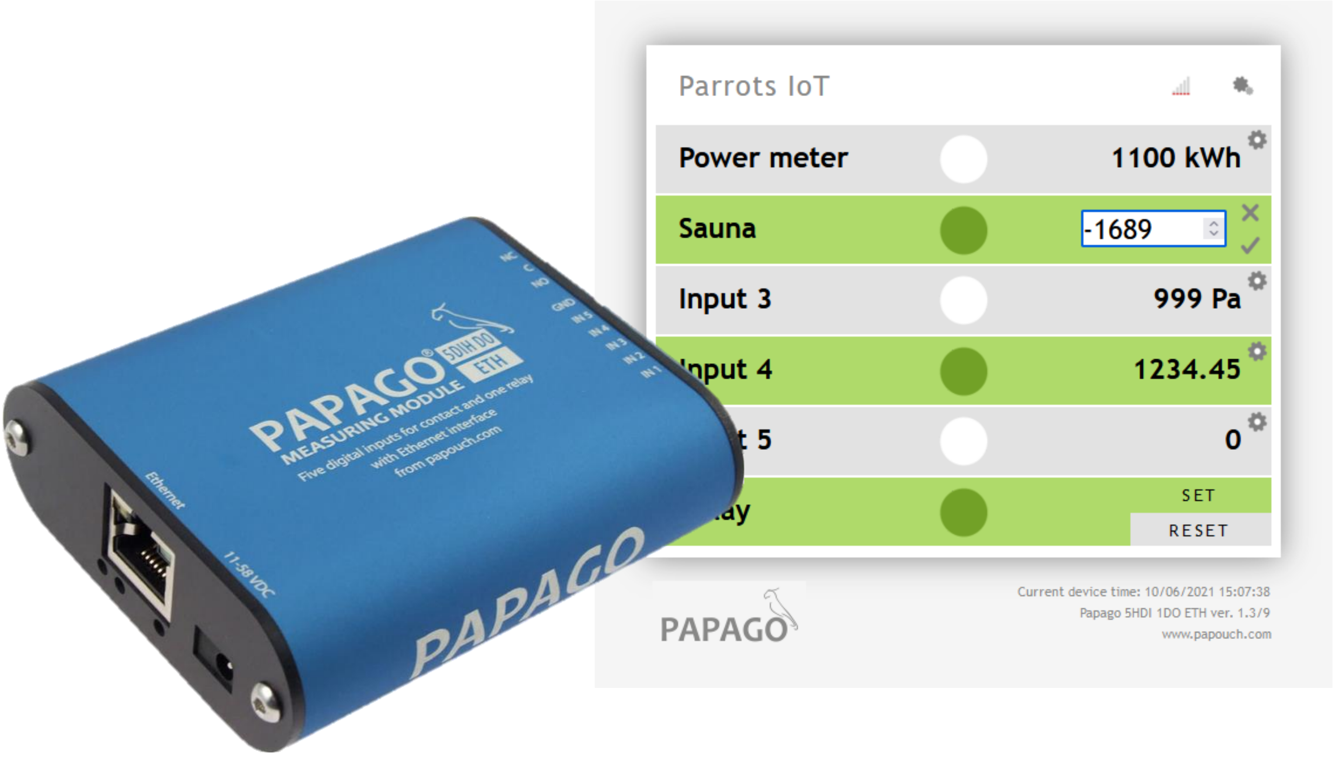
PAPAGO 5HDI DO

5 dry contact inputs and relay with counter functions

PAPAGO 5HDI DO ETH: Ethernet interface
PAPAGO 5HDI DO WIFI: WiFi interface


PAPAGO
MEASURING MODULE
5DIH DO ETH
Five digital inputs for contact and one relay with Ethernet interface from papouch.com
Ethernet
11-58 VDC
NC
C
NO
GND
IN 5
IN 4
IN 3
IN 2
IN 1
PAPAGO

Parrots IoT

Power meter	○	1100 kWh
Sauna	●	-1689
Input 3	○	999 Pa
Input 4	●	1234.45
t 5	○	0
ay	●	SET RESET

PAPAGO

Current device time: 10/06/2021 15:07:38
Papago 5HDI 1DO ETH ver. 1.3/9
www.papouch.com

PAPAGO 5HDI DO

Datasheet

Created: 5.8.2015

Last update: 23.8 2022 15:22

Number of pages: 36

Papouch s.r.o.

Address:

Strasnicka 3164
102 00 Prague 10
Czech Republic

Tel:

+420 267 314 267

Internet:

www.papouch.com

E-mail:

info@papouch.com

TABLE OF CONTENTS

Firmware versions

ETH v.1.3, WiFi v.1.4

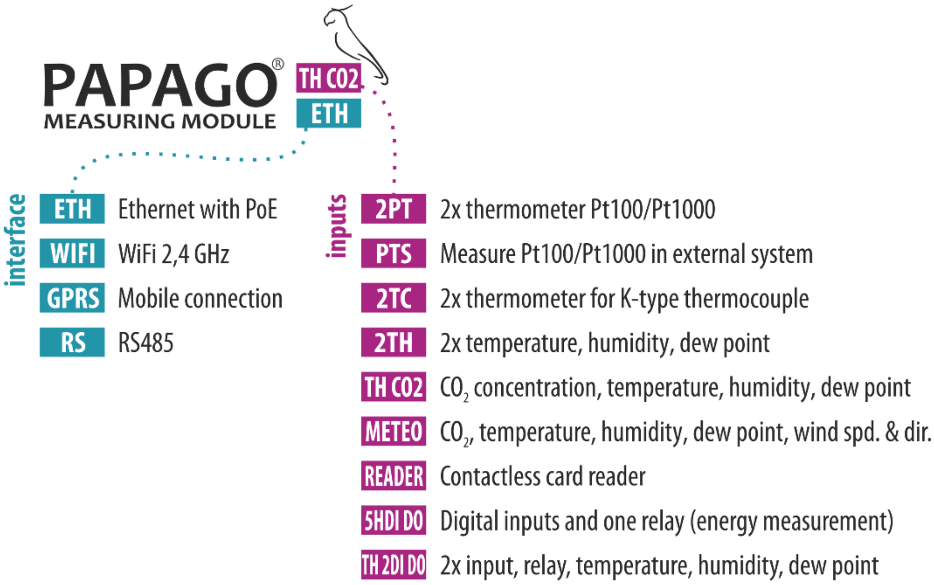
Parameter *per_index* added to the periodically sent HTTP GET. This parameter lets you easily know whether all the messages have been received.

Name change

- Papago 5DI 1DO has been changed to Papago 5HDI DO.
- Inputs have been modified to support 24V DC to allow direct connection to electrical meters (S0). The 5V version can be supplied upon request.

GETTING TO KNOW PAPAGO

PAPAGO is a family of devices with uniform appearance and communication capabilities. It allows to combine communication interfaces on one side and measuring sensors (inputs) on the other side.


PAPAGO®
MEASURING MODULE
TH CO2
ETH
interface
ETH Ethernet with PoE
WIFI WiFi 2,4 GHz
GPRS Mobile connection
RS RS485
inputs
2PT 2x thermometer Pt100/Pt1000
PTS Measure Pt100/Pt1000 in external system
2TC 2x thermometer for K-type thermocouple
2TH 2x temperature, humidity, dew point
TH CO2 CO_2 concentration, temperature, humidity, dew point
METEO CO_2, temperature, humidity, dew point, wind spd. & dir.
READER Contactless card reader
5HDI DO Digital inputs and one relay (energy measurement)
TH 2DI DO 2x input, relay, temperature, humidity, dew point

Applications

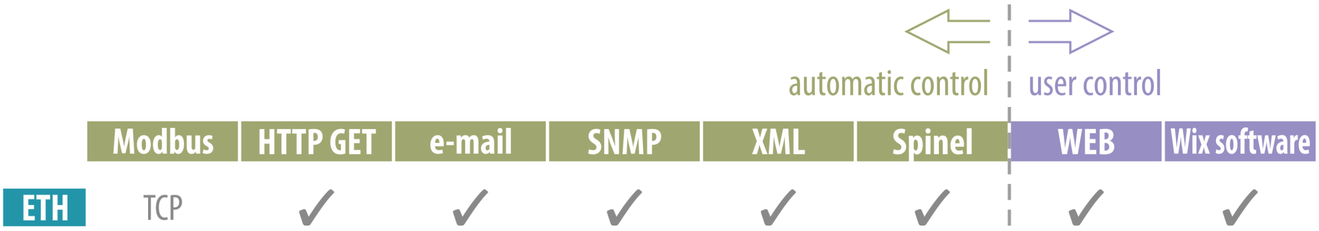
- Monitoring a production process
- **Monitoring energies consumptions**
- Two-state inputs surveillance (end point contacts, door contacts, window contacts etc.)
- Remote internet surveillance

Common Features

- Ethernet or WiFi interface to an internal website and many standard communication protocols.
- Ethernet versions with PoE power supply. This eliminates the need to use an external power supply, but the possibility to connect the AC adapter is available.
- Configuration of WiFi parameters via USB interface.
- Internal memory and real-time clock. Measured data including the time of measurement is automatically stored in the memory in the event that communication is lost. The data is automatically sent after the connection is restored.
- Elegant but robust metal box that can be mounted on a DIN rail. The box bears descriptions that allow connection without having to consult the manual. Also LED indicators for all important states help commissioning.
- The possibility to display, store and analyse data in the Wix program.

Communication Options

Depending on used interface PAPAGO has different communication options. PAPAGO can be controlled via a web interface or via software for Windows. Machine data-reading is possible using various standard methods, so PAPAGO can be easily integrated into your existing systems. You can choose the option that is appropriate for your location:



	Modbus	HTTP GET	e-mail	SNMP	XML	Spinel	WEB	Wix software
ETH	TCP	✓	✓	✓	✓	✓	✓	✓

Machine data-reading: Modbus TCP, HTTP GET with encryption, e-mail, SNMP, XML, Spinel

User control: Web interface, Wix software

Papago 5HDIDO module features

- 5 digital inputs for dry contact or S0 (electricity meter)
- 5 counters to connect to impulse outputs
- 1x switching contact (relay)

Papago family common features

- Family of measuring devices with Ethernet or WiFi interface.
- Modern WEB interface.
- User reading through WEB interface or WIX software.
- Machine data reading using HTTP GET, ModBus TCP, SNMP, Spinel, e-mail or XML protocols.
- HTTP GET encryption using a 128bit cipher.
- PoE or external power 11 to 58 V.
- PoE IEEE 802.3af standard.
- Power consumption of the Ethernet version 1.8 W typically.
- Power consumption of the WiFi version 0.5 W typically.

CONNECTION

1) Ethernet version: Connect the device by a normal uncrossed cable for computer networks to the switch.

2) Ethernet version: If the device cannot be powered by the switch via PoE according to the IEEE 802.3af standard, connect a power adapter to the coaxial connector next to the connector for the Ethernet. DC voltage in the range of 11-58 V is expected. (The positive pole is inside, the input for the power supply has reverse polarity protection.)

   WiFi version: Connect a power adapter to the coaxial connector next to antenna. DC voltage in the range of 11-58 V is expected. (The positive pole is inside, the input for the power supply has reverse polarity protection.)

