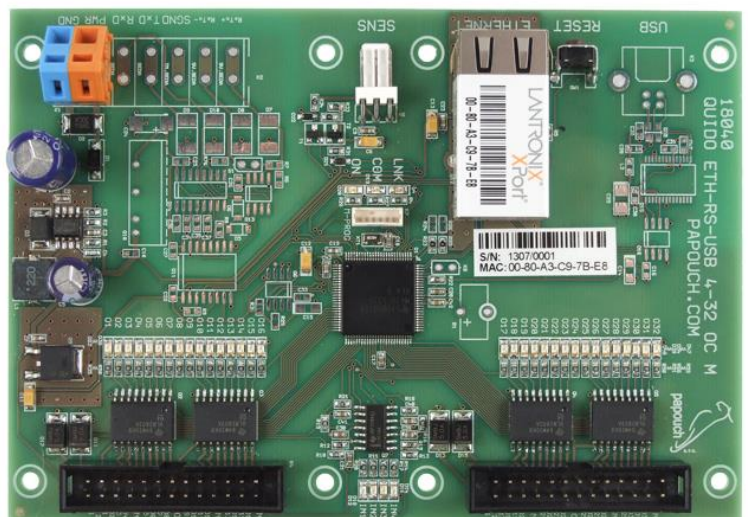
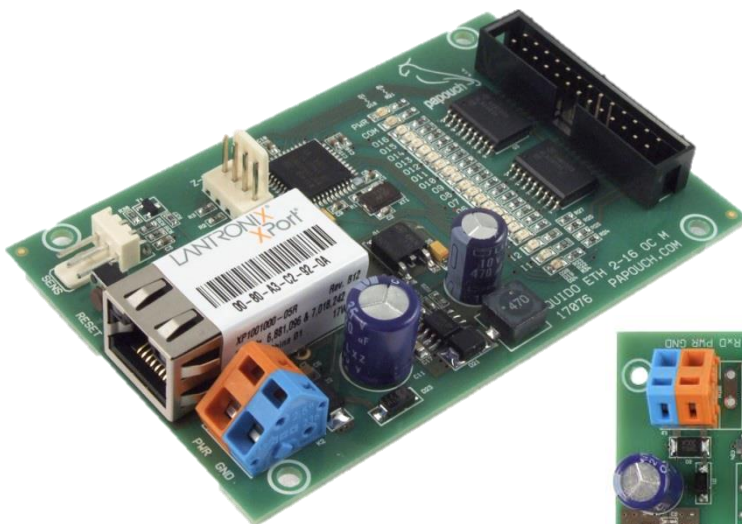

Quido ETH OC

2 or 4 inputs for contact

16 or 32 digital outputs

1 thermometer -55 to +125 °C

Communication via Ethernet



Quido ETH OC

Datasheet

Created: 14.3.2018

Latest update: 27.6 2022 15:34

Number of pages: 58

© 2022 Papouch s.r.o.

Papouch s.r.o.

Address:

**Strasnicka 3164/1a
Prague 10**

Phone:

+420 267 314 267

Internet:

www.papouch.com

E-mail:

papouch@papouch.com



TABLE OF CONTENTS

Description.....	6
Features	6
Basic functions.....	6
Application	6
Quido I/O module control options.....	7
Block diagram	7
Available designs	8
Available accessories	10
First connection	11
Wiring inputs and outputs.....	12
Inputs.....	13
Outputs	13
Configuration and Control	15
Web interface – Control	15
Network.....	17
Device mode.....	17
Remote device for TCP connection.....	18
Reset the device	18
Security	19
E-mail.....	20
SMTP authorization	20
SNMP.....	21
Sending	22
Inputs	23
Outputs	25
Thermometers.....	27
Other	28
Info.....	29
Wix software	30
Spinel protocol	30
MODBUS TCP protocol	31
Telnet Configuration.....	31
Connection	31
Unknown IP address.....	31
IP address is known.....	32
Telnet main menu.....	32
Server	32
Factory Defaults	33
Exit without save	33

Save and exit.....	33
XML file.....	34
Inputs: din	34
Outputs: dout	35
Thermometers: temp.....	35
status	36
HTTP GET	37
Sending the current status to a remote server	37
Sending method.....	37
Request parameters	38
Setting the Quido in the GET response.....	38
Controlling Quido by HTTP GET.....	39
HTTP GET response	40
Sending emails	42
Input or output change message.....	42
Temperature limit exceeded message	42
Sensor error message.....	42
Test message to verify settings.....	42
SNMP	43
Inputs.....	43
Outputs	44
Temperature monitoring.....	45
Temperature measurement.....	46
Additional features	47
Traps	48
Temperature monitoring modes	49
Mode 1.....	49
Mode 2.....	49
Mode 3.....	50
Mode 4.....	50
Mode 5.....	51
Mode 6.....	51
FAQ	52
What do I need to set up for Quido to work on my network?	52
How to find out the IP address of Quido?	52
Reset	52
Indications.....	53
Technical parameters.....	54
Summary	54
The principle of evaluating changes to inputs.....	56

DESCRIPTION

Quido ETH X/Y OC are I/O modules with digital inputs and outputs and a thermometer (X represents the number of inputs, Y the number of outputs). The inputs are designed for contact, the digital outputs are open collector type. The thermometer can be connected on a cable length of up to 15 meters and is capable of measuring temperatures from -55 to +125 °C. The Quido connects directly to a computer network (Ethernet) and can be completely controlled and managed via a web interface.

Features

- Digital inputs for contact.
- Digital outputs open collector type (NPN NO).
- Input for thermometer with measuring range -55 to +125 °C.
- Connection directly to the computer network LAN (Ethernet).
- Indication of switching on, communication and status of inputs and outputs with indicator lights.
- Power supply from an external 8 to 30 V supply.
- Control:
 - Standard protocols: Modbus TCP¹, SNMP, HTTP GET, Spinel¹
 - Through the web interface.
 - For all modern web browsers including iPhone and Android OS.
 - Secure access separately for viewing and configuration.
 - Optional status symbols for inputs and outputs.
 - Wix software.

Basic functions

- Reading the current status of inputs
- Automatic sending of information (email, HTTP GET, ...) about changes on inputs
- Input pulse counting or input state change counting
- Output control
- Setting outputs for a defined time
- Thermometer: temperature measurement -55 to +125 °C
- Temperature monitoring function (switching on the output at different temperatures, also switching on for a set time when the set temperature is reached, sending an email or HTTP GET)

Application

- Intelligent buildings, home automation.
- Reading the status of sensors or transducers from the PC.

¹ The documentation of Spinel and MODBUS TCP protocols is available in separate documents. (Downloadable from papouch.com.) The Spinel protocol comes with a free software – Spinel Terminal – for easy tuning of applications. Spinel.NET (C#) library is also available [on GitHub](#) (Czech documentation).

- Computer monitoring of safety sensors and control of various devices.
- Monitoring the status of door contacts and remote opening of entrance doors.
- Counting the number of passes.
- Small automation of objects and production processes.
- Autonomous control or remote control based on measured temperature.
- Integration of many inputs and outputs in different locations

Quido I/O module control options

(For more information on the following options, see page 15.)

- Through the web interface.
- The standard industrial protocol MODBUS TCP.²
- SNMP network protocol.
- Simple control and sending of current statuses by HTTP GET requests.
- Access to all information from Quido via XML.
- Via Spinel – a serial protocol used for communication by all devices provided by Papouch Company. The protocol is well documented, including examples for each instruction, and it comes with a convenient terminal for communication tuning.³
- Using .NET library [Spinel.NET](#) (C#) available [on GitHub](#) (with Czech documentation).
- Via Wix Software.
- We can modify or implement your protocol in Quido upon request. Do not hesitate to contact us.

Block diagram

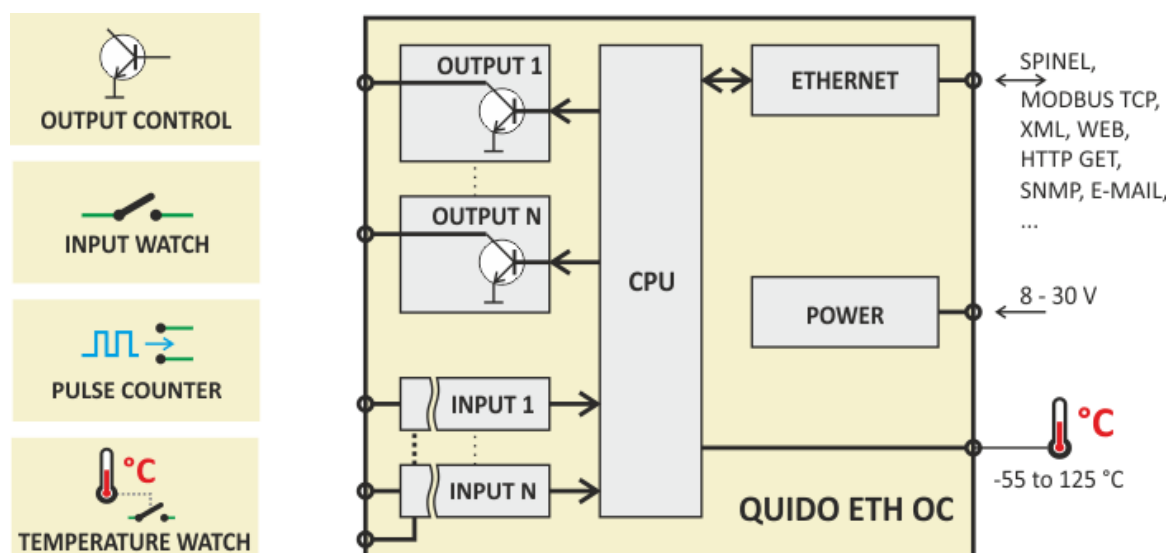


Fig. 1 - Block diagram of the Quido ETH OC module

² MODBUS documentation is available in a separate document.

³ SpinelTerminal can be downloaded free of charge from en.papouch.com.

Available designs

Temperature sensor design:

- Sealed in shrinkable coating (*standard design*)



Fig. 2 – sensor in shrinkable coating

- In metal housing



Fig. 3 – sensor in metal housing

Sensor cable length:

- 3 m (*standard design*)
- 1 m, 5 m, 10 m, 15 m

Thermal resistance of the sensor cable:

- -10 to +70 °C (*standard design*)
- -60 to +200 °C – silicone cable; blue color

Cover and mounting:

- Electronics board only (*standard design*)

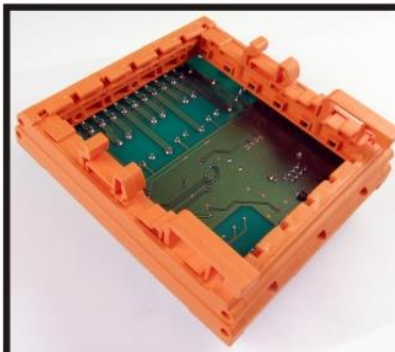


Fig. 4 – example of standard design (here: Quido ETH 2/16 OC M)

- All mounting options are illustrated below (here: Quido ETH 4/4):



Board with a DIN rail



Board with a DIN rail (below)



Board with a plexi cover



Board with a plexi and DIN



Board in a box with plexi



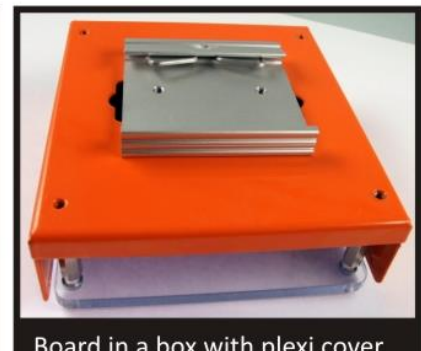
Board in a box



Board in a box (below)



Board in a box with DIN (below)



Board in a box with plexi cover and DIN rail mount (bottom)

Examples of functions that can be added upon customer's request:

- outputs will close depending on the preset combination of inputs
- modifications allowing you to use your communication protocol
- Memory for a greater number of changes in the input state.
- ... we will be happy to add features tailored to your application.

