Current time
`%date%`		Current date
`%timedate%`		Current time and date
`%ts%`		Current timestamp – number of seconds since the specified date: 1st of January 1970
`%mod_name%`		User-defined module name
`%mod_model%`		Device model
`%eip%`		Ethernet IP address *(IQIO IO ETH)*
`%emac%`		Ethernet MAC address *(IQIO IO ETH)*
`%wip%`		Wi-Fi IP address *(IQIO IO Wi-Fi)*
`%wmac%`		Wi-Fi MAC address *(IQIO IO Wi-Fi)*
`%sunrise%`		Calculated sunrise time
`%sunset%`		Calculated sunset time

REST API



- **REST API** – Enable REST API,
- **Port**,
- **Authentication type**:
  - **None**,
  - **Basic authentication** – User name and password authentication,
  - **API KEY** – API key authentication, utilizing a key generated in **Administration / Access**. The key should be entered as a parameter, preceded with `apikey=`.
- **User** – User name used with Basic authentication,
- **Password** – Password used with Basic authentication,
- **SSL/TLS** – Enable/disable encryption.
- **SSL Key file (pem)** – Allows upload of SSL key (pem format),
- **Certificate file (pem)** – Allows upload of certificate file (pem format).

Example

API key value: `12345678901234567890`
REST API request syntax: `http://<IP_address>:<port>/io/outputs/1?apikey=12345678901234567890`

Factory settings, backup

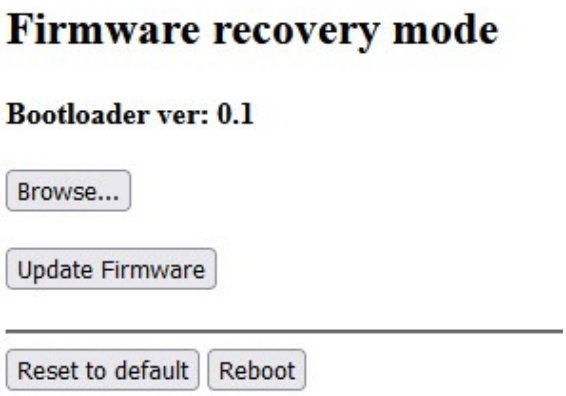
Emergency firmware restore / restoring factory settings

If device failure occurs, preventing normal access to the webpage, the emergency procedure should be put into effect:

1. Disconnect the power supply,
2. Depress the **RESET** button,
3. Power up the device and connect it to LAN,
4. Without releasing the **RESET** button, open the device's webpage:
    - **IP address:** 192.168.111.15
    - **IP mask:** 255.255.255.0

> **Tip**
>
> **In order to access the 192.168.111.15 address, computer's IP address must be in the same subnet (example computer address: 192.168.111.1).**

Referring to the specified IP address will give access to the device's bootloader. The **RESET** button can be released after the following webpage opens:



In this window, the following options are available:

- Firmware upload,
- Default settings reset,
- Device restart.

Warranty and manufacturer's liability

> **Warning**
>
> **The manufacturer provides a two-year warranty for the device and post-warranty service for a period of 10 years from the date of introduction of the device to the market. The warranty covers all material and production defects.**

The manufacturer undertakes to respect the warranty agreement, if the following conditions are met:

- All repairs, changes, expansions and device calibrations are carried out by the manufacturer or an authorized service center,
- The power supply system meets the applicable standards,
- The device is operated in accordance with the suggestions presented in this manual,
- The device is operated in accordance with its intended purpose.

The manufacturer assumes no responsibility for consequences resulting from improper installation, improper use of the device, failure to comply with the instruction manual, and repairs made by unauthorized personnel.

> **Warning**
>
> **The device contains no user serviceable parts inside.**

Storage, operation and transport conditions

The device should be stored in enclosed rooms, where the atmosphere is free from vapours and corrosive substances:

- Environment temperature from -30°C to +60°C (-22°F - 140°F),
- Humidity from 25% to 90% (condensation unacceptable),
- Atmospheric pressure from 700 to 1060 hPa.

The device is intended to operate in the following conditions:

- Environment temperature from -10°C do +55°C (14°F - 131°F),
- Humidity from 30% to 75%,
- Atmospheric pressure from 700 to 1060 hPa.

Recommended transport conditions:

- Environment temperature from -40°C do +85°C (-40°F - 185°F),
- Humidity from 5% to 95%,
- Atmospheric pressure from 700 to 1060 hPa.

Installation and device operation:

- The module should be operated in accordance with recommendations provided later in this manual.

Disposal and decommissioning

In an event the device needs to be decommissioned (eg. after its intended life period is surpassed), it is recommended to contact the manufacturer or his representative, who are responsible to respond appropriately, i.e., to collect the device from the user. The user can alternatively contact companies specializing in electronic device or computer equipment disposal and/or decommissioning. Under no condition should the device be placed with other waste.