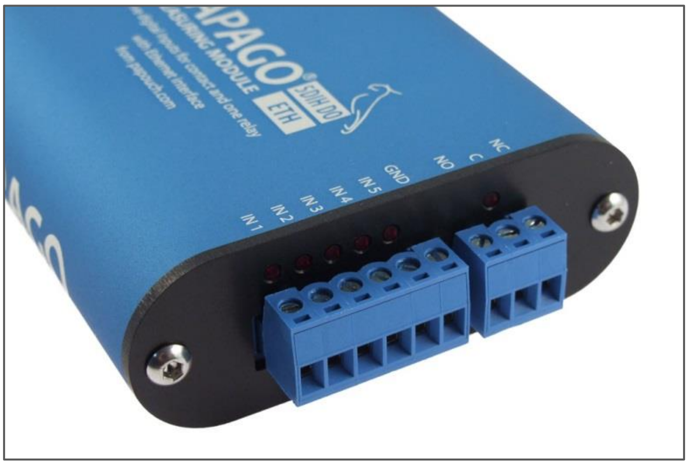


fig. 1 – Slip-on terminals separated for inputs and output

3) Connect inputs and output. Inputs are meant to be connected according to the following diagram against a common ground. The output is a two-state relay for up to 50V / 2A.

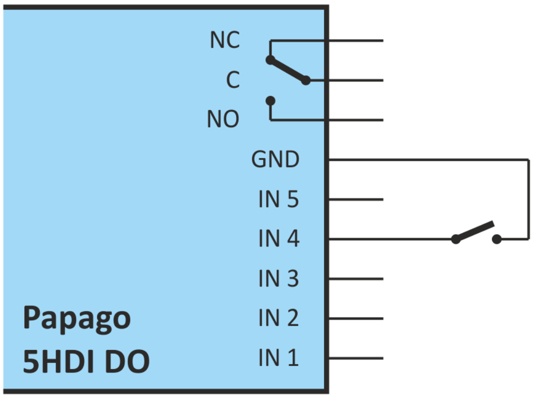
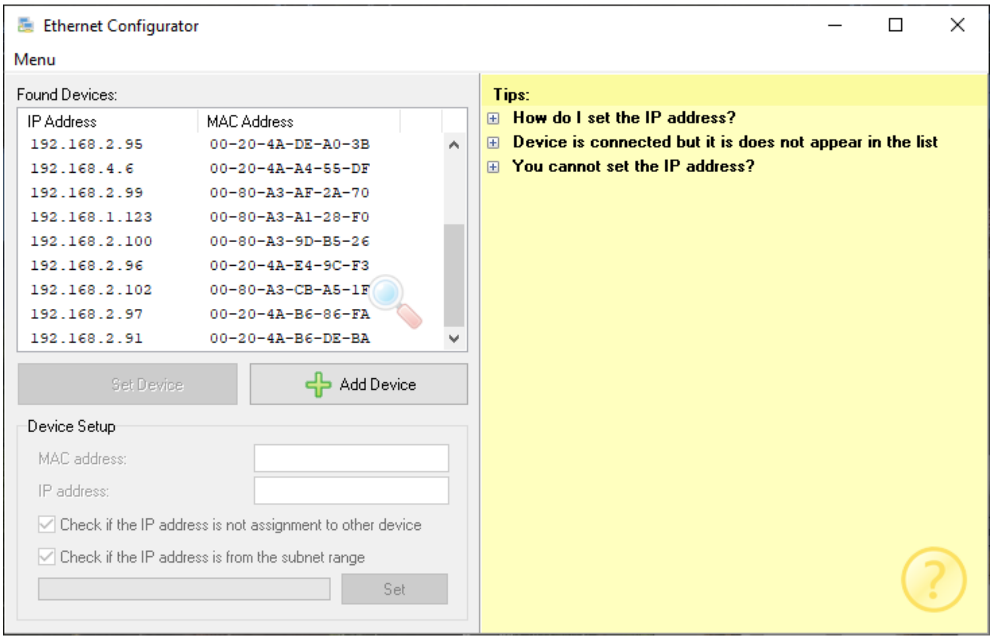


fig. 2 – Connecting a dry contact to one of tehe inputs and idle state of the relay output

4) Ethernet version: Now it is necessary to set the correct IP address of the device. The default IP address is 192.168.1.254 and network mask 255.255.255.0. If your network is not compatible with this range, set the IP address of the device using Ethernet Configurator.



Ethernet Configurator

Menu

Found Devices:

IP Address	MAC Address
192.168.2.95	00-20-4A-DE-A0-3B
192.168.4.6	00-20-4A-A4-55-DF
192.168.2.99	00-80-A3-AF-2A-70
192.168.1.123	00-80-A3-A1-28-F0
192.168.2.100	00-80-A3-9D-B5-26
192.168.2.96	00-20-4A-E4-9C-F3
192.168.2.102	00-80-A3-CB-A5-1F
192.168.2.97	00-20-4A-B6-86-FA
192.168.2.91	00-20-4A-B6-DE-BA

Device Setup

MAC address:

IP address:

☑ Check if the IP address is not assignment to other device

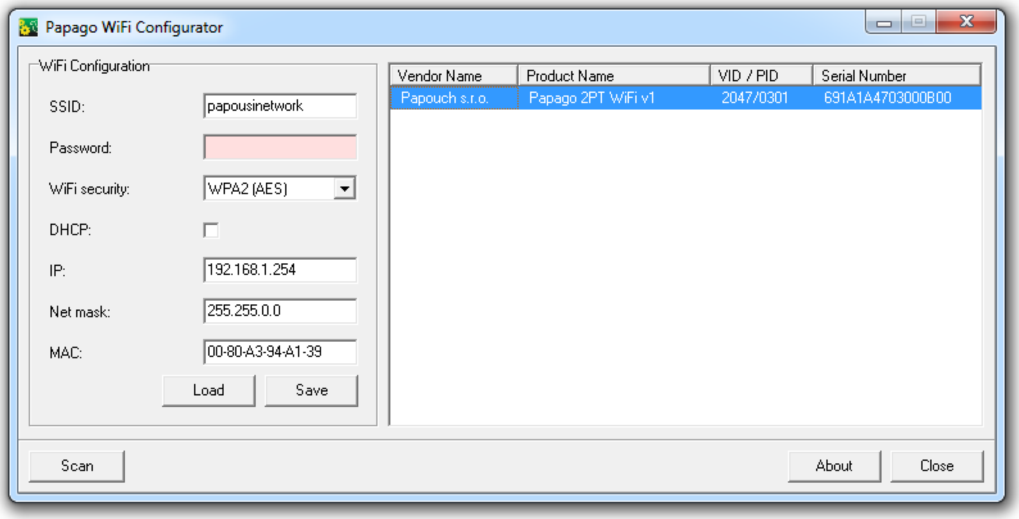
☑ Check if the IP address is from the subnet range

Tips:

- How do I set the IP address?
- Device is connected but it is does not appear in the list
- You cannot set the IP address?

Fig. 3 – Ethernet Configurator to set the IP address

5) <u>WiFi version:</u> Connect your Papago to a windows PC using the supplied micro-USB cable.[1] Run *Papago WiFi Configurator* software, you can download it on papouch.com. Set-up Papago to your WiFi network parameters so you can access it from that network.



Papago WiFi Configurator

WiFi Configuration

SSID: papousinetwork

Password:

WiFi security: WPA2 (AES)

DHCP: □

IP: 192.168.1.254

Net mask: 255.255.0.0

MAC: 00-80-A3-94-A1-39

Vendor Name	Product Name	VID / PID	Serial Number
Papouch s.r.o.	Papago 2PT WiFi v1	2047/0301	691A1A4703000B00

Fig. 4 – Setting WiFi parameters via USB

6) After setting the address, you can connect to a Web browser at the address specified as follows: *http://192.168.1.254/* (The example is given for the default IP address.)

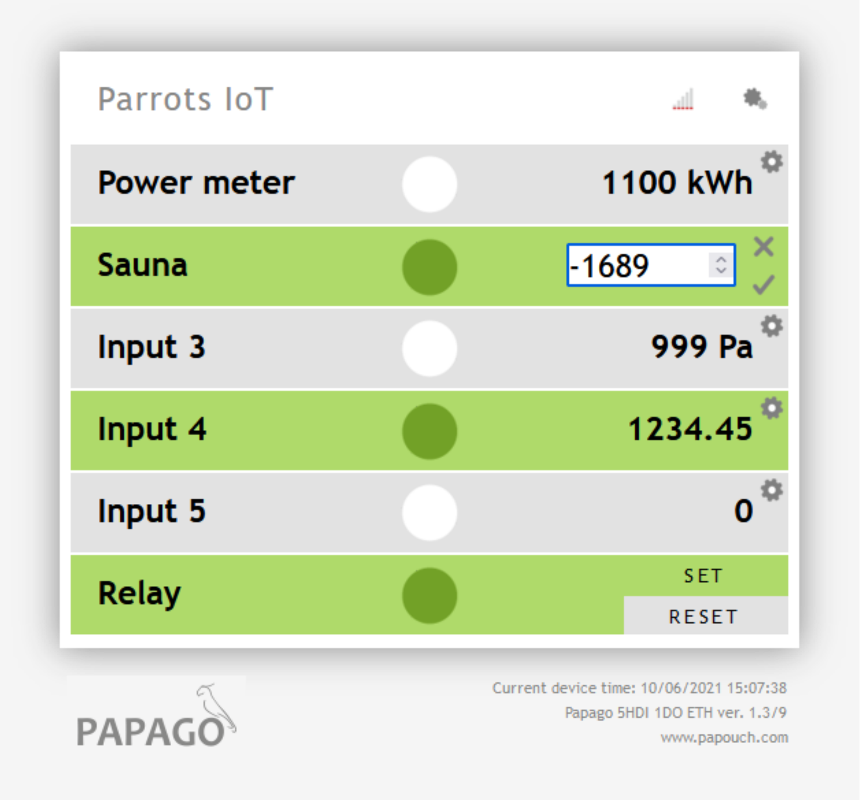
<u>Tip:</u> If your PC has the same subnet mask as your Papago, you can view its WEB interface by simply putting the Papago's name in the address line (The default name is NONAME.) This procedure only works for names without spaces.

[1] In Windows 7 or higher driver will be installed automatically.

CONFIGURATION

Configuration is done via a web interface. The basic network parameters can also be set via Telnet (see page 17). **The web interface** is accessible on the IP address of the device. (The default address is 192.168.1.254.)

After entering the IP address, the main page will appear showing the latest measured values.