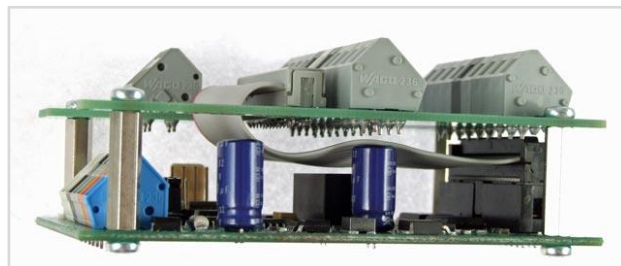
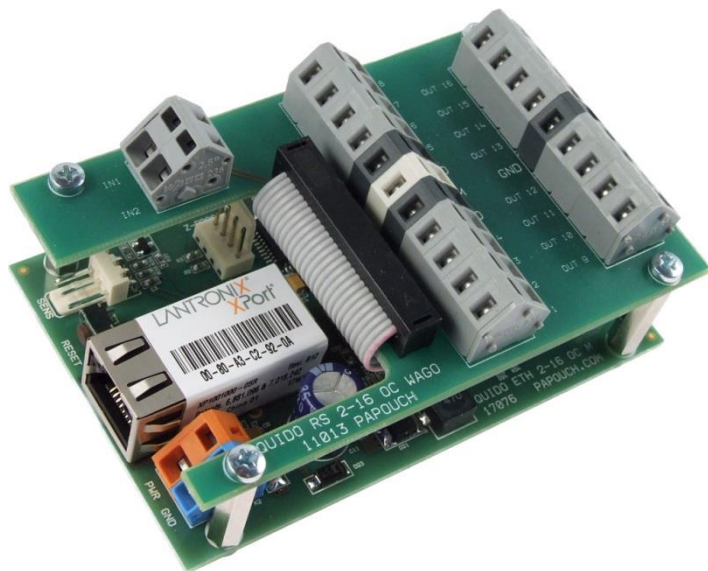
Do not hesitate to contact us if you have any other requirements concerning the design and functions of Quido ETH OC.

Available accessories

Additional terminal block for inputs and outputs

For easier connection of connectors, an additional plate with terminals for wires can also be ordered. On this board there is one free wire terminal for each input and output. The total height of the assembly is 43 mm.

The board is designed for 16 outputs and two inputs. One terminal block must be ordered for the Quido ETH 2/16 OC. For Quido ETH 4/32 OC, two expansion terminal blocks must be ordered.



Power supply unit for Quido – AC adapter

Switched power supply unit for 100 to 240 V; AC adapter design.

Available versions: 12V/0,5A; 12V/2A; 24V/1A



Power supply unit for Quido – mountable to DIN rail

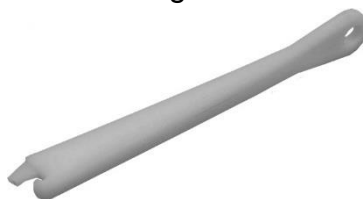
Switched power supply unit for 100 to 240 V; mountable to 35 mm DIN rail.

Available versions: 12V/0,5A; 12V/2A; 12V/4,5A; 24V/1,5A; 24V/2,5A



Lever for Wago 236 terminals – plastic

a tool enabling easy connection of wires to Wago 236 terminals (standard part of delivery)

**Lever for Wago 236 terminals – metal**

a tool enabling easy connection of wires to Wago 236 terminals (sturdy metal design)

**FIRST CONNECTION**

Tip: Quido comes with a plastic lever for easy connection of wires to the Quido. Using the lever can be seen in the following picture.⁴

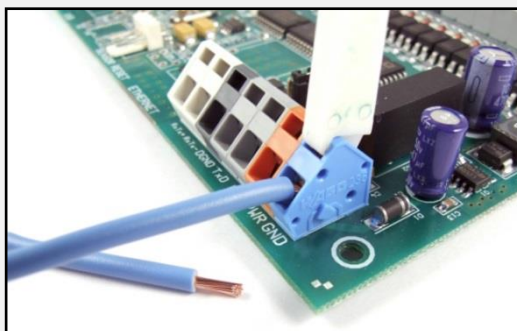


Fig. 5 - use of lever for easy connection of wires

- 1) Connect a DC supply voltage from the 8 to 30 V range to the PWR (+ orange terminal) and GND (– blue terminal) terminals. The Quido has built-in reverse-polarity protection. When power is connected, the PWR indicator on the board lights up.
- 2) Connect Quido to the computer network.
 - a. Connect the Quido to the network switch using a standard cable.⁵
 - b. If your network does not have an address range compatible with the IP address (**192.168.1.254**) and netmask (**255.255.255.0**) that Quido has factory set, set Quido to an address suitable for your network with the Ethernet configurator. This software is available for download at en.papouch.com/ec.

⁴ Terminals can also be operated with a small flathead screwdriver.

⁵ If you want to connect the Quido directly to PC, use a cross-wired cable.

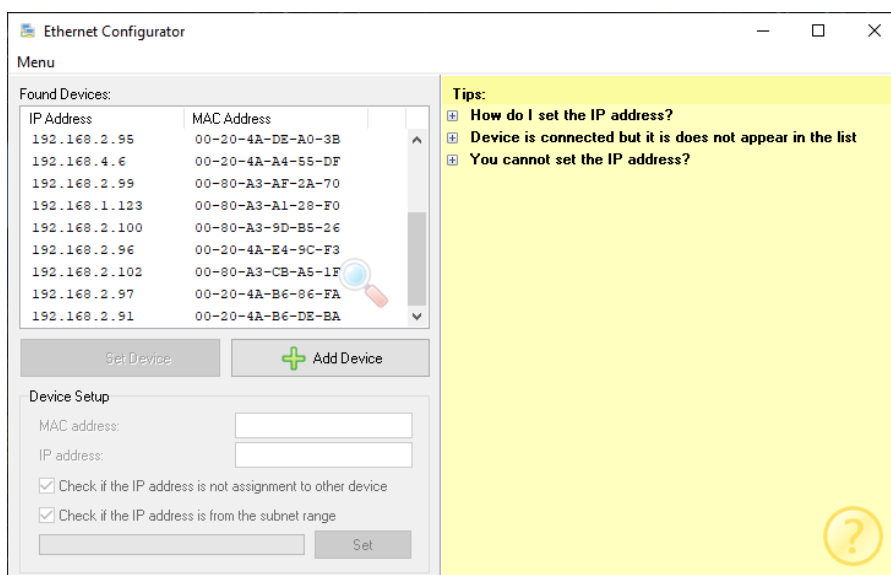


Fig. 6 - Ethernet Configurator for easy IP address setup

c. After setting the address, you can connect to Quido using a web browser. The web interface is available directly at the Quido IP address.

- 3) Connect the temperature sensor to the connector labeled SENS (or TEMP). (You only need to connect the thermometer if you will be using temperature measurement or one of the temperature monitoring functions.)⁶



Fig. 7 - connector for thermometer connection

- 4) Connect the input and output terminals as required (see the following section for a description).
- 5) For information on the web interface, control software and communication protocols that can be used to control and communicate with Quido, see page 15.

WIRING INPUTS AND OUTPUTS

For inputs and outputs, standard double-row 26-pin connectors with a pitch of 2.54 mm are available (see Fig. 8).

Tip: For easier connection of connectors, an additional plate with terminals for loose wires can also be ordered. On this board, there is one free wire terminal for each input and output. See detailed picture on page 10.

⁶ If the electronics board contains the connector but you do not have a temperature sensor, the sensor can be ordered separately.

Inputs

The inputs are designed for a switching contact connected to ground. The inputs are galvanically connected to the ground supplying the device.

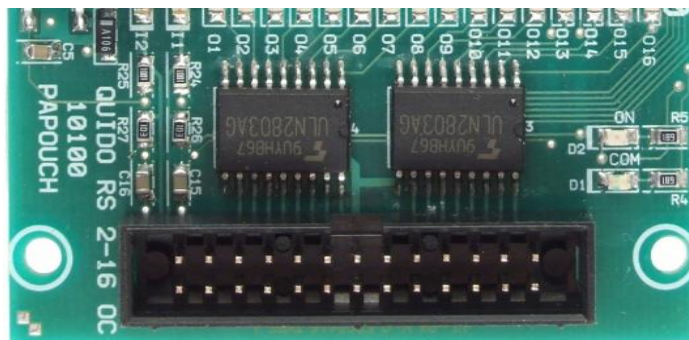


Fig. 8 - Inputs to Quido ETH 2/16 OC

Connection of connector(s) with inputs with example of contact connection to input 2 is shown in Fig. 9.

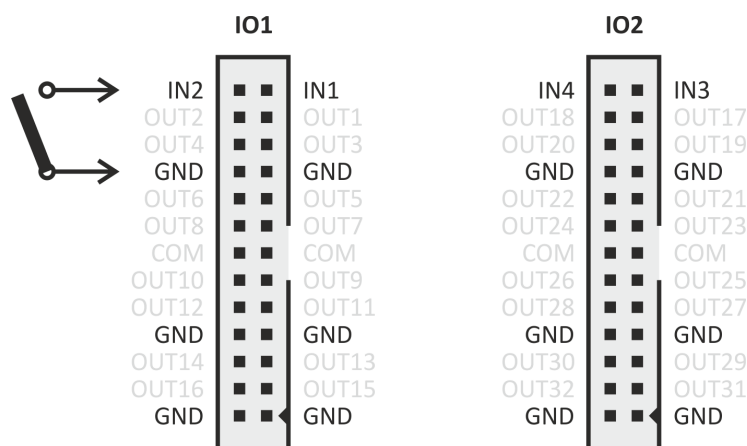


Fig. 9 - Connection of input pins and ground on I/O connectors
(on Quidos with lower number of I/O there is only IO1 connector)

Outputs

Outputs are fitted with a transistor array. Each output is implemented as an open collector. When the output is active, it has a logic zero (the transistor switches against ground). If it is inactive, the output is in a high impedance state.



Fig. 10 - Outputs on Quido ETH 2/16 OC

Output connector wiring with an example of the connection of output 2 is on Fig. 12. In the case of an inductive load connection, connect the output as shown in Fig. 13.

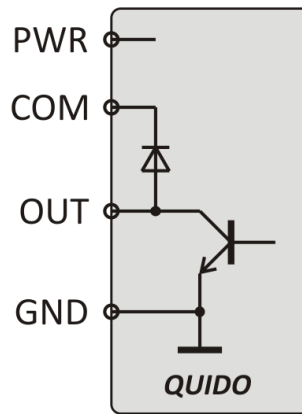


Fig. 11 - internal wiring of the output

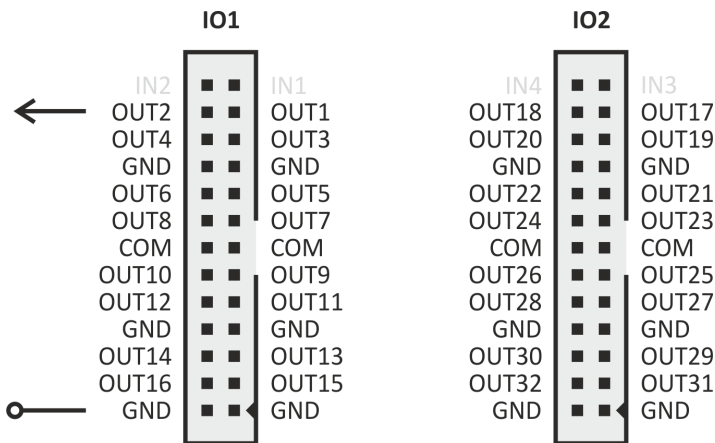


Fig. 12 - Connection of output pins and ground on I/O connector
(on Quidos with lower number of I/O there is only IO1 connector)

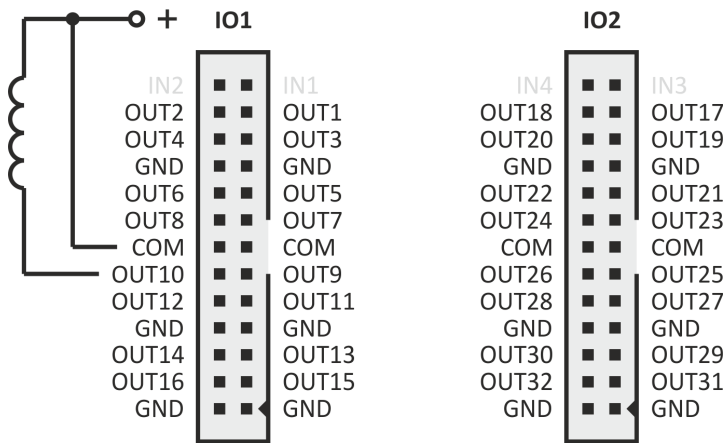


Fig. 13 - Example of inductive load connection
(on Quidos with lower number of I/Os there is only IO1 connector)

CONFIGURATION AND CONTROL

After being connected, Quido can be controlled and configured by one of the following ways:

	Monitoring	Control	Configuration
Web interface	✓	✓	✓
Wix Software	✓	✓	
Ethernet Configurator Software			✓
Spinel Protocol	✓	✓	✓
MODBUS TCP industrial protocol	✓	✓	
SNMP network protocol	✓	✓	
HTTP GET requests	✓	✓	
Via XML	✓		
Telnet protocol			✓
E-mail	✓		

WEB INTERFACE – CONTROL

Web interface allows complete monitoring, control and configuration of Quido. The web interface requires JavaScript to be enabled. The interface is optimized for Mozilla Firefox, Google Chrome and Microsoft Edge (Chromium core).

