IPS3000 - QUAD I/O
User manual

www.ip-sens.com

Table of Contents

Section 1: Introduction

IPS3000 is a multifunction Ethernet I/O controller. It can be fully configured, programmed and tested using its built-in web server.

Many features are integrated into the IPS3000: Email notification (encrypted or non-encrypted), HTTP post, Dynamic DNS, ThingSpeak cloud and Event scheduling.

The IPS3000 supports a number of Ethernet protocols including HTTP, SNMP, NTP, SMTP, XML and JSON.

1.1. Features

- Easy and fast configuration – full configuration can be performed through the Web using a web browser
- Control and configuration over SNMP v2
- IP security – access is protected by name and password
- 10/100 Mb Ethernet connectivity
- 2 analog inputs
- 4 dry contact digital inputs
- 4 relays with NO contacts; ON/OFF or Pulse modes
- 1-Wire input for up to 5 Temperature & humidity or other IP Sensors
- Removable terminal connectors
- Selectable TCP ports
- HTTP API commands
- XML or JASON(over HTTP)
- Alarm alert – SNMP trap and E-mail
- SMTP with SSL/TLS Encryption
- TLS 1.0, TLS 1.1 and TLS 1.2 support
- E-mails to up 2 recipients like alarm alert
- SNTP synchronization protocol
- Firmware update over IP
- Periodical HTTP Post of XML/JSON files for client-server systems
- Dynamic DNS with support of DynDNS and No-IP services
- ThingSpeak cloud support
- Real-time clock(RTC)
- Event scheduling with single or weekly task
- Wide power supply voltage range

1.2. Applications

IPS3000 can be used for industrial applications like: industrial and building automation, data acquisition systems, environmental monitoring and control of an electrical and non-electrical parameters, general remote control and monitoring.
It works as a standalone device and can be controlled using a web browser as a part of industrial control for SCADA systems.
A few example applications include:

- Environmental monitoring and control of server rooms
- Building management system
- Remotely control and scheduled tasks of motors, pumps, alarms, lights, valves, electric locks
- Industrial cooling and heating control
- Home automation

1.3. Technical information

- Power requirements

  Input Voltage: 10-35VDC / 0.5 A (adapter)

  Input current: 300mA @ 12VDC

  Power Connector: Jack Ø5.5 x 2.0 / 10mm
- Ethernet Interface

  Connector: 8-pin RJ45

  Magnetic Isolation Protection: 1.5 kV (built-in)

  Number of ports: 1

  Speed: 10/100 Mbps, Auto MDIX
- Protocols

  HTTP, DHCP, SNMP v1 and v2, SNTP, SMTP, SMTP TLS, XML, JASON
- Operating conditions

  Operating temperature: -20 to +60 °C

  Operating Relative Humidity: 5 to 85% (non-condensing)
- Digital inputs

  Number: 4

  Type: Dry contact

  Isolation: Non isolated

  Dry Contact Level: Logic “0”: Short to GND

  Logic “1”: Open
- Analog inputs

  Number: 2

  Type: Single ended

  Input range: 0 to 60 VDC

  Isolation: Non isolated

  Resolution: 10-bit

  Accuracy: ±1%

  Sampling Rate: 500mS per channel
- Relay outputs

  Number: 4

  Contact type: Form A (N.O. contact)

  Max. switched current: 3A

  Max. switched voltage: 30VDC

  Insulation Resistance: 1000 mega-ohms 500 VDC at 20°C, 50% RH

Mechanical Life Expectancy: 5 000 000 operations

Electrical Life Expectancy: 120 000 operations

- 1-Wire interface

  RJ11 for connecting up to 5 sensors

  Output voltage: 5.0 ± 0.3 VDC

  Maximum output current: 0.2 A

- Physical Characteristics

  Housing: ABS enclosure / wall mount (DIN-Rail option)

  Weight: 160 g

  Dimensions: 125 x 81 x 32 [mm]

- Warranty – 3 years

Section 2: Installation and setup

2.1. Installation

- IPS3000 controller must be installed by qualified personnel.
- Controller must not be installed in unprotected outdoor locations.
- IPS3000 must not be used for medical, life saving purposes, or for any purpose where its failure could cause serious injury or the loss of life.
- This unit must not be used in any way where it's function or failure could cause significant loss or property damage.