Parrots IoT

Power meter	○	1100 kWh
Sauna	●	-1689
Input 3	○	999 Pa
Input 4	●	1234.45
Input 5	○	0
Relay	●	SET RESET

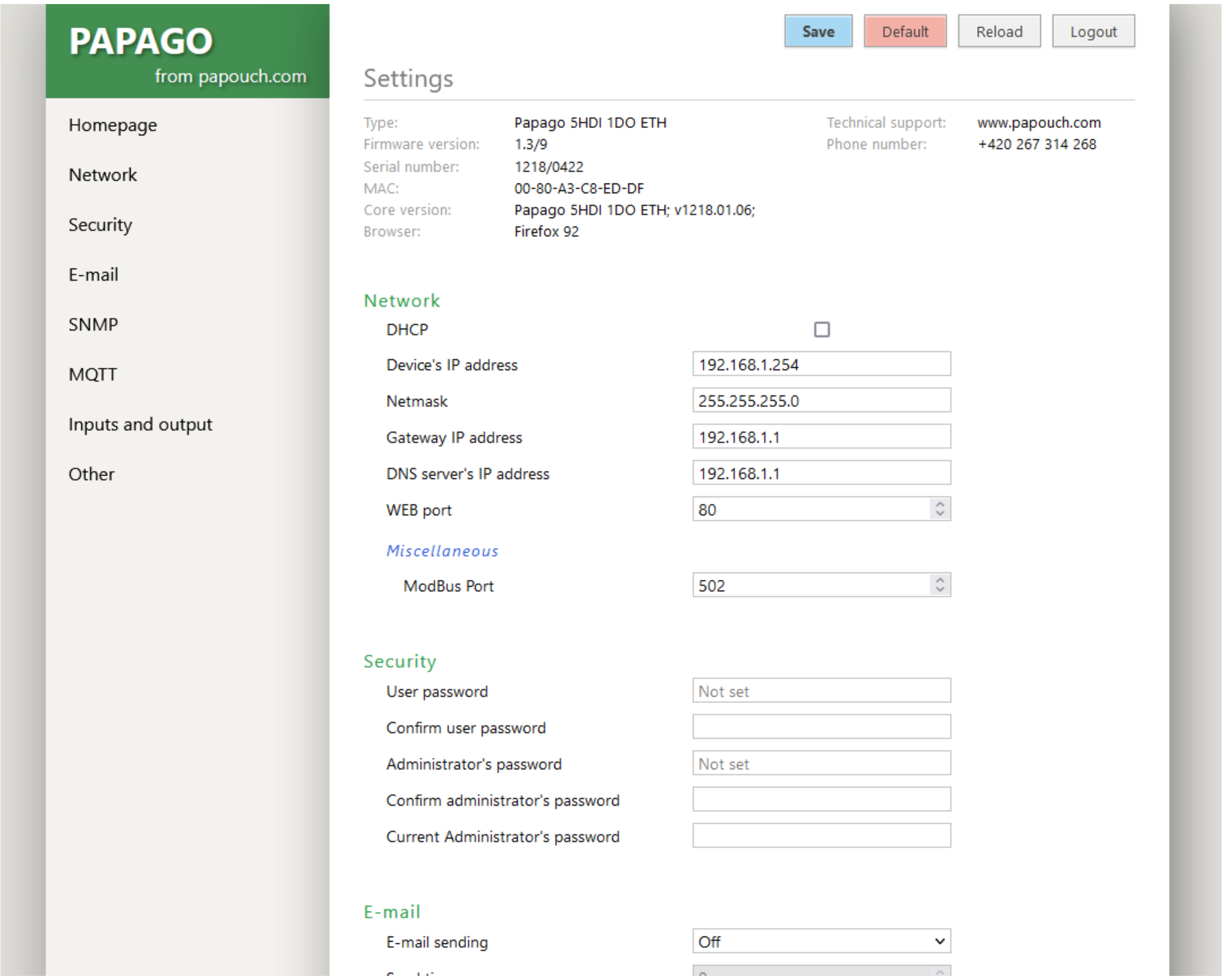
Current device time: 10/06/2021 15:07:38
Papago 5HDI 1DO ETH ver. 1.3/9
www.papouch.com

fig. 5 – Main page. Input 4 is open for editing the counter state.

The **web interface is secured** with a username and password. You can choose a separate password for the user (who can only display the values on the main page; user name is always **user**) and for the administrator (who can also change settings; administrator’s name is always **admin**).

The web interface is optimized for the following browsers: Mozilla Firefox and Google Chrome.

The configuration is displayed when you click the icon of gears in the upper right corner. The configuration is divided into sections according to the types of settings and is available in English and Czech.



Settings

Type:	Papago 5HDI 1DO ETH	Technical support:	www.papouch.com
Firmware version:	1.3/9	Phone number:	+420 267 314 268
Serial number:	1218/0422		
MAC:	00-80-A3-C8-ED-DF		
Core version:	Papago 5HDI 1DO ETH; v1218.01.06;		
Browser:	Firefox 92		

Network

DHCP	□
Device's IP address	192.168.1.254
Netmask	255.255.255.0
Gateway IP address	192.168.1.1
DNS server's IP address	192.168.1.1
WEB port	80

Miscellaneous

ModBus Port	502

Security

User password	Not set
Confirm user password	
Administrator's password	Not set
Confirm administrator's password	
Current Administrator's password	

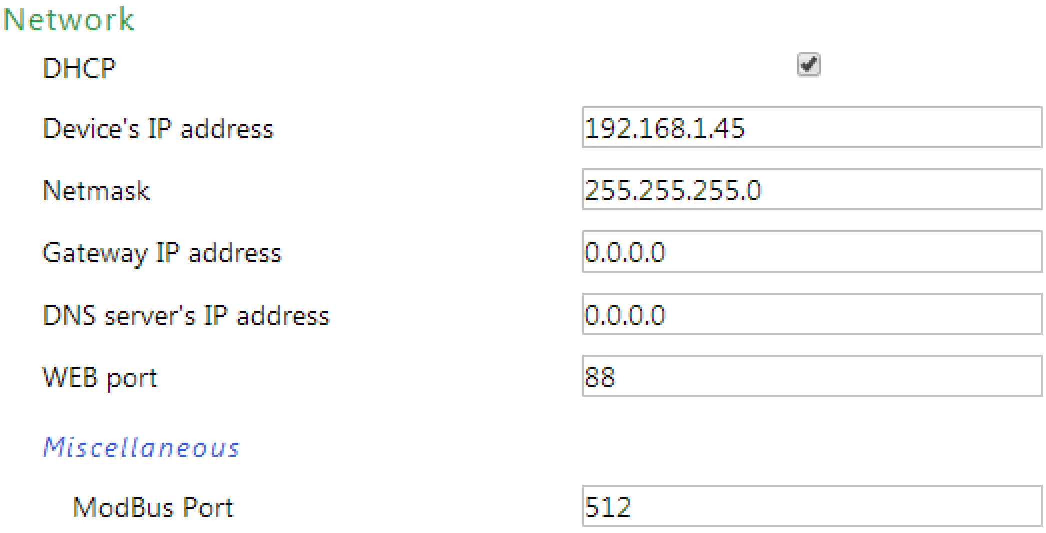
E-mail

E-mail sending	Off

fig. 6 - Configuration of Papago

Network

This section contains the configuration of network parameters.



Network

DHCP	☑
Device's IP address	192.168.1.45
Netmask	255.255.255.0
Gateway IP address	0.0.0.0
DNS server's IP address	0.0.0.0
WEB port	88

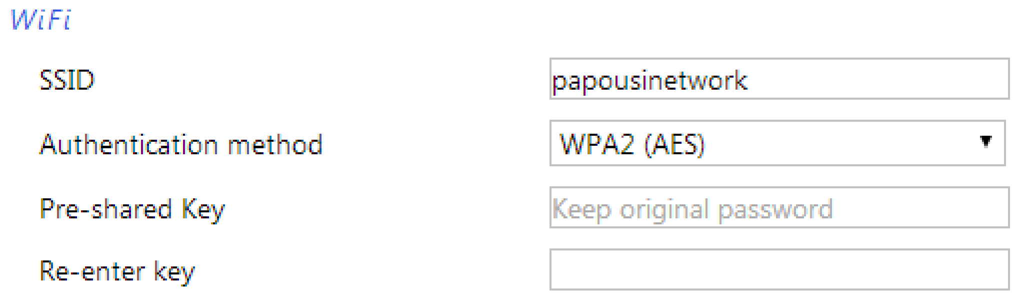
Miscellaneous

ModBus Port	512

fig. 7 - network configuration

If the box for assigning addresses via DHCP is ticked, the fields for *Device's IP address, Netmask, Gateway IP address* and *DNS server's IP address* are reset and upon reloading the settings they are filled again with data obtained from the DHCP server.

If you have a **version with WiFi interface** in the section *Network* is also following parameters:



WiFi

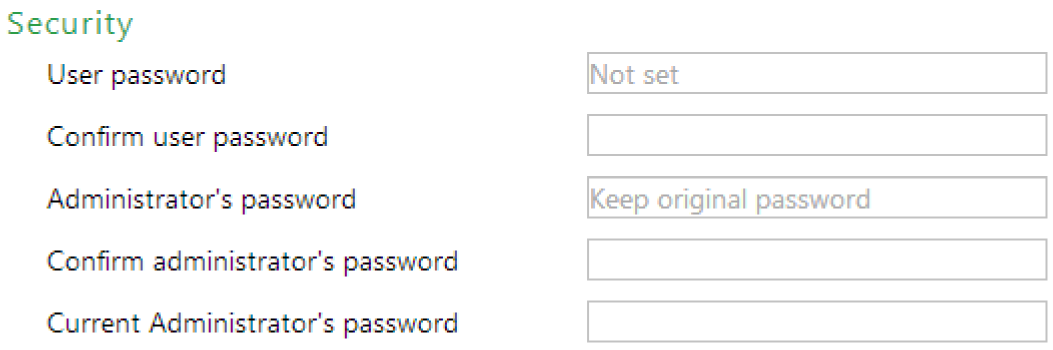
SSID	papousinetwork
Authentication method	WPA2 (AES) ▼
Pre-shared Key	Keep original password
Re-enter key	

fig. 8 - WiFi network parameters

As *Authentication method* is available these options: *Open, WEP (open), WEP (shared), WPA (TKIP), WPA (AES), WPA2 (TKIP), WPA2 (AES), WPA2 (Mixed)*.

Security

The section for setting the password of the user (can only access the main page) and the administrator (has access to both the main page and the settings).



Security

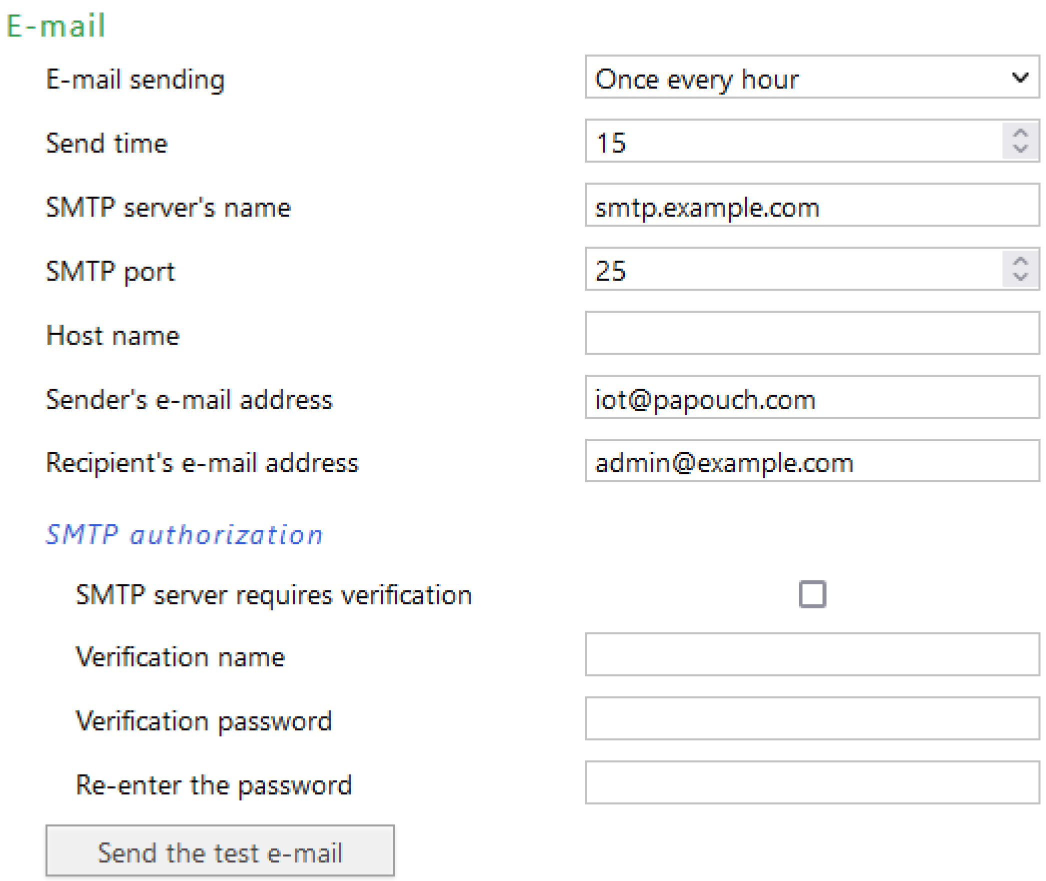
User password	Not set
Confirm user password	
Administrator's password	Keep original password
Confirm administrator's password	
Current Administrator's password	

fig. 9 - Access security settings

After saving, the passwords are no longer displayed for security reasons. The fields for entering the password show *Not set*, if the password has not been entered, or *Keep original password*, if the password has been entered but is not to be displayed.