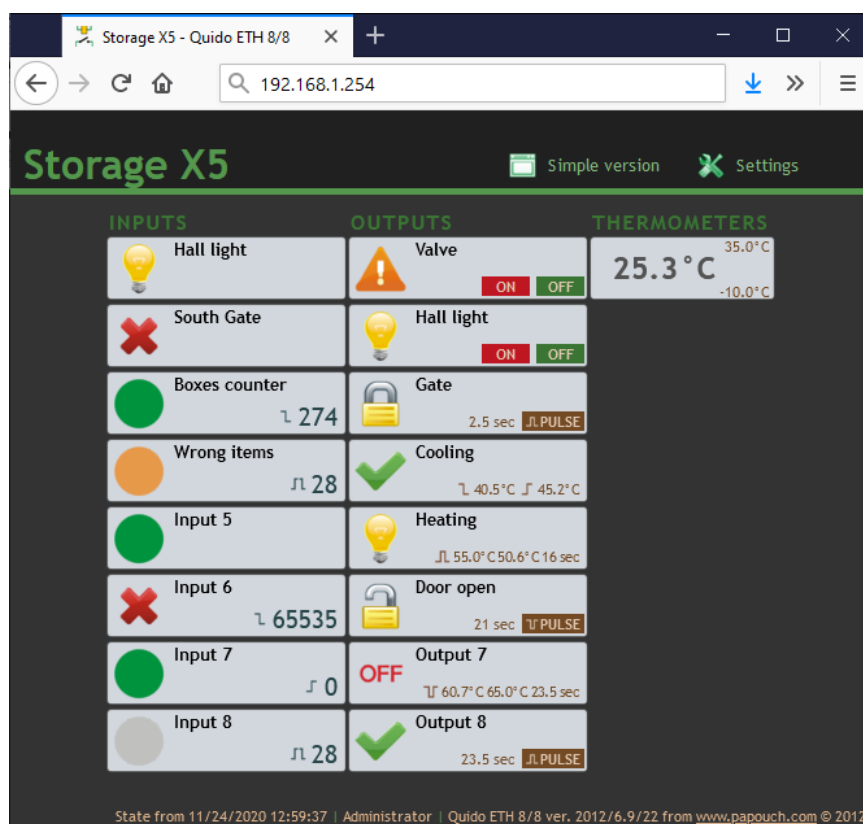


Fig. 14 – the main page of Quido ETH 8/8 opened in Chrome 81

To enter the settings, use the button on the top right of the main Quid web page. The settings are organized into the following ten panels:

- *Network* - network interface configuration, IP address, subnet mask, TCP channel and Modbus port
- *Security* - setting passwords for accessing the web page
- *Emails* - sending emails about changes and current status
- *SNMP* - integration into your network SNMP manager
- *Sending* - sending statuses to your server using HTTP GET + setting statuses
- *Inputs* - configuration of inputs, their names, images, counters, ...
- *Outputs* - configuration of outputs, their names, images, temperature monitoring, ...
- *Thermometers* - temperature monitoring
- *Other* - device name, site language selection, temperature units
- *Info* - information about the device

Tips for working with the settings:

- The default language of the website is English. The website can be switched to Czech language in the settings on the *Other* panel, using the *Language* parameter.
- The login name is always *user* or *admin*.
- Pop-up help for all setting items is displayed automatically when you hover over the field. (Help will also appear when you hover over the headings marked with a question mark at the end.)
- The settings window can be moved with the mouse by grabbing the upper right corner of the window.

Network

Setting of network parameters of the device, mode and communication via TCP/UDP.

Network settings

DHCP ☐

Device's IP address

Netmask

Gateway IP address

DNS server's IP address

WEB port

Local port for TCP/UDP connection

ModBus Port

Device mode?

Communication mode

Send message about inputs changes individually (TCP/UDP only) ☐

Remote device for TCP/UDP connection?

Remote IP address

Remote port

Fig. 15 - network settings panel

IP address of the device

The IP address of the device. If you are not sure of the correct IP address, consult your network administrator. (The factory default IP address is 192.168.1.254.)

Subnet mask

The mask of the network to which the device is connected.

Gateway IP address

Network gateway address. An important item especially when sending emails and HTTP GETs.

DNS server IP address

A DNS server is required if you have entered name addresses on the *SNMP*, *E-mail*, or *Sending* tabs.

Local port for TCP/UDP connection

The port number on which the device expects an incoming connection in TCP server mode.

Port for MODBUS TCP

Enter the port number on which the MODBUS TCP protocol is to communicate. The default port is 502.

Device mode

The following options are available:

WEB: A mode in which all functions are available except for data communication via the Spinel protocol.

In the following modes, you cannot monitor the current values on the web page, you cannot send emails and HTTP GET and POST messages, you cannot use SNMP and MODBUS TCP. In these modes, the device communicates using the Spinel protocol:

TCP server: The device expects a data connection on its own *IP address* and *Data port*.

TCP client: The device connects to the *Remote IP address* and *Remote port*.

UDP: The device communicates via the UDP protocol. It expects incoming messages on the *Data port*.

Remote device for TCP connection

Tip: If the other device is on a different network, be sure to set the IP address of the gateway as well.

Remote IP address

IP address of the remote device (usually a server) to which Quido connects in TCP client mode.

Remote port

Port number of the remote device (usually a server) to which Quido connects in TCP client mode.

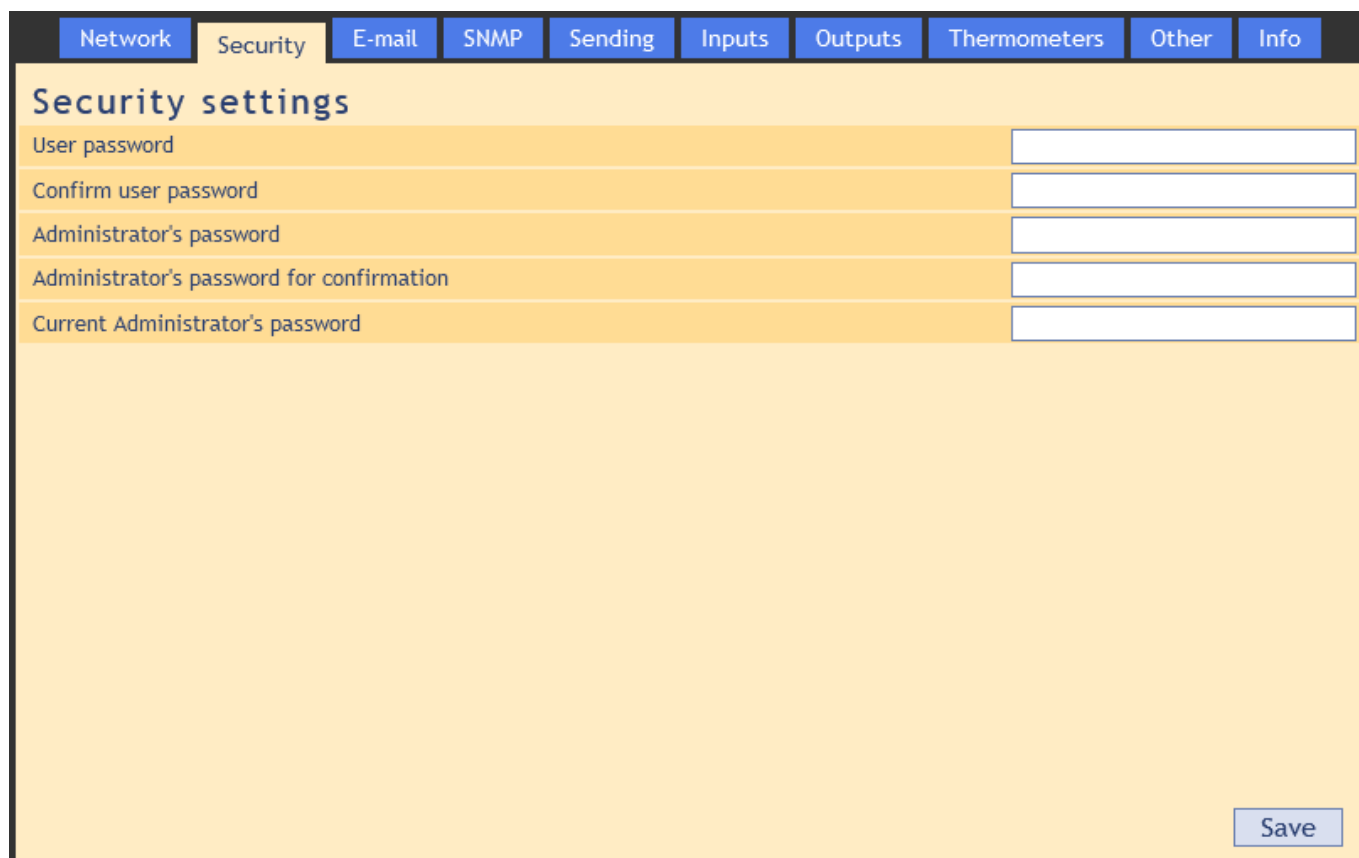
Reset the device

This button can be used to reset all Quido parameters to the default state. Only the IP address remains unchanged. The site port will be changed to 80.

(The *Reset* button is only available in WEB mode.)

Security

Security settings for access to the web interface and mobile site.



The screenshot shows the 'Security settings' panel. At the top, there is a navigation bar with tabs: Network, Security, E-mail, SNMP, Sending, Inputs, Outputs, Thermometers, Other, and Info. The 'Security' tab is active. Below the tabs, the title 'Security settings' is displayed. There are five input fields for passwords: 'User password', 'Confirm user password', 'Administrator's password', 'Administrator's password for confirmation', and 'Current Administrator's password'. A 'Save' button is located at the bottom right of the panel.

Fig. 16 - security settings panel

User Password and User Authentication Password ⁷

Enter the password for user access here. This security level only allows monitoring the current status of inputs and outputs, or controlling outputs and resetting counters. Access to settings is not allowed.

The user's name at login is always *user*

If a password is entered for the user, a password must also be entered for the administrator.

To cancel the password, leave the field blank.

Administrator password and Administrator password for authentication ⁷

Enter the password for administrator access here. This security level allows access to the device configuration.

The administrator's name at login is always *admin*

To cancel the password, leave the field blank.

Current administrator password ⁷

If the administrator has set a password for the current login, enter it here. It is not possible to change password without entering the current password.

⁷ The field is used only for entering values. After saving, no settings are displayed for security reasons.

E-mail

Setting of sending emails when changes in inputs and outputs occur, or when the set temperature limits are exceeded. The *Test* button can be used to test the current email settings. When clicking this button, at least one of the options *Send emails on changes* or *Send emails when temperature limits are exceeded* must also be checked.

E-mail settings

Activate the e-mails sending globally ☒

SMTP server's name

SMTP Port

Sender's e-mail address

Recipient's e-mail address

Send e-mails upon changes ☒

Send e-mail when the threshold is exceeded ☒

SMTP authorization

SMTP server requires verification ☐

Verification name

Verification password

Re-enter the password

Fig. 17 - e-mail settings panel

SMTP server name

Enter the name or IP address of the SMTP server through which you want to send e-mails.

Sender's email address

Quido e-mail address - the address from which information about changes will come.

E-mail address of the recipient

E-mail to which information from Quido should be sent.

Send emails when changes are made

Check if you want to send an email immediately after a change in the status of inputs and outputs that have status change monitoring set.

Send emails when temperature limits are exceeded

If checked, information emails will be sent whenever the temperature limits are exceeded.

SMTP authorization

You can enter login credentials here, if the SMTP server you are using requires sender identity verification.⁸

⁸ The field is for input only. After saving, no settings are displayed for security reasons.

SNMP

Parameters for integrating Quido into your SNMP manager.

The screenshot shows the 'SNMP settings' panel. At the top, there is a navigation bar with tabs: Network, Security, E-mail, SNMP (active), Sending, Inputs, Outputs, Thermometers, Other, and Info. The main content area is titled 'SNMP settings'. It contains three input fields: 'SNMP manager's IP address' (192.168.12.34), 'Read community name' (public), and 'Write community name' (private). Below these is the 'SNMP traps' section with three checkboxes: 'Allow trap sending' (checked), 'Send SNMP trap when the threshold is exceeded' (checked), and 'Send SNMP trap upon I/O change' (unchecked). A 'Save' button is at the bottom right.