2.2. Mounting

IPS3000 can be wall mounted. Ventilation is recommended for installations where ambient air temperatures are expected to be high.

2.3. Connecting

It is recommended to test and configure IPS3000 without any controlled device.

The correct wiring procedure:

- Make sure power is turned off
- Make wiring connections to the terminals
- Apply power supply

2.4. Connectors and indicators

2.4.1. Power connector – 12V

Jack Ø5.5 x 2.1 / 10mm, central positive

The power supply adapter for IPS3000 shall be resistant to short circuit and overload in a secondary circuit and in compliance with safety requirements.

2.4.2. Network connector -ETH

RJ-45 Ethernet connector. The green LINK LED is illuminated when the module is properly connected to an Ethernet network and is ready to communicate. The green LINK LED blinks when activity is detected on the network. The yellow 10/100 speed LED is illuminated when the network speed is 100Mbps.

The Ethernet connector support Auto MDI-X which automatically detects the required cable connection type and configures the connection appropriately. The IPS3000 can be connected to a switch, router or a PC with a TP patch cable.

2.4.3. Sensor connector- 1-Wire

RJ11 connector for connecting up to 5x 1-Wire sensors. Controller support temperature and temperature and humidity sensors. The sensors are connected with three wires – +5V out, ground (GND) and bidirectional data (Data).

Many parameters determine the maximum length of the wires – the type of cable, the number of sensors, ambient electromagnetic noise and sensor network topology. It is strongly recommended to use “daisy-chained” (linear topology) for multiple sensors.

When used UTP cable, total cable length of all connected sensors can be max. 50m.

2.4.4. Analog Input connector – AIN1, GND, AIN2

1x 3-position plug-in terminal for connecting up to 2 analog inputs to monitor of DC voltage up to 60VDC.

Analog inputs are not galvanic isolated!

2.4.5. Digital Input connectors – DIN1, GND, DIN2, DIN3, GND, DIN4

2x 3-position plug-in terminals for connecting up to 4 dry contact(volt free) inputs.

Digital Inputs can be used for monitoring sensors in “dry contact”(volt free) mode - motion PIR sensors, Water Leak sensors, Water level sensors, Door/window sensors, Smoke detectors, AC voltage detection sensors.

One side of the sensor is connected to DIN1 -:- DIN4 terminal and the other is connected to GND terminal.

Digital inputs are not galvanic isolated!

2.4.6. Relay output connectors

Relay 1 – NO1, COM1

Relay 2 – NO2, COM2

Relay 3 – NO3, COM3

Relay 4 – NO4, COM4

4x 2-position plug-in terminals to which relay contacts are directly connected.

Normal open(NO) and common(COM) contacts are available.

For loads greater than 3A/30V DC an external relay should be used.

2.4.7. LED indicators

- **PWR** - green led indicator. The Led shines in working mode.
- **STS** - red led indicator. The Led flashes in working mode.
- **R1-:-R4 -** green led indicators. The Led’s shines when the relays are activated.