E-mail

The device can send all counter states once an hour, once a day, once a week or a month. (The email sending feature in Papago ETH can only be used with SMTP servers that do not require SSL encrypted communication.)



E-mail

E-mail sending	Once every hour
Send time	15
SMTP server's name	smtp.example.com
SMTP port	25
Host name	
Sender's e-mail address	iot@papouch.com
Recipient's e-mail address	admin@example.com

SMTP authorization

SMTP server requires verification	□
Verification name	
Verification password	
Re-enter the password	

Fig. 10 – e-mail sending settings

E-mail is sent once an hour, once a day, once a week or a month depending on settings. Each of these can be set down to specific minute, hour, day of week or day of the month. This function serves to send energy meters states periodically.

Test e-mail example

Subject: ....Papago 5HDI 1DO ETH_info_NONAME

Body: ....TEST

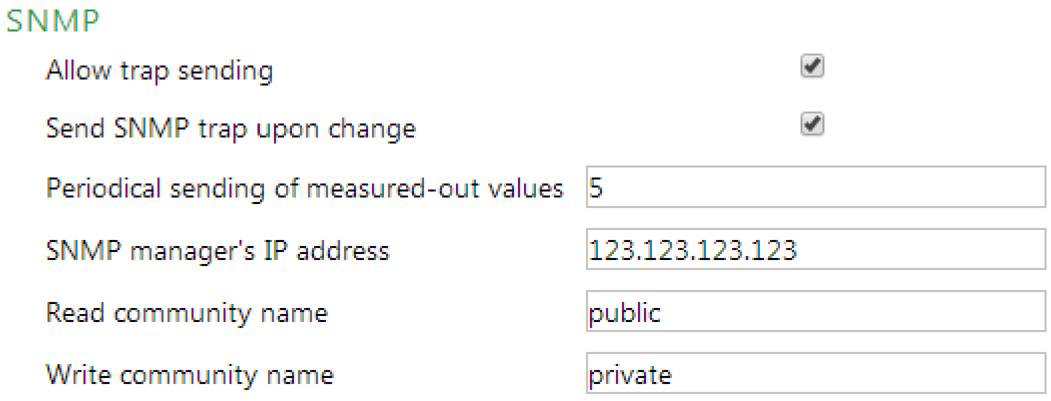
Examples of periodic e-mails

Subject: ....Papago 5HDI 1DO ETH_info_NONAME

Body: ....Input 1 is 199 °C. State: OFF
Body: ....Length is 2.1 m. State: OFF
Body: ....Electric meter 1 is 1999933392 kWh. State: ON
Body: ....Pressure is 3.656 Pa. State: OFF
Body: ....Relay state: OFF

SNMP

Here you can configure communication via SNMP used for data collection in large networks.



SNMP

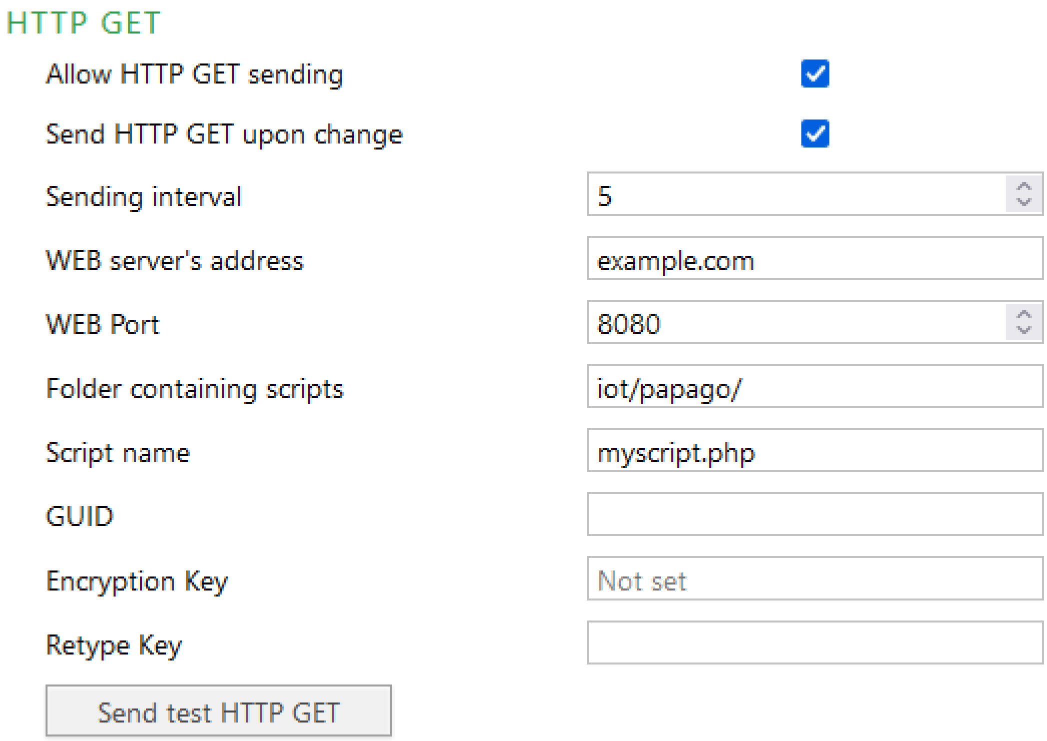
Allow trap sending	☑
Send SNMP trap upon change	☑
Periodical sending of measured-out values	5
SNMP manager's IP address	123.123.123.123
Read community name	public
Write community name	private

fig. 11 - settings for communication via SNMP

For description of SNMP objects see page 20.

HTTP GET

This section is used to set the sending of measured data to a remote server.



HTTP GET

Allow HTTP GET sending	☑
Send HTTP GET upon change	☑
Sending interval	5
WEB server's address	example.com
WEB Port	8080
Folder containing scripts	iot/papago/
Script name	myscript.php
GUID	
Encryption Key	Not set
Retype Key	

Fig. 12 - data sending via HTTP GET

If the sending interval is set to zero, the sending function is turned off. The interval can be set from 0 to 1440 minutes.

If the Send HTTP GET upon change field is checked, each valid change will result in HTTP GET being sent even outside the set interval. It should also be noted that the device may not be able to send all information about changes on the input when these are occurring fast.

If an encryption key is entered (16 bit), data inside the HTTP GET are encrypted using 128 bit AES encryption (Rijandel), CFB method.

If the message is not sent properly, it will be stored in a circle buffer with capacity of 120 messages. These will be sent once the connection is established again.

GET format

Example of periodic GET:

```
script.php?mac=0080A393A273&type=Papago 5HDI 1DO ETH&guid=PAPAGO-GUID-
TEST& description=PER&per_index=261&date_time=08/26/2015 13:12:37&
in1_name=Input 1&in1_state=0&in1_conv=199&in1_units=°C&in1_raw=199&
in2_name=Delka&in2_state=0&in2_conv=2.0&in2_units=m&in2_raw=4&
in3_name=Elektromer 1&in3_state=1&in3_conv=69&in3_units=kWh&in3_raw=69&
in4_name=Input 4&in4_state=0&in4_conv=271&in4_units=m&in4_raw=271&
in5_name=Tlakomer&in5_state=0&in5_conv=3.656&in5_units=Pa&in5_raw=3656&
out1_name=Output 1&out1_state=1
```

Example of a GET sent upon input change (same as the previous GET apart from description:

```
script.php? … &description=WATCH&index=42& …
```

Example of GET[2] after pressing the button in the settings:

```
script.php?mac=0080A393A273&type=Papago%202PT%20ETH
&guid=PAPAGO-GUID-TEST&description=TEST
```

Example of encrypted GET:

```
script.php?encrypted_data=%2C%60%32%08%25%03%44%2E%40%29%63%61%34%08
%44%62%67%CF%70%FE%D0%EA%E9%9C%C3%4C%9B%9D%E3%8B%31%18%10%E4%FB%9E%5
9%25%56%A4%60%68%1B%77%CC%EE%23%99%D1%CE%1A%AE%B5%E4%BC%D3%0C%84%9E%
7C%F4%2B%5F%B1%D4%99%C6%11%F8%75%C7%E5%27%10%93%DC%8D%43%EF%13%79%37
%F1%D2%5B%35%6B
```

The above shown encrypted part contains the following data:

```
mac=0080A394A139&type=Papago 2TH WIFI&guid=Papago-
GUID&description=TEST
```

The following parameters are sent in GET:

description........... Indicates a standard GET with acquired values (PER), GET sent upon a change on an input (WATCH) or a testing GET sent after pressing the button in WEB interface (TEST).

mac .... MAC address of the device.

type .... Type designation of the device.

guid .... Unique user-specified text string.

per_index .... Periodic message number. This number tells the user if all such messages have been delivered to the server. It ranges from 0 to 65535.

[2] It is 16 bytes of initialization vector followed by encrypted data as specified in the body of standard get. We have examples of get processing for Node.js and PHP in this article on the web (Czech section of website with comments in English): papouch.com/desifrovani-aes-v-http-getu-z-papaga-p3719/

index.................... Number of the message containing input change information. This number tells the user if all such messages have been delivered to the server. It ranges from 0 to 255.

date_time............. Date and time of recording in the format mm/dd/yyyy hh:mm:ss.

encrypted_data.... Parameter contains data from the encrypted GET. [2]

Following parameters are related to individual inputs:

inX_name .... User defined input name.

inX_state.............. Input state: Input is OFF (0) or ON (1).

inX_conv.............. Current counter value converted to real value based on an entered conversion equation.

inX_units.............. Unit.

inX_raw................ Counter value as a whole number without conversion.

Following parameters are related to individual outputs:

outX_name .... User defined output name.

outX_state .... Output state: Input is OFF (0) or ON (1).