Fig. 18 - SNMP settings panel

IP address of the SNMP manager

IP address of the server that collects SNMP messages from devices on the network.

The name of the reading community

The name of the SNMP read-only community.

Name of the enrolment community

The name of the SNMP enrollment community.

Global trap activation

Check this box if you want to globally enable sending SNMP traps. According to other settings, it can be periodic sending of traps with current statuses or sending traps when the set limits are exceeded.

Send trap when limits are exceeded

Check the box if you want to activate sending SNMP traps when the set limits are exceeded.

Send trap on changes

Check the box if you wish to activate sending SNMP traps when changes are made to inputs and outputs (according to the settings for specific inputs).

Sending

Parameters for automatic sending of current values to the server via HTTP protocol. The responses from the server can include parameters that change the state of the outputs or reset the counters.

Fig. 19 - panel of settings for sending current statuses via HTTP protocol

Web server address

Enter the name or IP address of the web server that should receive the current values. If you do not know what address to enter, contact your server administrator.

Port of the web

The web port number of the server that is to receive the measured temperature. Usually this is number 80, sometimes 8080, etc.

Script directory on the server

Enter the local path to the scripts directory here. If the script that receives the temperature is `www.server.net/scripts/get.php`, enter this part in this field: `scripts/`

Script name

The name of the script that receives HTTP GET.

Periodic sending of current values

If you wish to send values periodically, enter the time in seconds. If you do not want to send values periodically, enter 0.

Send GET on changes

Check if you want to send GET immediately after changing the state of inputs and outputs that have watching set.

Inputs

Complete settings directly related to each input. The input number is selected at the top of the window using the arrows or the drop-down list.

Fig. 20 - input settings panel

Input name

Enter the name to be used for this input.

Images for ON and OFF states

For both states, you can select the images that are best suited for a particular input use from a predefined list.

Hide input

This checkbox can be used to hide the input from the main page view. (It only hides the view. The input continues to work normally, you can monitor its status, etc.)

Watch for changes

If this field is checked, an e-mail, HTTP GET, or SNMP trap is sent when the status changes.

Counter mode

You can set any of the following modes:⁹

- *Off*: The change counter is disabled.
- *Rising edge counting*: at the moment of transition from OFF to ON, one is added to the counter state.

⁹ Note: The counter is reset (1) by clicking on the counter status on the main page, (2) when Quido is turned on, (3) when Quid is restarted, and (4) by changing the communication parameters.

- *Falling edges counting:* at the moment of transition from ON to OFF, one is added to the counter state.
- *Counts both rising and falling edges:* at the moment of transition from ON to OFF or from OFF to ON, one is added to the counter state.

For all inputs: number of samples for change of state

ATTENTION: This setting is common for all inputs! The change is common to all inputs. This setting is for experienced users only.

The device samples inputs every 1 ms. Specify the number of consecutive samples that must be identical for the input state change to be considered valid. A number from 1 to 255 can be entered. The default value is 10.

Outputs

Complete settings directly related to each output. The output number is selected at the top of the window using the arrows or the drop-down list.

Fig. 21 - Output settings panel

Output name

Enter a name to label this output.

Images for ON and OFF states

For both states, it is possible to select images from a predefined list that are best suited for a particular output application.

Hide input

This checkbox can be used to hide the output from the main page view. (Only the display is hidden. The output still works, it can be controlled normally, its status can be monitored, etc.)

Watch for changes

If this field is checked, an e-mail, HTTP GET, or SNMP trap is sent when the status changes.

Output mode

You can set any of the following modes:

- *Static:* The output status can be controlled by the ON and OFF buttons.
- *Pulse - Positive Pulse:* There is only one control button that causes the output to turn on for the time set for the *Pulse Length on Output* (in seconds). If the output has been switched on, it will be switched off after the specified time has elapsed.

- *Pulse - Negative Pulse:* There is only one control button that causes the output to open for the time set in the *Pulse length at the output* (in seconds). If the output has been switched off, it will be turned on after the specified time has elapsed.

Output mode	Pulse - positive pulse
Output pulse length	5 sec

Fig. 22 - items available when setting type of pulse output

- *Temperature monitoring - switch output:* in this mode, two temperature limits can be set to control the output status. The progression of the temperature and output status is displayed automatically in a graph. In addition, it is also possible to set what action should follow when the sensor is disconnected.

- *Temperature monitoring - open output*

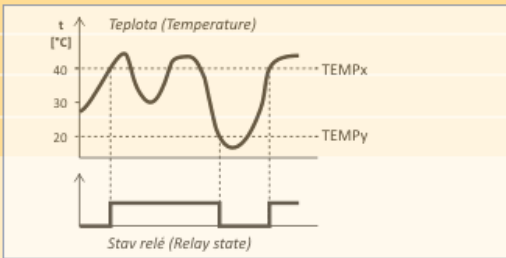
Output mode		Temperature watching - Turn th
TEMPx		40
TEMPy		20
Sensor detached action		Keep without change

Fig. 23 - items available in temperature monitoring mode - output switching

- *Temperature monitoring - switch on for the set time:* In this mode, two temperature limits and a switch on time can be set to control the output status. The progression of the temperature and output status is displayed automatically in a graph. In addition, the critical tendency can also be set (whether a rise or fall in temperature is monitored), as well as the action to follow when the sensor is disconnected.
- *Temperature monitoring – switch off for a set time:* In this mode, two temperature limits and an switch off time can be set to control the output status. The progression of the temperature and output status is displayed automatically in a graph. In addition, the critical tendency (whether the temperature rise or fall is monitored) can also be set, as well as the action to follow when the sensor is disconnected.

Thermometers

Here the parameters of the thermometer connected to the Quido are set.

Thermometer settings	
Hide thermometer	<input type="checkbox"/>
Watch temperature thresholds (limits)	<input checked="" type="checkbox"/>
Upper limit	80.0
Lower limit	10.0
Hysteresis	1.0
Periodic sending upon an event	60

Save

Fig. 24 - thermometer setting panel

Hide thermometer

This checkbox can be used to hide the thermometer from the main page view. (Only the display is hidden. If the thermometer is connected, it continues to work, the measured temperature is monitored, etc.)

Watch the temperature limits

If the box is checked, Quido checks that the temperature is within the specified limits.

Periodic sending when exceeded

If you wish to send periodic readings once limits are exceeded, enter the time in seconds. If you do not want to send the values periodically, but only at the time of the overrun, enter 0.

Other

Other settings

Name of the device

Language

Temperature units

Fig. 25 - panel for setting other parameters

Name of the device

This string can be used to name the device, for example, according to its location, etc. (Only characters without diacritics can be entered.)

Language

This is where you set the language in which the website communicates. English and Czech are available.¹⁰

Unit for temperature sensors

There are three temperature units to choose from: Celsius, Fahrenheit and Kelvin.

¹⁰ Additional language can be added upon request.

Info

More information about the device, MAC address, firmware version and useful links.

The screenshot shows a web interface with a top navigation bar containing tabs: Network, Security, E-mail, SNMP, Sending, Inputs, Outputs, Thermometers, Other, and Info. The 'Info' tab is selected. Below the tabs, the 'Information about the device' section lists: Device type: Quido ETH 10/1, S/N: 00228/03897, MAC address: 00-80-A3-DD-B4-FD, Firmware version: 7.2/22 (Created 17.08.2020), Core: Quido ETH 10/1; v0228.04.44; f66 97; t1, and Browser: Firefox 83. The 'Supplier of the device' section lists: Name: Papouch s.r.o. and Web site: www.papouch.com. The 'Links' section lists: XML file containing current configuration: [settings.xml](#).

Network	Security	E-mail	SNMP	Sending	Inputs	Outputs	Thermometers	Other	Info
Information about the device									
Device type: Quido ETH 10/1									
S/N: 00228/03897									
MAC address: 00-80-A3-DD-B4-FD									
Firmware version: 7.2/22 (Created 17.08.2020)									
Core: Quido ETH 10/1; v0228.04.44; f66 97; t1									
Browser: Firefox 83									
Supplier of the device									
Name: Papouch s.r.o.									
Web site: www.papouch.com									
Links									
XML file containing current configuration: settings.xml									

Fig. 26 - panel with additional information about the device

WIX SOFTWARE

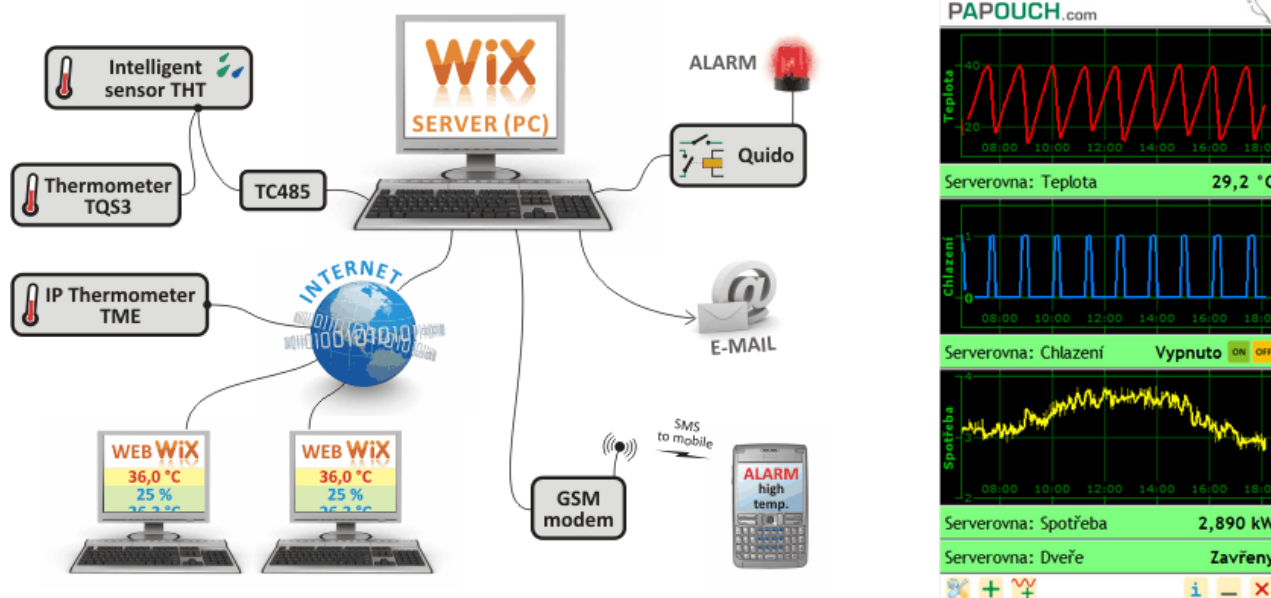


Fig. 27 – universal Wix software

The Wix universal software is able to work with most of our equipment so Quido can be easily integrated. It can be used to monitor the status of inputs, outputs, counters, and the thermometer. Besides, the program gathers and combines values from all devices connected to it and uses them to perform various actions: to control outputs, send e-mails, display alerts, send SMS via a connected modem, etc. Wix can be downloaded from en.papouch.com.

SPINEL PROTOCOL

Spinel is the default protocol that Quido uses for communication. It is available both in ASCII and binary versions. It can be used to fully control and configure Quido. All commands are well documented including examples for each instruction. Spinel also comes with SpinelTerminal program for comfortable tuning of communication via the Spinel protocol.

For full documentation see a separate document called Quido – Spinel. The SpinelTerminal can be downloaded from en.papouch.com.

Spinel.NET C# library

Spinel protocol can also be implemented using [Spinel.NET](http://en.papouch.com) library designed for .NET. It is freely available [on GitHub](http://en.papouch.com). (Czech documentation only.)

Example of turning output OUT3 on for 5 seconds: ([more here](#)):

```
if (MyQuido.CmdSetOutput(3, true, 10))
    Console.WriteLine("Ok");
else
    Console.WriteLine("Error");
```

Example of reading impulses from the IN2 counter: ([more here](#)):

```
if (MyQuido.CmdGetCounter(2, out int counter))
    Console.WriteLine($"Counter value is {counter}.");
```

MODBUS TCP PROTOCOL

Quido can also communicate via the industry standard protocol MODBUS TCP. Full documentation of MODBUS TCP can be downloaded from en.papouch.com.

TELNET CONFIGURATION

Connection

Unknown IP address

For setting the IP address we recommend using Ethernet Configurator (more on page 12).