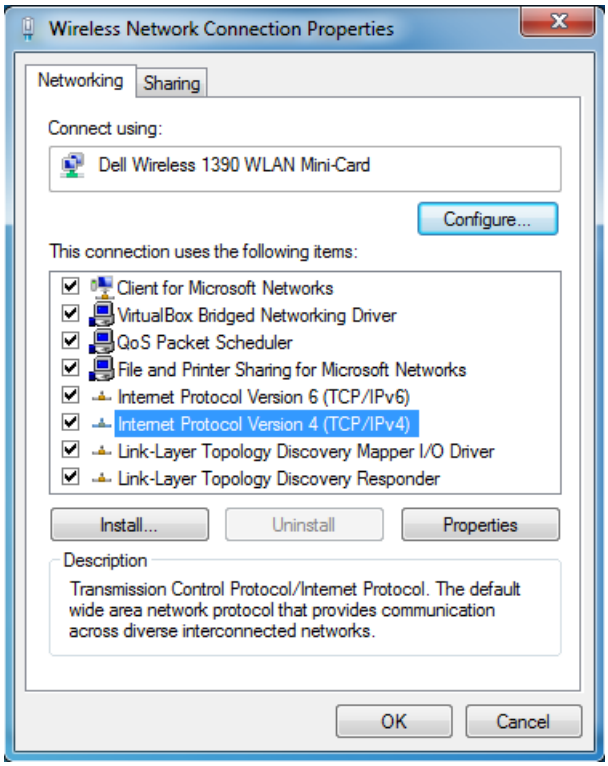
2.5. Configuration and setup

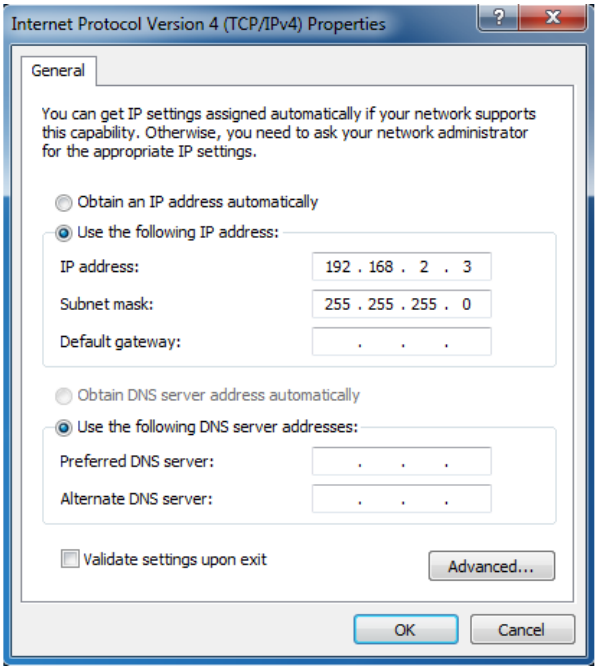
When configuring the IPS3000 using its built-in web server the controller and computer must be addressed on the same network.

By default, factory IP address of IPS3000 is 192.168.2.2. Communication with the IPS3000 may be established by assigning an IP address to the PC such that it is on the same network as the IPS3000 (for example, PC could be assigned to 192.168.2.3)

Free software “Ethernet Device Discoverer” for Windows can be downloaded from

www.ip-sens.com to easily found IPS3000 controllers connected to LAN.





Section 3: Web interface

IPS3000 can be configured, monitored and controlled through the web interface.

The controller support only HTTP.