Response to the HTTP GET

If you want to send a command to change output state within the http GET answer, or subtract a value from the counter state, the server should send the answer in XML format. Answer then should contain attributes *out1* and *cnt1* or *cnt2*, which can set the output and / or subtract a value from the counter state. (XML can only contain some of these attributes.) Values shall be sent in this format:

```
<root>
  <set valid="1" out1="1" cnt1="7" cnt2="5.5" cnt3="10" cnt5="1,256" />
</root>
```

If the HTTP GET is encrypted, the answer to it has to be encrypted as well and with a following format (the whole length must be no longer than 250 characters):

```
<root>
  <set
  encrypted_data=%DC%BD%5D%C1%DE%C4%0A%66%8B%69%0C%6D%8D%70%B9%11%EA%8C%1
  9%2A%93%F1%71%87%B7%47%94%77%C7%A2%71%D9%1A%3D%BA%21%CF%0D%D5%42%1F%01/
  >
</root>
```

Setting counters and output by an HTTP GET

Using HTTP GET, you can change counters or output states via *set.xml* script. This script accepts non-encrypted messages only. Papago understands these commands:

- **Setting counter to a value**

  set.xml?type=n&id=2&val=156

Parameter *id* is counter number, counted from 1. Parameter *val* is a new counter value. Papago expects whole or decimal number depending on the number of decimals set for this counter.

- **Subtract value from counter state**

  set.xml?type=m&id=1&val=37.2

  Parameter *id* is counter number, counted from 1. Parameter *val* is a value you want to subtract from the counter state. Papago expects whole or decimal number depending on the number of decimals set for this counter.

- **Turn output on**

  set.xml?type=s&id=1

- **Turn output off**

  set.xml?type=r&id=1

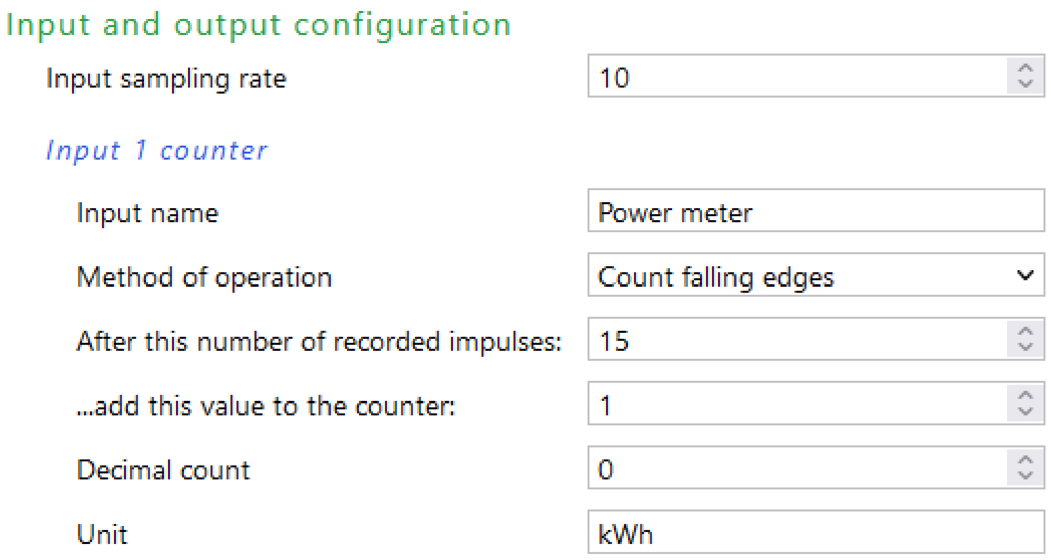
Answer to the sent GET is XML in this format:

```
<root>
  <result status="1" />
</root>
```

Should the attribute *status* be 0, it means the command was not performed because it contains errors or unexpected values.

Inputs and outputs configuration section

Inputs and outputs operation mode configuration.



Input and output configuration

Input sampling rate: 10

Input 1 counter

Input name: Power meter

Method of operation: Count falling edges

After this number of recorded impulses: 15

...add this value to the counter: 1

Decimal count: 0

Unit: kWh

Fig. 13 – inputs settings

Inputs sampling rate is common for both inputs and represents time of how long a value has to be on the input to be considered a valid one. Other settings are individual for each input.

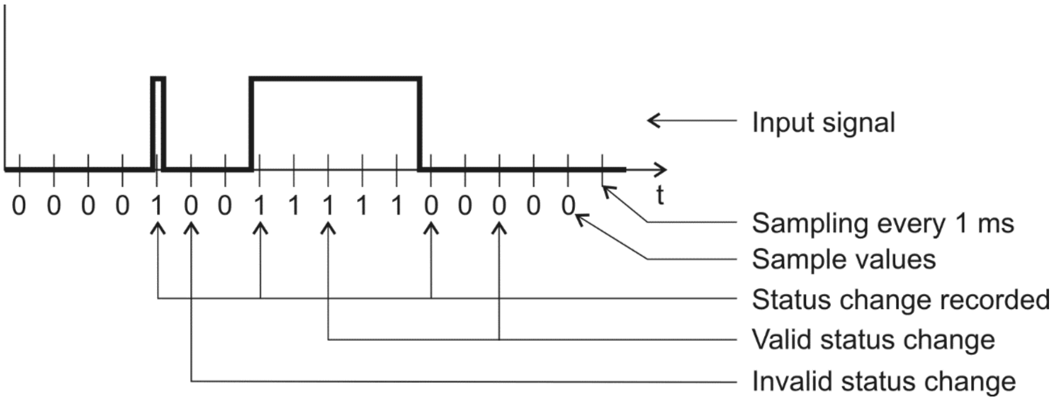
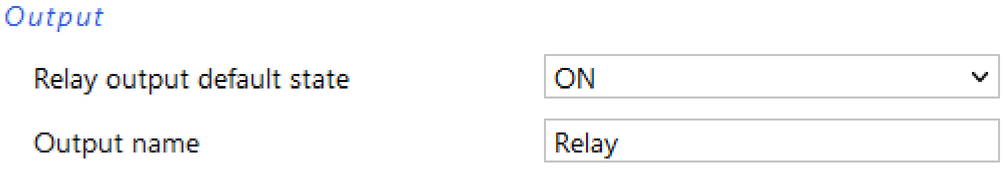


Fig. 14 – input change evaluation principal – example for 2ms setting

The input value is sampled in 1ms intervals. Input state is considered valid after several subsequent reads of the same value. The number of subsequent reads for the input state change is defined as *Input sampling rate*.

There are two more output settings at the end of this section:



Output

Relay output default state	ON
Output name	Relay

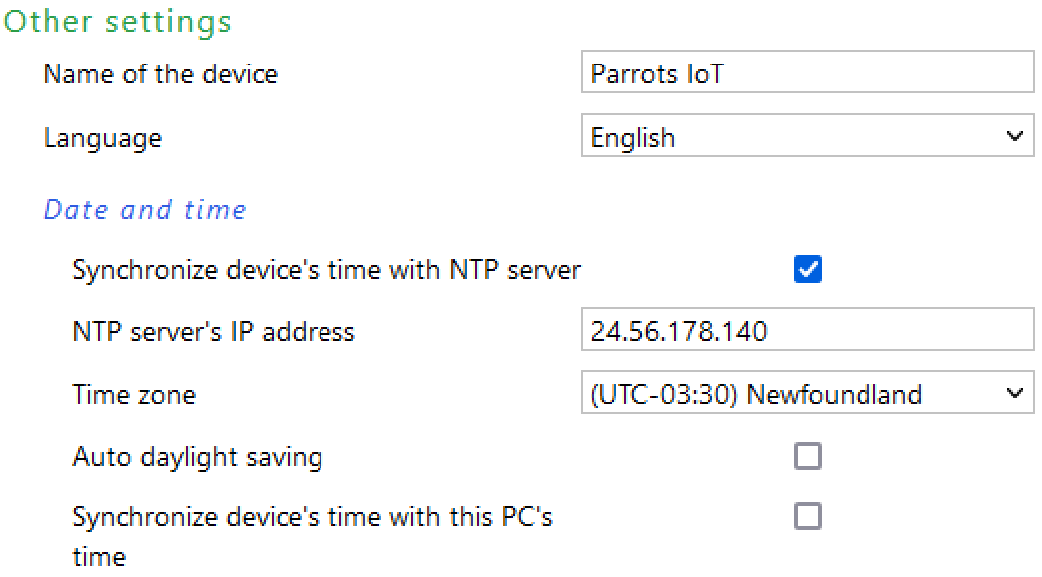
Fig. 15 – output settings

Other Settings

This section allows you to set the time, language of the website, etc…

You can choose Czech or English here.

The set name can be used to find PAPAGO in your network. Just enter the set name to the address line in your browser and you will be redirected. This does not work if the set name contains a space!



Other settings

Name of the device	Parrots IoT
Language	English

Date and time

Synchronize device's time with NTP server	☑
NTP server's IP address	24.56.178.140
Time zone	(UTC-03:30) Newfoundland
Auto daylight saving	□
Synchronize device's time with this PC's time	□

Fig. 16 – other settings

CONFIGURATION VIA TELNET PROTOCOL

Connection

IP address is not known

It is recommended that the IP address should be set using the Ethernet Configurator software (for more information see page 6).

1) Open the window of the cmd command. (In the Windows OS select Start/Run, enter `cmd` in the provided line and click Enter.)
2) Make the following entries into the ARP table:
   a. Type `arp –d` and confirm by Enter. This will delete the current ARP table.
   b. Use the following command to assign 192.168.1.254 to the module MAC address:

      `arp –s [new_ip_address] [MAC_address_of_device]`

      example: `arp -s 192.168.1.254 00-20-4a-80-65-6e`
3) Now open Telnet. (Type in `telnet` and click Enter. [3])
4) Enter `open [new_ip_address] 1` and confirm.
5) After a while, the terminal will display an error message saying that connection failed. However, this step is necessary for the module to enter the IP address into its ARP table.
6) Connect to the IP address of the module. (Type in `open [IP address in dotted format] 9999` and click Enter.)
7) So far you have only entered the configuration mode of the module. The IP address has not yet been set. It must be set in the menu Server Configuration > IP Address. If you close the configuration mode without saving the settings and IP address configuration, the whole procedure must be repeated!
8) If the entered IP address is valid, the device displays an introductory text ending with:

   `Press Enter for Setup Mode`

   Press Enter within 3 seconds, otherwise the configuration mode will close.
9) The device will display a preview of its settings.
10) The preview ends with a paragraph called “Change setup:” which lists the groups of parameters that can be configured. Network parameters can be changed in the “Server” section where you can set a new network address and other parameters.

[3] In Windows 10 or higher, Telnet client is not a standard part of system. Install it using following procedure:

a) Open the “Control Panels/Programs and Features” menu.
b) On the left, click “Enable or disable features of Windows system” (this option requires the administrator to log in).
c) The “Features of Windows system” window displays. Here tick the “Telnet service Client” field and click Ok. The client for Telnet will be installed.

IP address is known

1) In OS Windows choose Start/Run, enter `telnet` in the provided line and press Enter.[3]
2) Connect to the IP address of the module. (Type in `open [IP address in dotted format] 9999` and press Enter.)
3) If the entered IP address is valid, the device displays an introductory text ending with:

   ```
   Press Enter for Setup Mode
   ```

   Press Enter within 3 seconds, otherwise the configuration mode will close.
4) The device will display a preview of its settings.
5) The preview ends with a paragraph called "Change setup:" which lists the groups of parameters that can be configured. Network parameters can be changed in the "Server" section.

Telnet main menu

Individual items can be chosen using the numbers written next to them. Choose the required number and press Enter.

The menu structure is as follows:

```
Change Setup:
  0 Server
    ...
  7 Defaults
  8 Exit without save
  9 Save and exit            Your choice ?
```

Server

Basic Ethernet settings.