- 1) Open a window with command prompt (In Windows OS, click *Run* and type `cmd`.)
- 2) Make a record to ARP table:
 - a. Type `arp -d` and confirm by Enter. That deletes current ARP table.
 - b. Following command assigns IP address to MAC address:
`arp -s [new_ip_address] [MAC_address]`
example: `arp -s 192.168.1.254 00-20-4a-80-65-6e`
- 3) Open Telnet. (by typing `telnet` and pressing Enter. ¹¹)
- 4) Type `open [new_ip_address] 1` and confirm.
- 5) Terminal shows an error message after a moment. Nevertheless this action is essential for correct IP address ARP entry.
- 6) Connect to the IP address of the TH2E. (by typing `open [IP address in dot format] 9999` and Enter.)
- 7) Now you have entered to the configuration of the Module. IP address is not yet set. It needs to be set in menu `Server Configuration > IP Address`.
- 8) If the IP address is valid, TH2E writes welcoming information ending by text:
Press Enter for Setup Mode
Now press Enter or the configuration will be terminated.
- 9) Device writes complete settings.
- 10) At the end of the entry, you can see „Change setup:” text containing groups of parameters that are configurable. For changing the network parameters choose *Server*. Here you can set a new IP address and other parameters.

¹¹ In OS Windows 10, the client for Telnet is not a standard part of the system. Installation procedure:

- a) Enter “windows features” to search bar in Windows 10 and choose *Turn Windows features on or off*.
- b) Windows features window opens, check Telnet Client and click OK.

IP address is known

- 1) In OS Windows click on Start/Run and type `telnet` and confirm.¹¹
- 2) Connect to the IP address of the TME. (by typing `open [IP address in dot format] 9999` and Enter.)
- 3) If the IP address is valid, TME writes welcoming information ending by text:
Press Enter for Setup Mode
 Now press Enter or the configuration will be terminated.
- 4) Device writes complete settings.
- 5) At the end of the entry you can see „Change setup:" text containing groups of parameters that are configurable. For changing the network parameters choose Server.

Telnet main menu

You can choose the menu items by pressing the number before them and confirming by Enter.

Menu structure is:

```

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

Server

Basic Ethernet settings.

Following items are in this part:

```

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

Quido module IP address. Enter the numbers separately and divide them by Enter.

Default: 192.168.1.254

Set Gateway IP Address

Gateway IP address

Type "Y" in the "Set Gateway IP Address" item to change the address. Enter the numbers separately and divide them by Enter.

Netmask

You can set here how many bits from IP address make the network part.

Subnet mask is entered as a number of bits determining range of possible IP addresses of the local network. If for example 2 is entered, subnet mask is 255.255.255.252. Entered value indicates number of bits from right. Maximum is 32.

Default: 8

Example:

Mask 255.255.255.0 (binary 11111111 11111111 11111111 00000000) -> number 8.

Mask 255.255.255.252 (binary 11111111 11111111 11111111 11111100) -> number 2.

Change telnet config password

Enter new Password

This item sets the password that is required before entering the Telnet configuration or via WEB interface (Administrator's password).

Type "Y" at „Change telnet config password“ to change the password

Factory Defaults

By pressing number 7 you can set the device to factory defaults.

This option sets the TH2E to defaults. IP address does not change, WEB port number will be changed to 80.

Exit without save

Ends the telnet session without saving.

Save and exit

This choice saves the changes. If some settings are changed, device restarts. That may take up to 30 seconds.

XML FILE

It is possible to get the current values from Quido in a text file in XML format. The file is available at [http://\[IP_address\]/fresh.xml](http://[IP_address]/fresh.xml) - for example at <http://192.168.1.254/fresh.xml> for Quido by default.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<root xmlns="http://www.papouch.com/xml/quido/act">
  <din id="1" name="Svetlo chodba" sts="0" val="0" pic="12" cmo="0" cnt="0" />
  <din id="2" name="Jizni brana" sts="0" val="0" pic="17" cmo="0" cnt="0" />
  <din id="3" name="Pocet krabic" sts="0" val="0" pic="1" cmo="1" cnt="274" />
  <din id="4" name=" " sts="0" val="1" pic="3" cmo="3" cnt="28" />
  <din id="5" name="" sts="0" val="0" pic="1" cmo="0" cnt="0" />
  <din id="6" name="" sts="0" val="1" pic="17" cmo="1" cnt="65535" />
  <din id="7" name="" sts="0" val="0" pic="1" cmo="2" cnt="0" />
  <din id="8" name="" sts="0" val="1" pic="4" cmo="3" cnt="28" />
  <dout id="1" name="Klapka" sts="0" val="0" pic="7" mde="0" pars="" />
  <dout id="2" name="Osvetleni haly" sts="0" val="1" pic="12" mde="0" pars="1;0;300;351;5" />
  <dout id="3" name="Brana" sts="0" val="0" pic="10" mde="1" pars="0;0;0;0;5" />
  <dout id="4" name="Chlazení" sts="0" val="0" pic="16" mde="4" pars="2;0;405;452;6" />
  <dout id="5" name="" sts="0" val="0" pic="12" mde="5" pars="1;1;506;550;32" />
  <dout id="6" name="Jizni vjezd" sts="0" val="1" pic="11" mde="2" pars="0;0;0;0;42" />
  <dout id="7" name="" sts="0" val="0" pic="15" mde="6" pars="1;1;607;650;47" />
  <dout id="8" name="" sts="0" val="0" pic="16" mde="1" pars="3;1;600;653;47" />
  <temp id="1" sts="0" val=" 25,3" th=" 35" tl=" -10" tenb="1" />
  <status location="U Papoucha" unit="C" />
</root>
```

There are four types of tags in the XML file: *din*, *dout*, *temp* and *status*

Inputs: din

id

Unique identifier of the digital input on the device. The first input has the number 1.

name

User-defined input name.

cnt

The current state of the change counter on this input.

cmo

Change counter mode:

- 0..... The counter is off.
- 1..... The counter adds one at each rising edge.
- 2..... The counter adds one on each descending edge.
- 3..... The counter adds one at each rising and falling edge.

val

Current input status (0 or 1).

sts

Describes the current state of the input. It can be one of the following:

- 0 value *val* is valid and represents the current state of the input
- 1 awaiting first status determination
- 4 an error occurred during status detection

Outputs: dout

id

Unique output identifier on the device. The first output has the number 1.

name

User-defined name of the output.

val

Current output status (0 or 1).

sts

Describes the current state of the output. It can be one of the following:

- 0 value *val* is valid and represents the current state of the output
- 1 awaiting first status determination
- 4 an error occurred during status detection

mde

Current output mode. It can be one of the following:

- 0 Manual control mode
- 1 Time switch (positive pulse)
- 2 Time release (negative pulse)
- 3 Temperature monitoring - switching
- 4 Temperature monitoring - opening
- 5 Temperature monitoring - time switch (positive pulse)
- 6 Temperature monitoring - time switch (negative pulse)

Thermometers: temp

id

Unique thermometer identifier on the device. Numbered from 1.

Val

Current measured temperature in set units.

sts

Describes the current state of the thermometer. It can take the following values:

- 0 Value *val* is valid and represents the current temperature
- 1 Waiting for the first temperature measurement
- 4 sensor error

tl

User-defined lower temperature limit.

th

User-defined upper temperature limit.

status

location

User-defined device name.

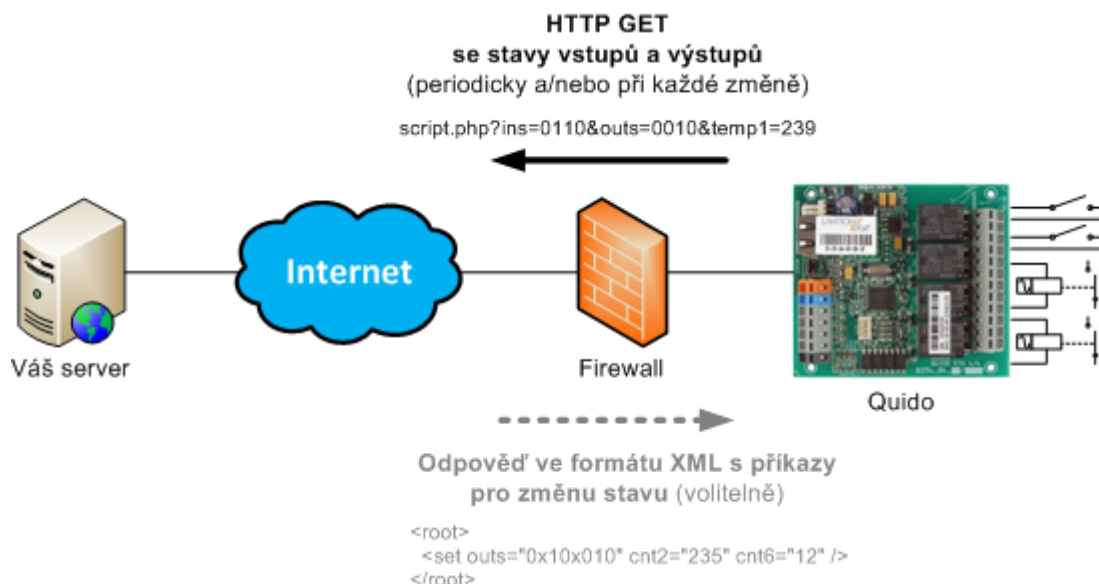
unit

The currently set temperature unit symbolized by the capital letter C (Celsius), F (Fahrenheit) or K (Kelvin).

HTTP GET

HTTP GETs can be used to **send the current status of Quido** to a remote server or to **control Quido outputs and counters** (page 39).

Sending the current status to a remote server



Data from Quido can be easily sent to a web server that processes the values sent by Quido as parameters of an HTTP GET request. Quido periodically calls a script on a server on the Internet (or company Intranet) with the current data in the HTTP GET parameter. On the server, these values can be stored, for example, in a database and further processed - analyzed or just displayed on the web to other users.

Quido settings specify the path to the script on the server. Depending on the settings, Quido sends the current values periodically or whenever the input or output state changes.

In addition to the actual data, the MAC address is also sent to the server to uniquely identify the Quido, in case the server receives data from multiple devices.

If the HTTP GET response includes a setup command, the server can also set the state of outputs and counters on inputs in this way.

Sending method

In this type of request, parameters are sent in the message address as standard HTTP GET parameters. Example:

```
www.example.com/script.php?mac=00-20-4A-B4-8D-F7
&name=Office&ins=01101010&outs=00100010&tempS=0&tempV=21.8&cnt2=235&cnt6=126
```

As you can see from the example, the request has a format familiar to web programmers from standard form data sending. There is no need to learn new programming techniques and the mechanism known from web form processing is sufficient for processing (< form name="myform" action=...).

HTTP GET is sent in two cases:

- 1) **Periodically:** If the *Periodical Sending* option is selected in the configuration of the *Sending* tab, HTTP GET is sent every X seconds (with the maximum of 3600 seconds).

- 2) **Immediately after change:** It can be set separately for each input and output whether to monitor its status changes. If a change occurs on one of these inputs/outputs, the HTTP GET is sent immediately.

Request parameters

The parameters in GET are as follows:

- `mac` MAC address of the device to uniquely identify the sender
- `name` Name of the device, set by the user.
- `ins` String with the status of individual inputs. For each input on the device there is a 0 (inactive) or 1 (active) character in the string depending on the current status.
- `outs` String with the status of individual outputs. For each output on the device, the string contains a 0 (open) or 1 (closed) character depending on the current state.
- `tempS` Describes the status of the measured temperature. It can be one of the following:
- 0value is valid and represents the currently measured value
 - 1waiting for the first measurement
 - 2the measured value exceeded the user-set upper limit
 - 3the measured value has fallen below the user-set lower limit
 - 4value not valid - measurement error or sensor error (meaning damaged sensor or cable)
- `tempV` Measured temperature as a decimal number without units.
- `cntX` Number of units counted by the counter on the X input. This parameter is given as many times as there are inputs. The X character is replaced by the input number (input numbering starts from one).

Setting the Quido in the GET response

The status of outputs and counters on Quido can also be easily set by a string directly in the HTTP GET response. This is particularly useful in cases where it would not be desirable for security reasons to open the possibility of connecting to the local computer network from the Internet. The server on the Internet will then wait for a regular HTTP GET from Quido and send a command in response to change the state of the inputs or to reset the counters.

The response structure is formatted as XML and looks like this:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<root>
  <set outs="0x10x010" cnt2="235" cnt6="12" />
</root>
```

In the `set` node there are individual parameters that can be used to set the state of the outputs or change the state of the counters. There should be no more information in the XML than the ones listed above.

outs

A string with the desired state of each output. The parameter must contain a string of length exactly corresponding to the number of outputs. Each output corresponds to one character. The first one from the left is the first output. The following characters can be at each position:

0 Switch the output OFF. (Only works when the output is in manual mode.)

1 Switch the output ON. (Only works when the output is in manual mode.)

X Leave output unchanged. (x is the only valid value for temperature controlled outputs.)