This section contains the following parameters:

```
IP Address: (192) .(168) .(001) .(122)
Set Gateway IP Address (N) ?
Netmask: Number of Bits for Host Part (0=default) (16)
Change telnet config password (N) ?
```

IP Address

(IP address)

IP address of the module. The digits must be entered one by one and separated by Enter.

Default value: 192.168.1.254

Set Gateway IP Address

(set the IP address of the gateway)

Gateway IP addr

(IP address of the gateway)

In “Set Gateway IP Address” enter “Y“ to change the IP address. The system then prompts you to change the Gateway IP address. The digits must be entered one by one and separated by Enter.

Netmask

(network mask)

Here you specify the number of bits of the IP address that make up the network part.

The Netmask is set as a number of bits determining the range of available IP addresses of the local network. If, for example, value `2` is entered, the structure of the Netmask is `255.255.255.252` . The entered value specifies the number of bits from the right. The maximum is 32.

Default value: 8

Example:
The mask 255.255.255.0 (binary form: 11111111 11111111 11111111 00000000) =. number 8.
The mask 255.255.255.252 (binary form: 11111111 11111111 11111111 11111100) = number 2.

Change telnet config password

(Set the password for Telnet)

Enter new Password

(Enter the password for Telnet)

This parameter is used to set a new password which is required prior to any configuration via Telnet or via WEB interface (admin password).

For item “Change telnet config password”, enter “Y” to change the password. The system then prompts you to change the password.

Factory Defaults

By pressing number 7 the device restores the default settings.

The default setting means that all parameters will return to their initial factory settings. The IP address remains unchanged; the web interface port is set to 80.

Exit without save

To close the configuration mode without saving the changed parameters.

Save and exit

This option saves the changes. If any parameter has been changed, the device is restarted. The restart takes several tens of seconds.

XML

It is possible to obtain the last measured values, limits (thresholds) and device name from the device in the form of a text file in the XML format. The file is available at *http://[IP-adresa]/fresh.xml* – i.e. for example at http://192.168.1.254/fresh.xml for the default settings.

```
<root>
  <din id="1" name="Power meter" bin="0" val="1100 kWh" raw="1100000"/>
  <din id="2" name="Sauna" bin="1" val="1689 kWh" raw="1689"/>
  <din id="3" name="Input 3" bin="0" val="999 Pa" raw="999999"/>
  <din id="4" name="Input 4" bin="1" val="1234.45" raw="123445"/>
  <din id="5" name="Input 5" bin="0" val="0" raw="0"/>
  <dout id="1" name="Relay" bin="1"/>
  <status location="NONAME" time="08/31/2015 12:57:38"/>
</root>
```

Fig. 17 – example of XML with actual values

The XML contains a *din* tag for each input, one *dout* tag for output and *status* tag:

status

location

User-defined name of the device.

time

The current system time of the device in the format *mm/dd/yyyy hh:mm:ss*.

din

id

Input number. (First number is 1.)

name

User defined input name.

bin

Number 0 or 1 based on the input status - off (0) or on (1).

val

Calculated counter value as an integer or a decimal number including units (when entered).

raw

Current counter status without calculations.

dout

id

Output number. (First number is 1.)

name

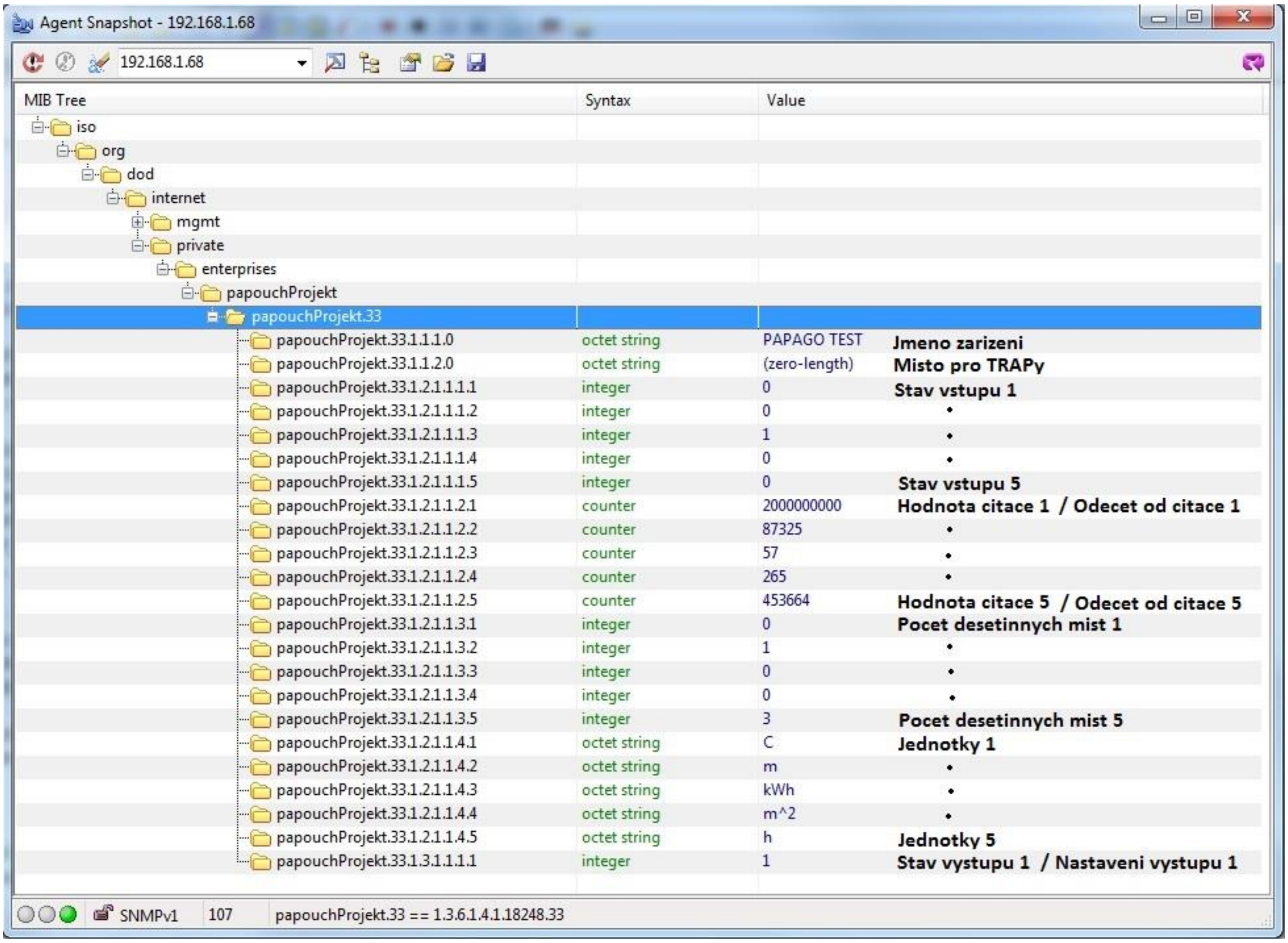
User defined output name.

bin

Number 0 or 1 based on the output status - off (0) or on (1).

SNMP

The SNMP protocol (version 1) contains objects with individual values. For a detailed description of the objects see below. The MIB table you can import into your SNMP manager can be downloaded from papouch.com.



Agent Snapshot - 192.168.1.68

192.168.1.68

MIB Tree	Syntax	Value	
iso			
org			
dod			
internet			
mgmt			
private			
enterprises			
papouchProjekt			
papouchProjekt.33			
papouchProjekt.33.1.1.1.0	octet string	PAPAGO TEST	Jmeno zarizeni
papouchProjekt.33.1.1.2.0	octet string	(zero-length)	Misto pro TRAPy
papouchProjekt.33.1.2.1.1.1.1	integer	0	Stav vstupu 1
papouchProjekt.33.1.2.1.1.1.2	integer	0	•
papouchProjekt.33.1.2.1.1.1.3	integer	1	•
papouchProjekt.33.1.2.1.1.1.4	integer	0	•
papouchProjekt.33.1.2.1.1.1.5	integer	0	Stav vstupu 5
papouchProjekt.33.1.2.1.1.2.1	counter	2000000000	Hodnota citace 1 / Odecet od citace 1
papouchProjekt.33.1.2.1.1.2.2	counter	87325	•
papouchProjekt.33.1.2.1.1.2.3	counter	57	•
papouchProjekt.33.1.2.1.1.2.4	counter	265	•
papouchProjekt.33.1.2.1.1.2.5	counter	453664	Hodnota citace 5 / Odecet od citace 5
papouchProjekt.33.1.2.1.1.3.1	integer	0	Pocet desetinnych mist 1
papouchProjekt.33.1.2.1.1.3.2	integer	1	•
papouchProjekt.33.1.2.1.1.3.3	integer	0	•
papouchProjekt.33.1.2.1.1.3.4	integer	0	•
papouchProjekt.33.1.2.1.1.3.5	integer	3	Pocet desetinnych mist 5
papouchProjekt.33.1.2.1.1.4.1	octet string	C	Jednotky 1
papouchProjekt.33.1.2.1.1.4.2	octet string	m	•
papouchProjekt.33.1.2.1.1.4.3	octet string	kWh	•
papouchProjekt.33.1.2.1.1.4.4	octet string	m^2	•
papouchProjekt.33.1.2.1.1.4.5	octet string	h	Jednotky 5
papouchProjekt.33.1.3.1.1.1.1	integer	1	Stav vystupu 1 / Nastaveni vystupu 1

SNMPv1 107 papouchProjekt.33 == 1.3.6.1.4.1.18248.33

Fig. 18 – meaning of Papago objects

Objects – variables

Input state

Name: inState

Object ID: 1.3.6.1.4.1.18248.33.1.2.1.1.1.1 to 5

Description: Input state as a number 0 (off) or 1 (on).

Counter value or readout

Name: inCounter

Object ID: 1.3.6.1.4.1.18248.33.1.2.1.1.2.1 to 5

Description: In case of read you can read the value as a whole number. In case of write the number will be subtracted from the current value.

Number of decimals

Name: inDecNum

Object ID: 1.3.6.1.4.1.18248.33.1.2.1.1.3.1 to 5

Description: Number of decimals to apply to the previous value inCounter, to get the real counter value after the calculation.

Unit

Name: inUnit

Object ID: 1.3.6.1.4.1.18248.33.1.2.1.1.4.1 to 5

Description: String with user defined unit.

Output state

Name: outState

Object ID: 1.3.6.1.4.1.18248.33.1.3.1.1.1.1

Description: Output state as a number 0 (off) or 1 (on).

SNMP objects – general

The following two objects relate to the entire device.

Device name

Name: deviceName

Object ID: 1.3.6.1.4.1.18248.31.1.1.1.0

Description: User-defined device name.

Alarm text

Name: psAlarmString

Object ID: 1.3.6.1.4.1.18248.31.1.1.2.0

Description: Text of the alarm message sent when input state changes. (Texts are the same as in e-mail.)

Traps

Trap 1 – Current measured values

The trap contains all current values as well as the name of the device set by the user.

The trap is sent only if a non-zero frequency of sending has been set.

MODBUS TCP

Outputs

Read outputs state

To access these values use *Read Coils* function.

Address	*Access*	*Function*	*Description*
0	read	0x01	**Output 1 state** 0 = output is off 1 = output is on

Set outputs state

To access these values use *Write Single Coil* or *Write Multiple Coils* function.