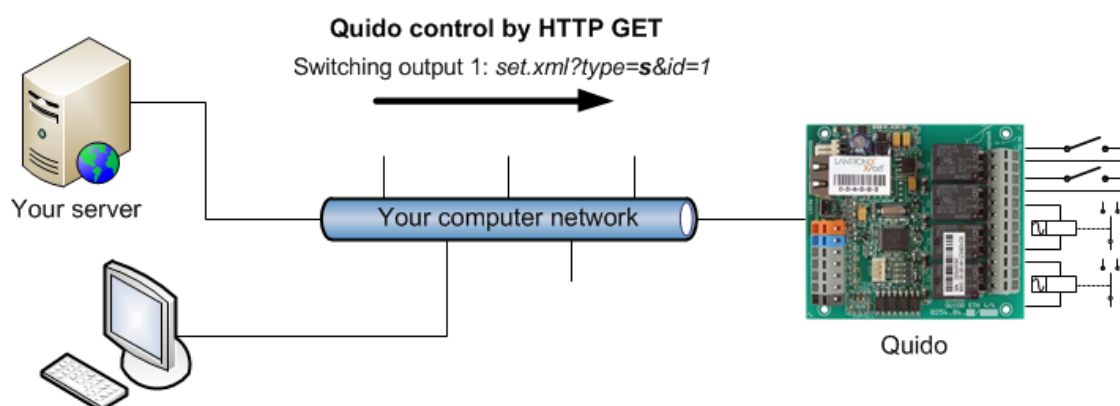
The above example is valid for Quido ETH 8/8.

cntX

These parameters (use the output number instead of X) can be used to subtract a value from the current state of the counter in Quido. The specified number is subtracted from the current counter state in Quido.¹² (The number must be equal to or less than the current counter state.)

There can be a maximum of ten cntX parameters in one instruction at a time. Quido will not process more than that.

Controlling Quido by HTTP GET



The following commands can be used to easily control the outputs and counters on the Quido. The *set.xml/* script in Quido is used to receive GETs. For Quido by default, the complete script address is: <http://192.168.1.254/set.xml>. **A user name and password** (if set in Quido) **is required** to run this script.

Switching an output ON

Example: *set.xml?type=s&id=1*

This command switches ON the output with the number in the *id* parameter (in this example it is the OUT1 output).

By adding the *time* parameter, it is possible to set how long the output should be switched on. It is possible to enter a number from the range 1 to 255, which corresponds to switching for 0.5 - 127.5 sec.

Switching an output OFF

Example: *set.xml?type=r&id=5*

This command switches OFF the output with the number in the *id* parameter.

By adding the *time* parameter, it is possible to set for how long the output should be open. It is possible to enter a number from the range of 1 to 255, which corresponds to an open time of 0.5 - 127.5 sec.

¹² The counter is not zeroed because zeroing could miss some pulses that would have come between the last counter state reading and zeroing.

Invert output

Example: `set.xml?type=i&id=12`

This command inverts the output state with the number in the *id* parameter.

Starting a pulse on the output

Example: `set.xml?type=p&id=2`

This command starts a set pulse on the output with the number in the *id* parameter.

Subtract from counter

Example: `set.xml?type=c&id=3&cnt=274`

This command can be used to subtract a specified number from the current counter state. The counter number is in the *id* parameter. The subtracted value is in the *cnt* parameter. (The *cnt* value must be equal to or less than the current counter value.)

Resetting of all counters

Example: `set.xml?type=C`

This can be used to zero all counters at once.

Switching all outputs ON

Example: `set.xml?type=S`

Using this command, all outputs can be switched ON at once.¹³

Switching all outputs OFF

Example: `set.xml?type=R`

Using this command, all outputs can be switched OFF at once.¹³

Bulk setting multiple outputs at once

Example: `set.xml?type=A&sts=00110x01`

Using this command, the status of multiple outputs can be set simultaneously.¹³ The *sts* parameter contains exactly as many characters as there are Quido outputs. (The example above is for Quido ETH 8/8, so the number of characters is 8.) The first character in the sequence represents the OUT1 output. Any of the following can be used as the character:

- 0 ... means to open the output
- 1 ... means to switch the output
- x ... means to leave the output unchanged

HTTP GET response

The response to the setting is in XML format. An example of the response is here:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<root>
  <result status="1" outs="6" val="0" pic="7"/>
</root>
```

The meaning of the individual parameters in XML is as follows:

¹³ Attention, the command only works if all outputs are in manual mode!

- There is always only one tag **result**.
- The **status** attribute contains the number 1 (the command was executed) or 0 (the command was not executed).
- The **outs** attribute contains a number that represents the state of all outputs. The number is decadic, after converting it to binary the state of each output can be found.

Example: the number 6 on the Quido ETH 8/8 means that output 2 and 3 are switched ON. (The number 6 in decadic is equal to 00000110 in binary.)

- **The val** attribute contains the number 0 (the output is on) or 1 (the output is off). This attribute is only valid if the command was for one specific output.

SENDING EMAILS

Quido can email the following information:

Input or output change message

If the input or output status changes for any reason, and it is set to¹⁴, Quido will send an email message. Example email:

Subject: QUIDO_info_Warehouse

Message body: Input 1:(Window) is ON

The subject line of the email contains the user-defined name of the Quido. The body of the message contains information about whether it is an input or output, its user-set name, and its status (ON or OFF).

Temperature limit exceeded message

If the temperature from the connected thermometer exceeds set limits, Quido sends an e-mail message. Depending on the settings, Quido can send the message repeatedly until the temperature returns to the set limits. Example:

Subject: QUIDO_info_Warehouse

Body of the report: Temperature alert! Upper limit: 26.0 C Current: 28.3 C

The subject line of the email contains the user-defined name of the Quido. The body of the message contains information about which limit has been exceeded and what the current temperature is.

Sensor error message

If temperature limit monitoring is enabled and a sensor error occurs, Quido sends one message about it. The error can be a disconnected or damaged sensor. Example:

Subject: QUIDO_info_Warehouse

Body of the report: Temperature sensor error

The subject line of the email contains the user-defined name of the Quido.

Test message to verify settings

Clicking the Test button in the Quido settings on the Email tab sends a test message to verify the settings.

Subject: QUIDO_info_Warehouse

Body of the report: Test message

¹⁴ This means that the monitoring of changes on that particular input or output is enabled and also the sending of emails is set to the appropriate tabs in the settings.

SNMP

SNMP protocol contains objects with current states. MIB table, which you can import to your SNMP manager is freely available on papouch.com. Quido uses SNMP version 1.

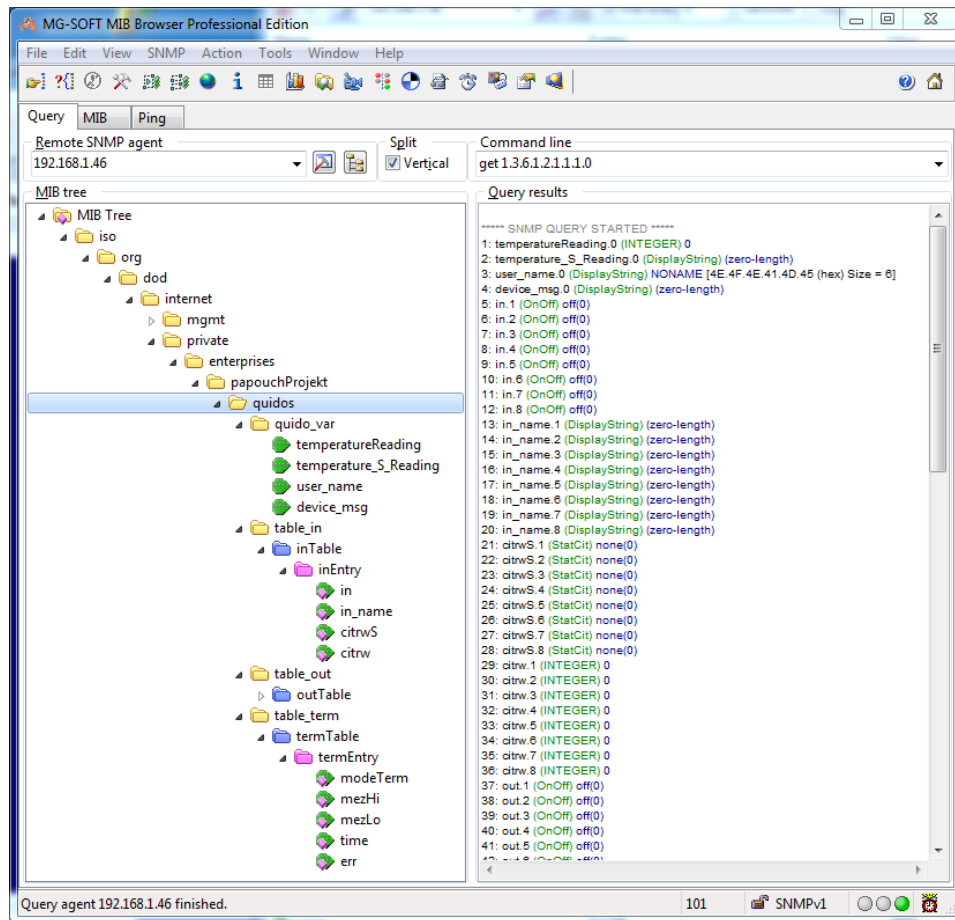


fig. 28 – MIB tree in Quido ETH OC

Tip: If you want to browse the whole SNMP tree using SNMPWALK utility (Linux), you need to specify exactly where the readout should start after the IP address. For example:

```
snmpwalk -v1 -c public 192.168.1.254 1.3.6.1.4.1.18248
```

If you only read with IP address, you will get just basic system IOD objects of the device.

Inputs

The inputs are designed for contact or voltage connection and can recognize two states: active/inactive input. A counter function can be activated on each input, where the number of state changes (according to the setting) is counted separately for each input. This can be used to count products, people, etc. The counter can count up to 65535 events.

Entry status

Name: inValue

ID: 1.3.6.1.4.1.18248.16.2.1.1.1.1 to number_of_entries

Type: Integer

Action: Read only

The object contains the current state of the input as number 0 (inactive) or 1 (active).

Name of input

Name: *inName*

ID: 1.3.6.1.4.1.18248.16.2.1.1.2.1 to *number_of_entries*

Type: *string*

Events: *Read, Write*

Length: 1 to 20 characters

The name of a specific input can be stored here for easier identification.

Counter settings

Name: *inCountEdgeSettings*

ID: 1.3.6.1.4.1.18248.16.2.1.1.3.1 to *number_of_entries*

Type: *Integer*

Events: *Read, Write*

This object contains the current counter settings:

- 0 = counter on this input is disabled
- 1 = counter counts falling edges on input (changes from active to inactive state)
- 2 = counter counts rising edges on input (changes from inactive to active state)
- 3 = counter counts all changes (falling and rising edges)

Counter value

Name: *inCountValue*

ID: 1.3.6.1.4.1.18248.16.2.1.1.4.1 to *number_of_entries*

Type: *Integer*

Events: *Read, Write*

The object contains the current counter value. By writing to the object, the recorded value is subtracted from the current counter value.

Outputs

Outputs can be controlled instantly (on or off) or with timing. For example, switch on for 5 sec, switch off for 30 sec, etc.

Output status

Name: *outValue*

ID: 1.3.6.1.4.1.18248.16.3.1.1.1.1 to *number_of_outputs*

Type: *Integer*

Events: *Read, Write*

The object contains the current state of the output. The output can be controlled by writing. It is possible to enter the number 1 to switch the output ON or 0 to switch the output OFF. Entering a value of 2 activates the pulse function on the output according to the setting of the "Output timing" object.

Output name

Name: *outName*

ID: 1.3.6.1.4.1.18248.16.3.1.1.2.1 to *number_of_outputs*

Type: *string*

Events: *Read, Write*

Length: 1 to 20 characters

The name of a specific output can be stored here for easier identification.

Output timing

Name: *outPulseLen*

ID: 1.3.6.1.4.1.18248.16.3.1.1.3.1 to *number_of_outputs*

Type: *Integer*

Events: *Read, Write*

The upper byte contains the current timing status of the output. It can be one of the following:

0 = output is not timed

2 = if this output is activated in the "output status" object, it will be switched ON for the time set in the lower byte of this parameter

3 = if this output is activated in the "output status" object, it will be switched OFF for the time set in the lower byte of this parameter

The lower byte contains the current output interval. If it contains the value 0, the output is not timed. If it contains another value, this time (in seconds) remains until the interval expires:

$$\text{remaining_time} = \text{value} \times 0.5$$

Example for switching for 5 sec: Enter the number 0x020A into the parameter of this object (0x02 = switching; 0x0A = 5 / 0.5) and then activate the output by entering the number 2 into the "Output status" object.

Example for switching off at 50 sec: Enter the number 0x0364 into the parameter of this object (0x03 = switching on; 0x64 = 50 / 0.5) and then activate the output by entering the number 2 into the "Output status" object.

Temperature monitoring

Temperature monitoring is a function that allows you to control outputs based on temperature measurements. Limits can be set for each output. When they are exceeded, the output switches on, off, on for a set time, etc., according to the settings.

For an overview of all temperature monitoring modes, see the Appendix at the end of this document (starting from page 49).

Temperature monitoring mode

Name: *thermMode*

ID: 1.3.6.1.4.1.18248.16.4.1.1.1.1 to *number_of_outputs*

Type: *Integer*

Events: *Read, Write*

The object contains the number of the current temperature monitoring mode. It is possible to enter a number from 0 to 6. For an overview of the modes, see the Appendix at the end of this document (starting from page 49).

Upper limit setting - TEMPx

Name: *thermHigh*

ID: 1.3.6.1.4.1.18248.16.4.1.1.1.2.1 to *number_of_outputs*

Type: *Integer*

Events: *Read, Write*

The object contains the upper temperature limit as an integer. The upper temperature limit in degrees Celsius can be obtained as follows:

$$\text{upper_limit} = \text{value} / 10$$

The upper limit must not be less than the lower limit.

Setting the lower limit - TEMPy

Name: *thermLow*

ID: 1.3.6.1.4.1.18248.16.4.1.1.3.1 to *number_of_outputs*

Type: *Integer*

Events: *Read, Write*

The object contains the lower temperature limit as an integer. The lower temperature limit in degrees Celsius can be obtained as follows:

$$\text{lower_limit} = \text{value} / 10$$

The lower limit must not be greater than the lower limit.

Length of switching on or off

Name: *thermPulseLen*

ID: 1.3.6.1.4.1.18248.16.4.1.1.4.1 to *number_of_outputs*

Type: *Integer*

Events: *Read, Write*

The length of time the output will be on or off after the event is triggered. This parameter only applies to modes 3 to 6. For an overview of the modes, see the Appendix at the end of this document (starting from page 49).

Output behavior in case of temperature sensor error

Name: *snsError*

ID: 1.3.6.1.4.1.18248.16.4.1.1.5.1 to *number_of_outputs*

Type: *Integer*

Events: *Read, Write*

The object controls the state of the output in the event of a thermometer failure. If temperature monitoring is set and the temperature sensor measurement fails (for example, due to mechanical damage of the cable to the sensor), the following output action is performed:

0 = output remains in the same state as before the fault

1 = the output is switched OFF

2 = the output is switched ON

Temperature measurement

Temperature measurement status

Name: *temperatureStatus*

ID: 1.3.6.1.4.1.18248.16.1.6.0

Type: *Integer*

Action: *Read only*

The current state of the temperature measurement can be one of the following values:

0 = temperature is correct and within range

1 = temperature not available

2 = upper limit exceeded (thermHigh)

3 = lower limit exceeded (thermLow)

4 = measurement error

Reading the measured temperature as a number

Name: *temperatureValInt*

ID: 1.3.6.1.4.1.18248.16.1.1.0

Type: *Integer*

Action: *Read only*

The object contains the current measured temperature as an integer. The actual measured temperature in degrees Celsius is obtained as follows:

$$temperature = object_value / 10$$

Example: if the object contains the number 321, it means that the temperature was measured at 32.1 °C.

Reading the measured temperature as a string

Name: *temperatureValStr*

ID: 1.3.6.1.4.1.18248.16.1.2.0

Type: *string*

Action: *read only*

The object contains the current measured temperature as a string.

Example: at a temperature of -10.5 °C, if the object contains the following string: -10.5C

Temperature unit

Name: *temperatureUnit*

ID: 1.3.6.1.4.1.18248.16.1.5.0

Type: *Integer*

Action: *Read only*

Temperature unit:

- 0 = degrees Celsius
- 1 = degrees Fahrenheit
- 2 = Kelvins

Additional features

User text

Name: *userName*

ID: 1.3.6.1.4.1.18248.16.1.3.0

Type: *string*

Events: *Read, Write*

Length: *16 characters*

A 16-character variable for storing arbitrary user text. It can be used, for example, to name a device. The variable is stored in FLASH memory - it is retained even in case of a power failure.

System message

Name: *deviceMsg*

ID: 1.3.6.1.4.1.18248.16.1.4.0

Type: *string*

Action: *Read only*

This object contains the system trap messages. See the following page for an overview of possible traps.

Traps

Input change status

When the state changes on one of the inputs, a trap is sent containing the following information:

- 1) User text - text from the *userName* object, to identify the device).
- 2) Entry number.
- 3) Entry status.

The temperature has left the set limits

If the temperature exceeds the limits set for any of the outputs, a trap is sent containing this information:

- 1) User text - text from the *userName* object, to identify the device).
- 2) Output number.
- 3) The current temperature and also the limit that has been exceeded.

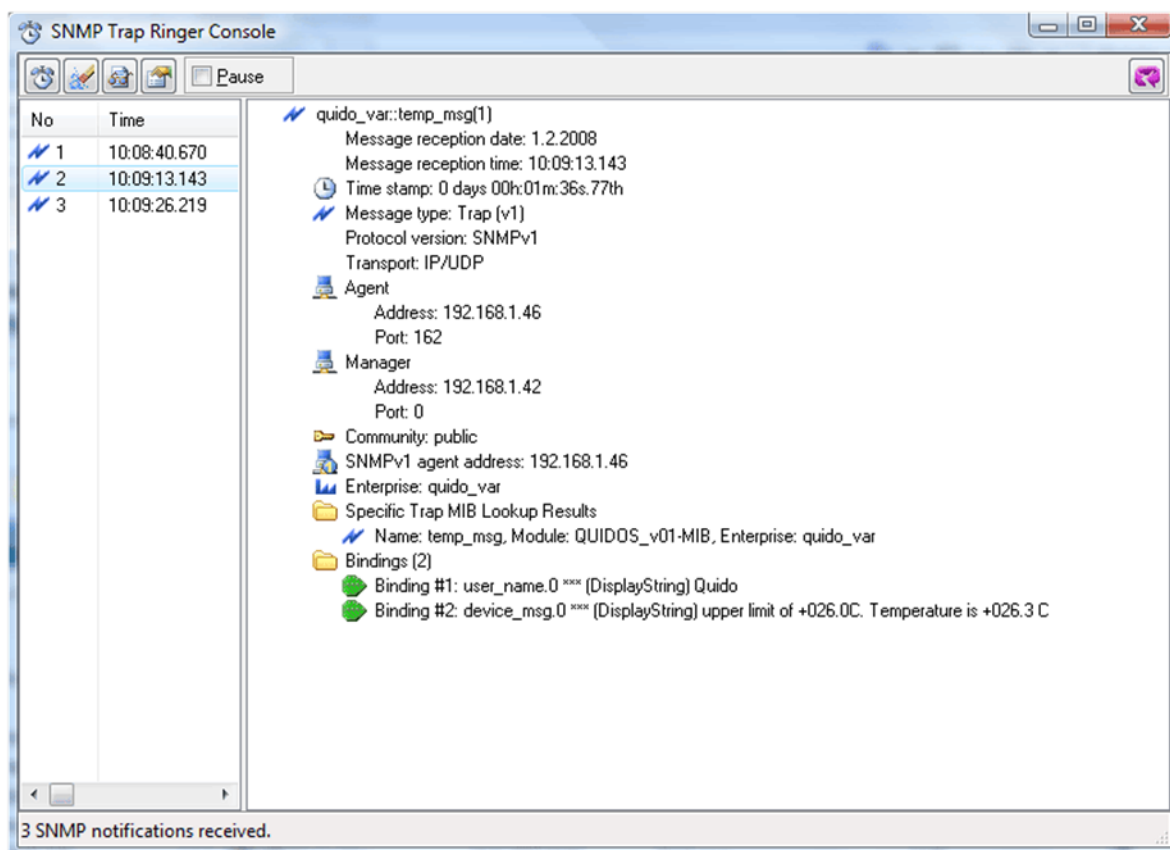


Fig. 29 - trap demonstration of exceeding the temperature range

Temperature monitoring

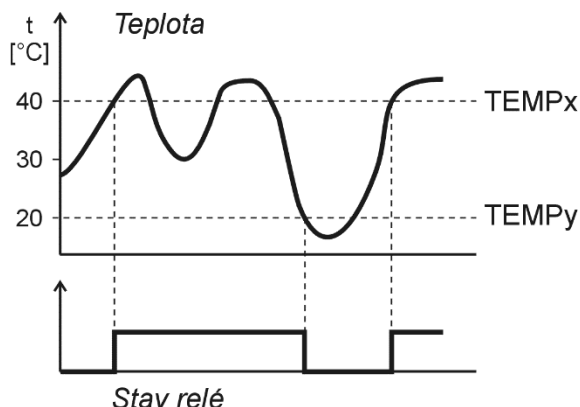
If the temperature exceeds the separate limits set via Telnet before the SMTP section, a trap containing this information is sent:

- 1) User text - text from the *userName* object, to identify the device).
- 2) The current temperature and also the limit that has been exceeded.

TEMPERATURE MONITORING MODES

The following pictures show the different options for setting the temperature monitoring in the Quido modules.¹⁵ The labeling of temperatures and other variables in the figures and text is identical to the labeling of the parameters of the temperature monitoring instructions.

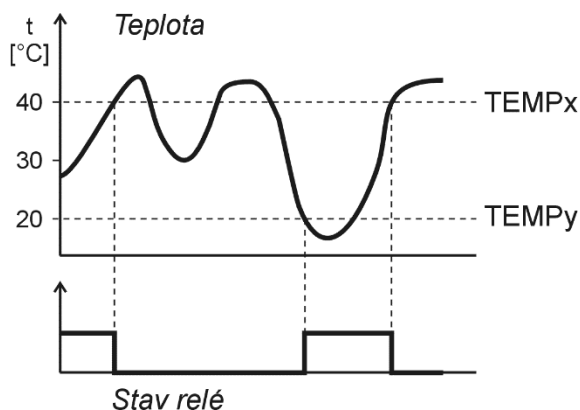
Mode 1