Address	*Access*	*Function*	*Description*
0	write	0x05 0x0F	**Output 1 state** 0 = output is off 1 = output is on

Read inputs states

To access these values use *Read Discrete Inputs* function.

Address	*Access*	*Function*	*Description*
0 – 4	read	0x02	**Inputs 1 to 5 states** 0 = output is off 1 = output is on

Counters

Read counters state

To access these values use *Read Holding Register* function.

Address	*Access*	*Function*	*Description*
Counter 1			
0	read	0x03	**Function** Counter mode as one of these codes: 0 = unused counter (set to Off in configuration) 1 = counting trailing edges 2 = counting leading edges 3 = counting both edges
1, 2	read	0x03	**Date and time** Date and timer of the device in the NTP format.
3, 4	read	0x03	**Counter value as an integer** Counter value as an integer. Number of decimals to calculate real values is in the following register.

Address	*Access*	*Function*	*Description*
5	read	0x03	**Number of decimals** Number of decimals. This number of decimas is to be applied to the value in previous register. This way the real value as a decimal number is calculated.
6, 7	read	0x03	**Counter value as a decimal number** Counter value as a decimal number (32 bit float according to IEEE 754).
Counter 2			
od 100	Counter values 2.		
Counter 3			
od 200	Counter values 3.		
Counter 4			
od 300	Counter values 4.		
Counter 5			
od 400	Counter values 5.		

Set counter states

To access these values use *Write Multiple Registers* function.

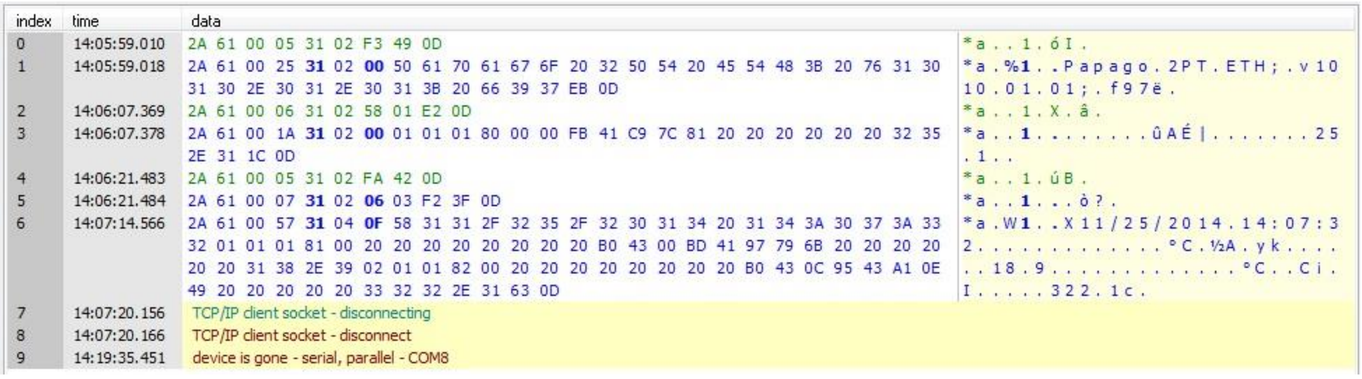
Address	*Access*	*Function*	*Description*
Counter 1			
3, 4	write	0x10	**Counter value as an integer** Enter counter value as an integer. Number of decimal places is taken from settings on the WEB interface.
6, 7	write	0x10	**Counter value as a decimal number** Enter counter value as a decimal number (32 bit float according to IEEE 754).
8, 9	write	0x10	**Subtract value – entered as an integer** Enter counter value as an integer. This number will be subtracted from the current counter value.[4] Number of decimal places is taken from settings on the WEB interface.
10, 11	write	0x10	**Subtract value – entered as a decimal number** Enter counter value as a decimal number (32 bit float according to IEEE 754). This number will be subtracted from the current counter value.[4]
Counter 2			
od 103	Counter values 2.		

[4] If such a number is entered that the subtraction would result in negative number, function will be cancelled and exception code 4 will be sent.

Address	*Access*	*Function*	*Description*
Counter 3			
od 203	Counter values 3.		
Counter 4			
od 303	Counter values 4.		
Counter 5			
od 403	Counter values 5.		

SPINEL

The device contains the standard Spinel protocol (format 97) for communication via the TCP data channel. Application development with this protocol is easy due to Spinel Terminal, Spinel.NET SDK on Github and Spinel online parser.



```
index time          data
0     14:05:59.010  2A 61 00 05 31 02 F3 49 0D                                              *a..1.óI.
1     14:05:59.018  2A 61 00 25 31 02 00 50 61 70 61 67 6F 20 32 50 54 20 45 54 48 3B 20 76 31 30  *a.%1..Papago.2PT.ETH;.v10
                    31 30 2E 30 31 2E 30 31 3B 20 66 39 37 EB 0D                            10.01.01;.f97ë.
2     14:06:07.369  2A 61 00 06 31 02 58 01 E2 0D                                           *a..1.X.â.
3     14:06:07.378  2A 61 00 1A 31 02 00 01 01 01 80 00 00 FB 41 C9 7C 81 20 20 20 20 20 20 32 35  *a..1........ûAÉ|.......25
                    2E 31 1C 0D                                                             .1..
4     14:06:21.483  2A 61 00 05 31 02 FA 42 0D                                              *a..1.úB.
5     14:06:21.484  2A 61 00 07 31 02 06 03 F2 3F 0D                                        *a..1...ò?.
6     14:07:14.566  2A 61 00 57 31 04 0F 58 31 31 2F 32 35 2F 32 30 31 34 20 31 34 3A 30 37 3A 33  *a.W1..X11/25/2014.14:07:3
                    32 01 01 01 81 00 20 20 20 20 20 20 20 B0 43 00 BD 41 97 79 6B 20 20 20 20  2............°C.½A.yk....
                    20 20 31 38 2E 39 02 01 01 82 00 20 20 20 20 20 20 20 B0 43 0C 95 43 A1 0E  ..18.9............°C..Ci.
                    49 20 20 20 20 20 33 32 32 2E 31 63 0D                                  I.....322.1c.
7     14:07:20.156  TCP/IP client socket - disconnecting
8     14:07:20.166  TCP/IP client socket - disconnect
9     14:19:35.451  device is gone - serial, parallel - COM8
```

Fig. 19 – example of device communbication in Spinel terminal

Overview of implemented instructions:

Output state control

Allows to control output relays.

Request:

Instruction code: 20H

Parameters: (state)

state	Output number and state	length: 1 byte
Enter 81H to turn on or 80H to turn off the output.		

Response:

Acknowledge code: ACK 00H

Examples:

Request:
`2AH,61H,00H,06H,FEH,02H,20H,81H,CDH,0DH`
Response:
`2AH,61H,00H,05H,31H,31H,00H,0DH,0DH`

Read output state

Read output relay state.

Request:

Instruction code: 30H

Response:

Acknowledge code: ACK 00H

Parameters: (state)

state	Output number and state	length: 1 byte

Output is on (01H) or off (00H).

Examples:

Request:
`2AH,61H,00H,05H,FEH,02H,30H,3FH,0DH`
Response:
`2AH,61H,00H,06H,31H,02H,00H,01H,3AH,0DH`

Read input states

Reads input state.

Request:

Instruction code: 31H

Response:

Acknowledge code: ACK 00H

Parameters: (state)

state	Input number and state	length: 1 byte
Input state as a bit-oriented byte. Individual bits have these meanings: *(MSb) xxx54321 (LSb)*		

Examples:

Request:
`2AH,61H,00H,05H,FEH,02H,31H,3EH,0DH`
Response:
`2AH,61H,00H,06H,31H,02H,00H,01H,3AH,0DH`

Read counters

Instruction reads one or more counters

Request:

Instruction code: 60H

Parameters: (counter)

counter	Counter number	length: 1 byte
Number of counter or counters to read. Enter 00H (all counters) or number from 01H to 05H.		

Response:

Acknowledge code: ACK 00H

Parameters: {[channel][value][status][int][float][str][unit][decimals][rawint][rawstr]} {…}

channel Input number	id: 00H length: 1 byte
Input number ranging from 1 to 5.	

value	id: 01H

Current input state	length: 1 byte
Current input state as a value 00H (OFF) or 01H (ON).	

status Counter mode	id: 02H length: 1 byte
Can contain these codes for counter modes: 00H … no added actions	

int Counter value as an integer	id: 03H length: 4 bytes
Counter value as an integer. (Real value can be calculated by dividing this value based on number of decimals. Number of decimals is in *decimals* parameter)	

float Counter value as a decimal number	id: 04H length: 4 bytes
Counter value as a decimal number after calculation (32 bit float according to IEEE 754).	

str Counter value as a string	id: 05H length: 10 bytes
Counter value as a string. A dot is used as a decimal separator. String is aligned to the right.	

unit Unit	id: 06H length: 10 bytes
User defined unit. String is aligned to the right.	

decimals Number of decimals	id: 07H length: 1 byte
Number of decimals to calculate the real value.	

rawint Raw value as a decimal	id: 08H length: 4 bytes
Raw counter value without calculation as an integer. (Real value can be calculated by dividing this value based on number of decimals. Number of decimals is in *decimals* parameter)	

rawstr Raw value as a string	id: 09H length: 10 bytes
Raw counter value as a string. A dot is used as a decimal separator. String is aligned to the right.	

Examples:

Request – read counter 1:
`2AH,61H,00H,06H,FEH,01H,60H,01H,0EH,0DH`
Response:

2AH,61H,00H,3DH,31H,01H,00H,	
00H,01H,	- counter number: 0
01H,00H,	- input state: 0
02H,00H,	- counter state:
03H,00H,00H,00H,D2H,	- counter value as an integer
04H,43H,52H,00H,00H,	- counter value as a decimal number
05H,20H,20H,20H,20H,20H,20H,20H,32H,31H,30H,	- counter value as a string
06H,20H,20H,20H,20H,20H,20H,20H,C2H,B0H,43H,	- unit as a string
07H,00H,	- number of decimals
08H,00H,00H,00H,D2H,	- raw value as an integer
09H,20H,20H,20H,20H,20H,20H,20H,32H,31H,30H,	- raw value as a string
23H,0DH	

Read name and version

Reads device name, internal firmware version and a list of possible communication formats. This is set during manufacturing.

Request:

Instruction code: F3H

Response:

Acknowledge code: ACK 00H

Parameters: (string)

string	Name and version	length: 1 byte
Papago 2PT ETH; v1010.01.01; f97 String can also contain other information separated by semicolon, space and a small letter determining the information that is following.		

Examples:

Request:
2AH,61H,00H,05H,31H,02H,F3H,49H,0DH
Response:
2AH,61H,00H,25H,31H,02H,00H,50H,61H,70H,61H,67H,6FH,20H,32H,50H,54H,20H,45H,54H,48H,3BH,20H,76H,31H,30H,31H,30H,2EH,30H,31H,2EH,30H,31H,3BH,20H,66H,39H,37H,EBH,0DH,

Read device info

Instruction reads device info.

Request:

Instruction code: FAH

Response:

Acknowledge code: ACK 00H

Parameters: (product_number)(serial_number)(other)

product_number	length: 2 bytes
Product number. It is 277 for device with number 0227.00.03/0001.	

serial_number	length: 2 bytes
Unique serial number. It is 1 for device with number 0227.00.03/0001.	

other	length: 4 bytes
Other device information.	