A mode in which the output switches ON when temperature limits are exceeded. The controlled output switches ON when the TEMPx temperature is exceeded and switches OFF when the TEMPy temperature falls below. This introduces hysteresis in the temperature control. (Both temperatures can be set to the same value to cancel the hysteresis.)

The output is normally switched OFF.

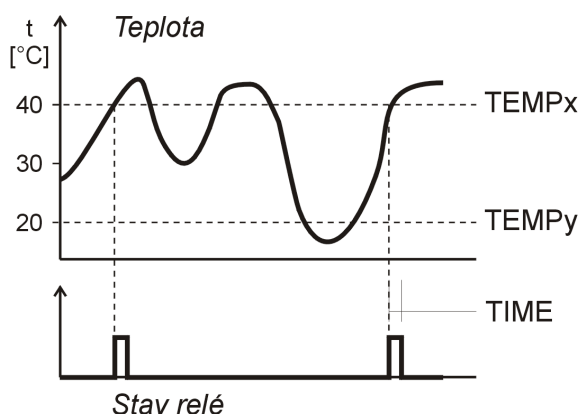
Mode 2



A mode in which the output switches OFF when the temperature limits are exceeded. The controlled output switches OFF when the TEMPx temperature is exceeded and switches ON when the TEMPy temperature falls below. This introduces hysteresis in the temperature control. (Both temperatures can be set to the same value to cancel the hysteresis.)

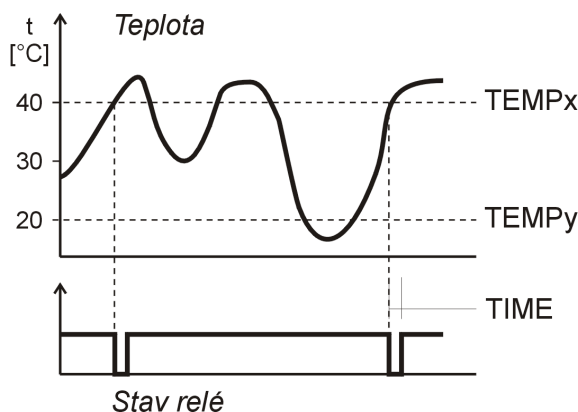
The output is normally switched ON.

¹⁵ Temperature monitoring is only available on modules with at least one thermometer and at least one output.

Mode 3

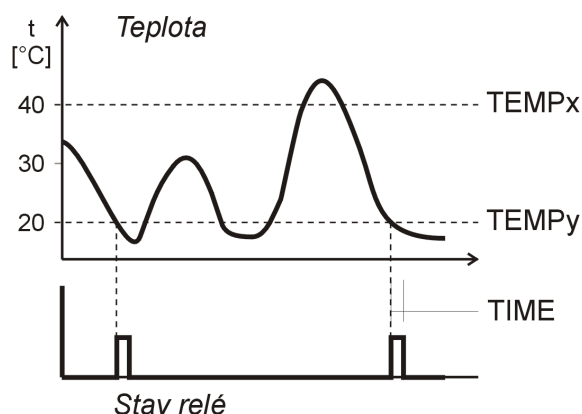
A mode in which the output switches OFF for a set period of time when the temperature limits are exceeded. The controlled output switches ON for a set time when the TEMPx temperature is exceeded. The output can only switch ON again when the temperature falls below TEMPy and then rises again to TEMPx . (Both temperatures can be set to the same value, thus cancelling the protection against frequent switching when the temperature fluctuates around the set limit.)

The output is normally switched OFF.

Mode 4

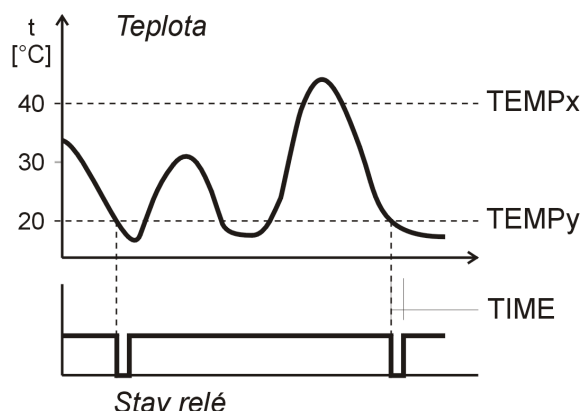
A mode in which the output switches OFF for a set period of time when the temperature limits are exceeded. The controlled output switches OFF for a set time when the TEMPx temperature is exceeded. The output can only switch OFF again when the temperature drops below TEMPy and then rises again to TEMPx . (Both temperatures can be set to the same value, thus cancelling the protection against frequent switching when the temperature fluctuates around the set limit.)

The output is normally switched ON.

Mode 5

A mode in which the output switches ON for a set period of time when it falls below the temperature limits. The controlled output switches ON for a set time when the temperature drops below the TEMPy limits. The output can only switch ON again when the temperature rises above TEMPx and then falls below TEMPy again. (Both temperatures can be set to the same value, thus overriding the protection against frequent switching when the temperature fluctuates around the set limit.)

The output is normally switched OFF.

Mode 6

A mode in which the output switches OFF for a set period of time when it drops below the temperature limits. The controlled output switches OFF for a set period of time when it drops below the TEMPy temperature. The output can only switch off again when the temperature rises above TEMPx and then drops again to TEMPy . (Both temperatures can be set to the same value, thus cancelling the protection against frequent switching when the temperature fluctuates around the set limit.)

The output is normally switched ON.

FAQ

What do I need to set up for Quido to work on my network?

You only need to adjust the Quido network parameters for your network (IP address and Subnet Mask.) The settings are described for Windows OS.

- 1) Connect the Quido to the network and start [Ethernet Configurator](#) program (see Fig. 6).¹⁶
- 2) Click *Add Device* and enter the MAC address of the Quido and the desired IP address.
- 3) Tap *Setup*.
- 4) You can now open the Quido web interface with your internet browser.

How to find out the IP address of Quido?

- 1) The default IP address of Quido is 192.168.1.254. If you have changed the address or cannot connect to Quido at this address, follow these steps.
- 2) Start the [Ethernet Configurator](#) program (see Fig. 6).¹⁶ If the Quido is connected to your network and has a compatible IP address, it will be seen as one device in the *Device Discovery List*.
- 3) If you do not see Quido in the *List*, check that he is connected and assign it a new IP address according to the previous FAQ.

RESET

Use the following procedure to reset the device to its default state. Unlike the reset that can be performed through the web interface (see page 18) or via Telnet (see page 33), the IP address will also be reset to 192.168.1.254.

- 1) Disconnect power to the device.
- 2) Push the RESET button (can also be marked as RST).
- 3) Turn on the power and wait 10 seconds.
- 4) Release the button.
- 5) The device reset process is complete.

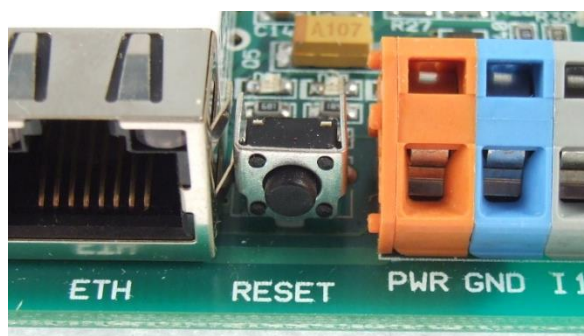


Fig. 30 - Example of reset button on Quido

¹⁶ The programme can be downloaded at www.papouch.com.

INDICATIONS

Power supply

The power indicator (PWR) lights up when the relevant supply voltage is connected.

Communication

In the Web mode: COM indicator is permanently on to indicate proper operation of the equipment.

In other modes: COM indicator flashes to indicate different states depending on the length of the flashes:

- Long flash (0.5 sec): After switching on, when an instruction with a valid address has been received and when sending an automatic message.
- Flash (0.1 sec) every 10 sec, indicates that the internal program in Quido runs correctly.

Inputs

Input LEDs (INx) are lit when the input voltage is applied or the connected contact is closed

Outputs

Output LEDs (OUTx) are lit when the corresponding relay is energized (closed).

Link indicator (Left LED on the Ethernet connector)

No light..... not connected

Yellow connected at 10Mbps

Green..... connected at 100Mbps

Connection type indicator (right LED on the Ethernet connector)

No light..... no communication

Yellow Half-Duplex communication

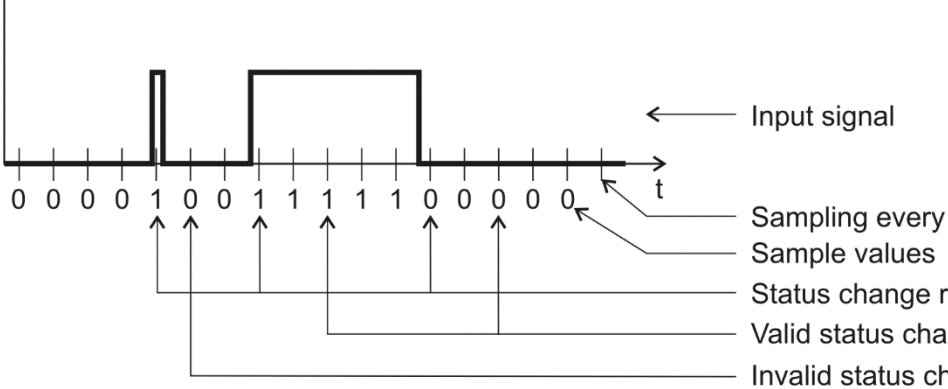
Green..... Full-Duplex communication

TECHNICAL PARAMETERS

Summary

Inputs:

- Number of digital inputs 2 or 4 (depending on type)
- Type of input for ground contact
- Galvanic isolation no
- Sampling from 1 ms (based on settings; see



- Fig. 31; default 20 ms)
- Maximum number of recorded changes in input change counting mode65 535
 - Input current..... 3.3 mA (with contact on)
 - Idle state biasing pull-up resistor 1 kΩ
 - Input voltage for logic "1" 0 - 1 V
 - Maximum input voltage 3,3 V

Outputs:

- Number of digital outputs 16 or 32 (depending on type)
- Type open collector (NPN NO)
- Maximum switching voltage DC 50 V
- Maximum switching current..... 500 mA
- Maximum power loss per output 1 W
- Maximum sum of power loss at outputs 1 to 8 2.25 W
- Maximum sum of power loss at outputs 9 to 16 2.25 W

Thermometer:

- Sensors..... 1
- Sensor type semiconductor
- Measured temperature range..... -55 °C to +125 °C
- Accuracy ±0.5 °C in the range -10 °C to +85 °C; otherwise ± 2 °C
- Temperature drift ±0.2 °C per 1000 hours at 125 °C

The temperature sensor must not be used for:

- Temperature measurement in chemically aggressive environments.
- Temperature measurement in areas with high electrical interference.
- Measuring the temperature of live objects or equipment.

Sensor in shrink wrap enclosure:

Temperature response time..... 6 s

PVC cable to the sensor in the shrink wrap:

Insulation material PVC

Length..... 1, 3, 5, 10 or 15 meters

Operating temperature range -10 to +70 °C

Cable diameter max. 4 mm

Sensor in metal housing:

Temperature time response..... $\tau_{50} = 6 \text{ s}, \tau_{90} = 18 \text{ s}$

Ingress protection IP68 (immersion to a max. depth of 1 m)

Housing material..... stainless steel CSN 17240 (DIN 1.4301)

Housing diameter $5.7 \pm 0.1 \text{ mm}$

Length of the rod 60 mm

Insulation resistance min 200 M Ω at 500 Vdc, at a temperature of 15 to 35 °C and a max. relative humidity of 80 %

Nominal pressure PN 25

Silicone cable to the sensor in a metal housing:

Insulation material silicone rubber, blue

Length..... 1, 3, 5, 10 or 15 metres

Operating temperature range – continuous ...-60 °C to +200 °C

Maximum permissible temperature.....+220 °C

Cable diameter4.3 mm ($\pm 0.1 \text{ mm}$)

Communication interface:

Type.....10/100 Ethernet

ConnectorRJ45

Communication speed of virtual port115200 Bd (fixed)

SMTP port25

Default IP address192.168.1.254

Default netmask.....255.255.255.0

Default gateway IP address.....0.0.0.0

Parameters of Quido ETH 2/16 OC

Current consumption at 12 V (idle)typ. 140 mA

Current consumption at 24 V (idle) typ. 70 mA
 Dimensions 96.5 × 61 × 20 mm
 Weight..... 40 g

Parameters Quido ETH 4/32 OC

Current consumption at 12 V (idle) ... typ. 190 mA
 Current consumption at 24 V (idle) ... typ. 100 mA
 Dimensions 96.5 × 137.4 × 20 mm
 Mounting holes corners of the rectangle 80 × 129 mm (ø 3.2 mm)
 Weight..... 75 g

Other parameters:

Power..... 5 to 30 V, DC, with reverse polarity protection
 Input and output connectors..... PFL connector (2x13pins; 2.54mm pitch)
 Operating temperature -20 C to +70 C
 Ingress protection IP00 (designed for built-in installation)

The principle of evaluating changes to inputs

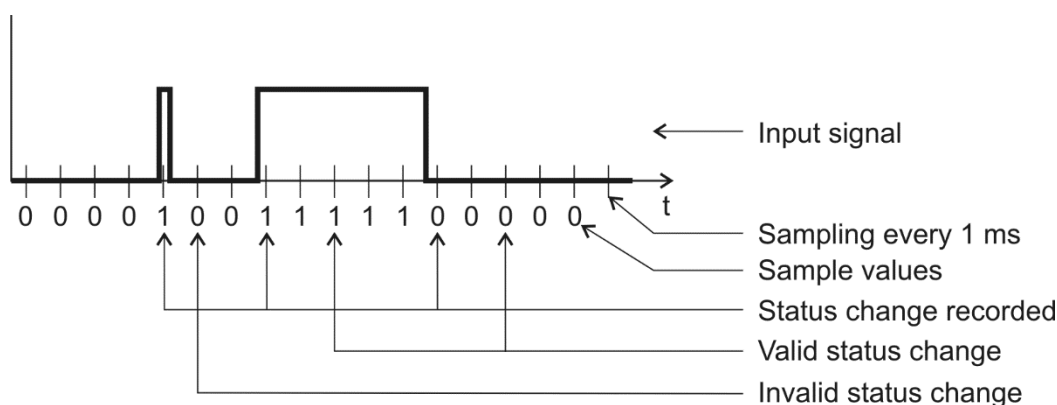


Fig. 31 - principle of evaluating changes on inputs

The input value is sampled with a period of 1 ms. A change in the input is considered valid if the same value is read repeatedly. The number of samples for a valid state change can be changed in the settings, the default number is 20 (i.e. 20 ms).

When a valid change is made, automatic information about the change is sent to the input (if sending is active). If a counter is active on the corresponding input, it is incremented according to its settings.

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 3164/1a
Prague 10**

Phone:

+420 267 314 267

Internet:

www.papouch.com

E-mail:

papouch@papouch.com