Examples:

Request:
`2AH,61H,00H,05H,FEH,02H,FAH,75H,0DH`

Automatic message upon input change

This message is generated upon a change on any of inputs.

Acknowledge code: ACK 0DH

Parameters: {[channel][value][status][int][float][str][unit][decimals][rawint][rawstr]} {…}

channel Input number	id: 00H length: 1 byte
Input number from 1 to 5.	

value Current input state	id: 01H length: 1 byte
Current input state as a value 00H (OFF) or 01H (ON).	

status Counter mode	id: 02H length: 1 byte
Can contain these codes for counter modes: 00H … no added actions	

int Counter value as an integer	id: 03H length: 4 bytes
Counter value as an integer. (Real value can be calculated by dividing this value based on number of decimals. Number of decimals is in *decimals* parameter)	

float Counter value as a decimal number	id: 04H length: 4 bytes
Counter value as a decimal number after calculation (32 bit float according to IEEE 754).	

str Counter value as a string	id: 05H length: 10 bytes
Counter value as a string. A dot is used as a decimal separator. String is aligned to the right.	

unit Unit	id: 06H length: 10 bytes
User defined unit. String is aligned to the right.	

decimals Number of decimals	id: 07H length: 1 byte
Number of decimals to calculate the real value.	

rawint Raw value as a decimal	id: 08H length: 4 bytes
Raw counter value without calculation as an integer. (Real value can be calculated by dividing this value based on number of decimals. Number of decimals is in *decimals* parameter)	

rawstr Raw value as a string	id: 09H length: 10 bytes
Raw counter value as a string. A dot is used as a decimal separator. String is aligned to the right.	

Example:

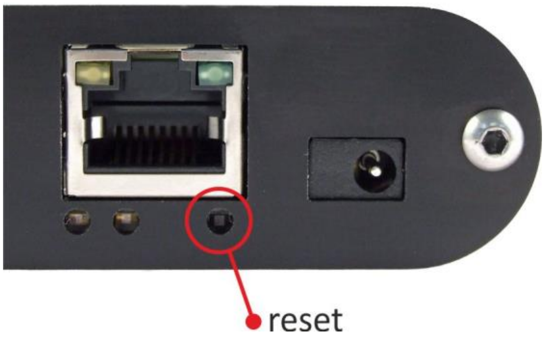
Automatic message:

```
2AH,61H,01H,1DH,31H,17H,0DH,

00H,01H,                                          Counter 1
01H,01H,                                            input ON
02H,00H,                                            No action
03H,00H,00H,00H,CCH,                                Integer
04H,43H,4CH,00H,00H,                                Decimal number
05H,20H,20H,20H,20H,20H,20H,20H,32H,30H,34H,        String
06H,20H,20H,20H,20H,20H,20H,20H,C2H,B0H,43H,        Unit
07H,00H,                                            Number of decimals
08H,00H,00H,00H,CCH,                                Raw value – integer
09H,20H,20H,20H,20H,20H,20H,20H,32H,30H,34H,        Raw value – decimal number

00H,02H,                                          Counter 2
01H,00H,
02H,00H,
03H,00H,00H,00H,23H,
04H,40H,60H,00H,00H,
05H,20H,20H,20H,20H,20H,20H,20H,33H,2EH,35H,
06H,20H,20H,20H,20H,20H,20H,20H,20H,20H,6DH,
07H,01H,
08H,00H,00H,00H,07H,
09H,20H,20H,20H,20H,20H,20H,20H,20H,20H,37H,

00H,03H,                                          Counter 3
01H,00H,
02H,00H,
03H,00H,00H,00H,4EH,
04H,42H,9CH,00H,00H,
05H,20H,20H,20H,20H,20H,20H,20H,20H,37H,38H,
06H,20H,20H,20H,20H,20H,20H,20H,6BH,57H,68H,
07H,00H,
08H,00H,00H,00H,4EH,
09H,20H,20H,20H,20H,20H,20H,20H,20H,37H,38H,

00H,04H,                                       Counter 4
01H,00H,
02H,00H,
03H,00H,00H,00H,22H,
04H,42H,08H,00H,00H,
05H,20H,20H,20H,20H,20H,20H,20H,20H,33H,34H,
06H,20H,20H,20H,20H,20H,20H,20H,20H,20H,6DH,
07H,00H,
08H,00H,00H,00H,22H,
09H,20H,20H,20H,20H,20H,20H,20H,20H,33H,34H,

00H,05H,                                       Counter 5
01H,00H,
02H,00H,
03H,00H,00H,0EH,49H,
04H,40H,6AH,0CH,4AH,
05H,20H,20H,20H,20H,20H,33H,2EH,36H,35H,37H,
06H,20H,20H,20H,20H,20H,20H,20H,20H,50H,61H,
07H,03H,
08H,00H,00H,0EH,49H,
09H,20H,20H,20H,20H,20H,20H,33H,36H,35H,37H,

5EH,0DH
```

RESET

Follow the instructions below to restore the default configuration set by the manufacturer. Unlike when the reset is performed via the web interface or using the Telnet protocol (see page 19), the IP address is also reset to the default value of 192.168.1.254.

1) Disconnect the device from the power supply.
2) Press the button located in a small hole on the right side under the Ethernet connector resp. WiFi antenna.
3) Turn on the power and wait for about 10 seconds until the yellow light below the Ethernet connector (resp. WiFi antenna) flashes 4 times.
4) Release the button.



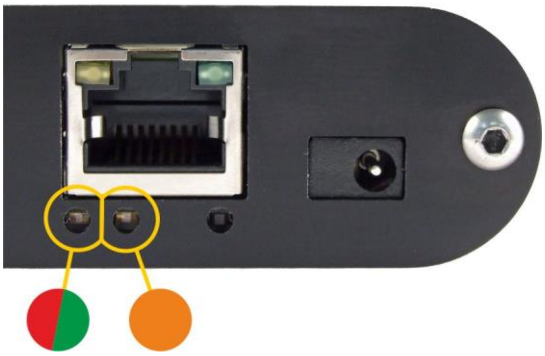
INDICATIONS

Two LEDs integrated in the Ethernet connector:

Yellow – LINK: is lit when the device is connected by cable to a switch or PC.

Green – ACT: indicates communication over the Ethernet.

Two LEDs to the left under the Ethernet connector:



Yellow (right): is lit when the connection is established via Spinel or Modbus.

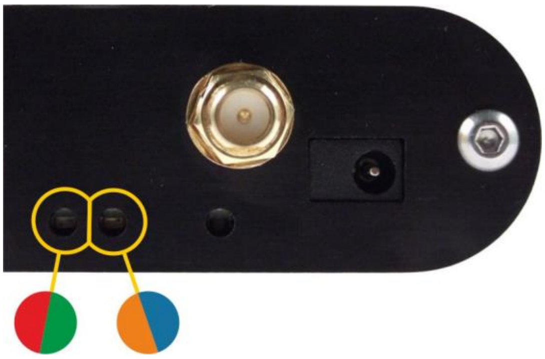
Red-green (left):

- Green: Indication of correct initialization.
- Red: Flashes briefly when switched on. If it lights up briefly afterwards, this indicates an internal device error.

Papago with WiFi connection:

Yellow-blue (right):

- Yellow lights up if Spinel or ModBus connection is established.
- Blue lights up when the Papago is connected to a WiFi network.



Red-Green (left):

- Green: Indication of correct initialization.
- Red: Flashes briefly when switched on. If it lights up briefly afterwards, this indicates an internal device error.

Input and output indicators:

There is a red indicator above each input or output terminal indicating that a given input / output is ON.

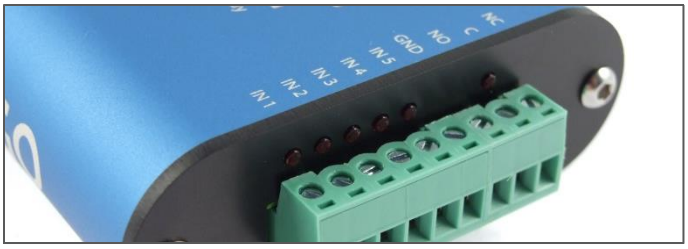


Fig. 20 – row of indicators above terminals

TECHNICAL PARAMETERS

Inputs

Type	dry contact or S0 (electricity meters)
Number of inputs	5
Current through ON input	13 mA
Input working voltage	24 V
Maximum sampling frequency	1 kHz
Connector	slip-on terminal

Output

Type	SPDT relay
Maximum contact voltage AC	50 V
Maximum contact voltage DC	85 V
Maximum load current proud	2 A
Maximum load (resistive)	62,5 VA / 60 W
Protection varistor	U_{AC} = 60 V; E_{MAX} = 5 J; C = 0,64 nF
Connector	slip-on terminal

Ethernet interface

Ethernet standard	TBase 10/100 Ethernet
Connector	RJ45
HTTP GET encryption	128 bit AES; Rijndael; CFB method

Wi-Fi interface

Specification	IEEE 802.11 b/g a IEEE 802.11n (single stream), IEEE 802.11 d/h/i/j/k/w/r
Frequency	2,4 GHz
Antenna connector	SMA RP

Clock circuit and internal memory

Clock backup method (RTC)	capacitor (not replaceable by the user)
RTC backup time after power outage	5 days (if the device was previously connected to a power source for at least three hours without interruption)

Device electronics

PoE power supply	according to IEEE 802.3af
Power supply from an external source	11 to 58 V DC (with reverse polarity protection)
Current consumption from ext. source at 15 V	typically 120 mA; *Wi-Fi version: 31mA*
Current consumption from ext. source at 24 V	typically 72 mA; *Wi-Fi version: 31mA*
Current consumption from PoE	typically 32 mA

Consumption	typically 1.8 W
Power supply connector	coaxial 3.8 × 1.3 mm; + inside
Operating temperature range	-20 to +70 °C
Dimensions (without connectors)	88 × 70 × 25 mm
Housing material	anodized aluminum
Degree of protection	IP 30

Other parameters

Weight	typically 130 g

Default settings of the Ethernet

IP address	192.168.1.254
Netmask	255.255.255.0 (8 bits; mask C)
IP address of the gateway	0.0.0.0

Available designs

Mountable on 35 mm DIN rail	optional accessory
Wall-mountable	optional accessory

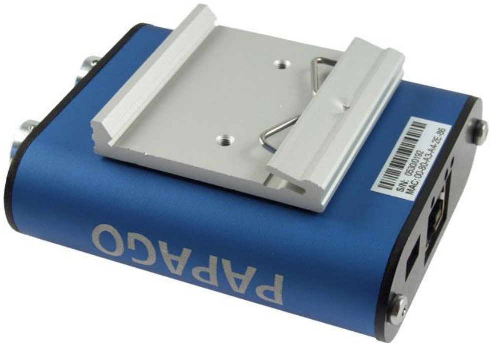


fig. 21 – Papago with DIN rail holder

Do not hesitate to contact us if you have any other requirements concerning the design and functions of PAPAGO 5HDI DO.

Papouch s.r.o.

Data transmission in industry, line and protocol conversions, RS232/485/422/USB/Ethernet/GPRS/ WiFi, measurement modules, intelligent temperature sensors, I/O modules, and custom-made electronic applications.

Address:

Strasnicka 3164
102 00 Prague 10
Czech Republic

Tel:

+420 267 314 267

Internet:

www.papouch.com

E-mail:

info@papouch.com