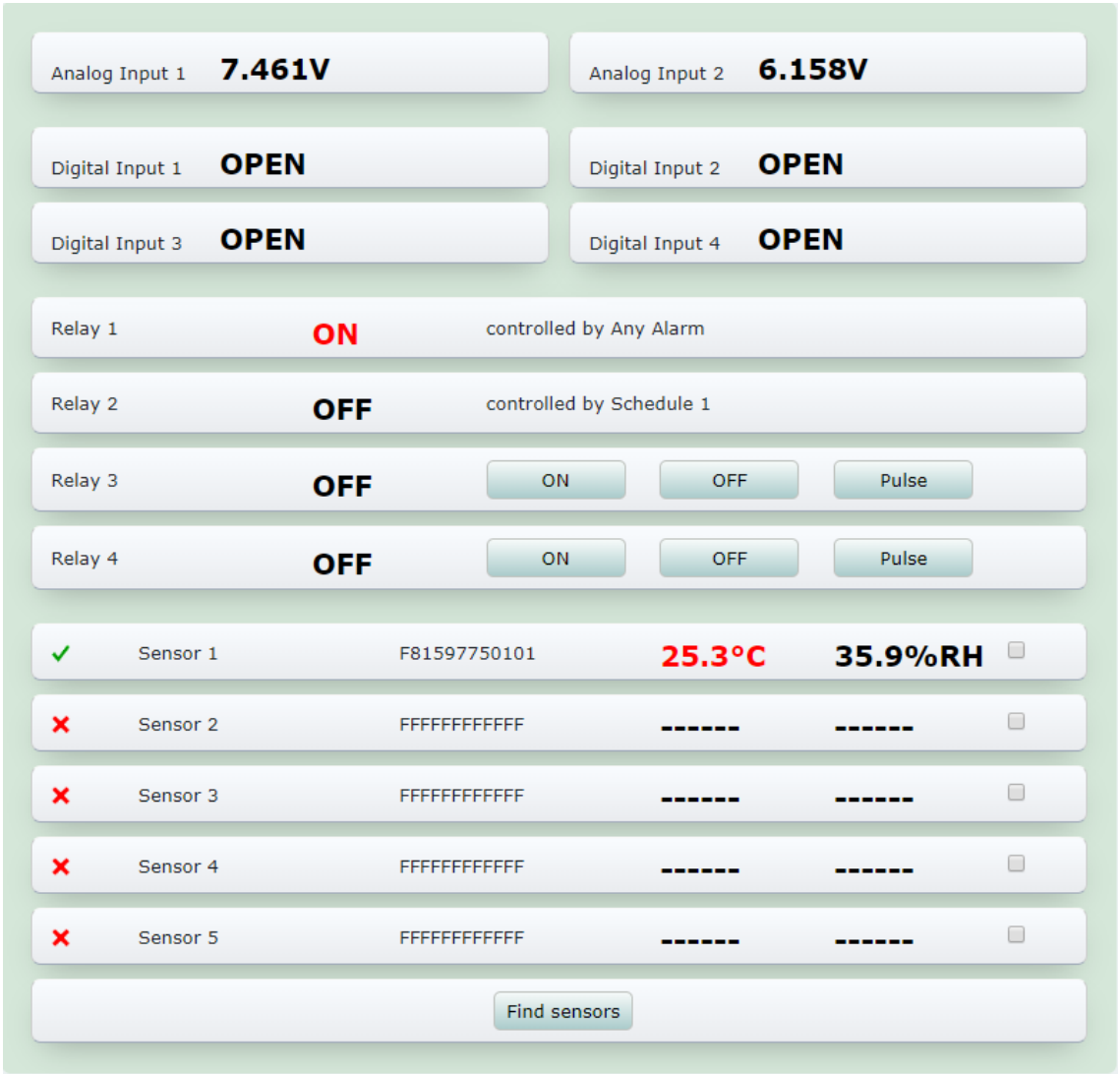
Web interface is created with modern technologies that allow it to change and adjust according to the size of the screen of the device - monitors, laptops, tablets and smartphones.

This makes it easy to work through a web browser even on a smartphone.

3.1. Login page

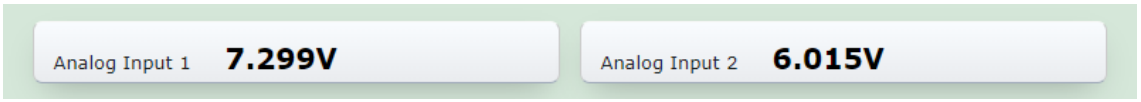
Access to the IPS3000 web server is protected by name and password.

3.2. Monitor and control page



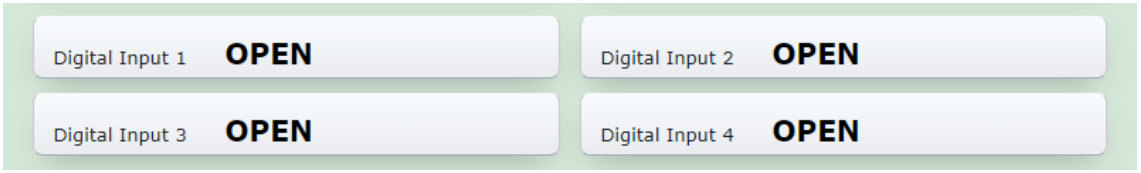
Monitoring and control page has 4 sections – “Analog inputs” , “Digital inputs”, “Outputs” and “Sensors”. All they can be displayed/hidden from monitoring page independently - see “General Setup->System->Display”.

- **Analog Inputs section** can be used for monitoring directly batteries, power supplies, solar panels and any analog sensor with voltage up to 10V DC.



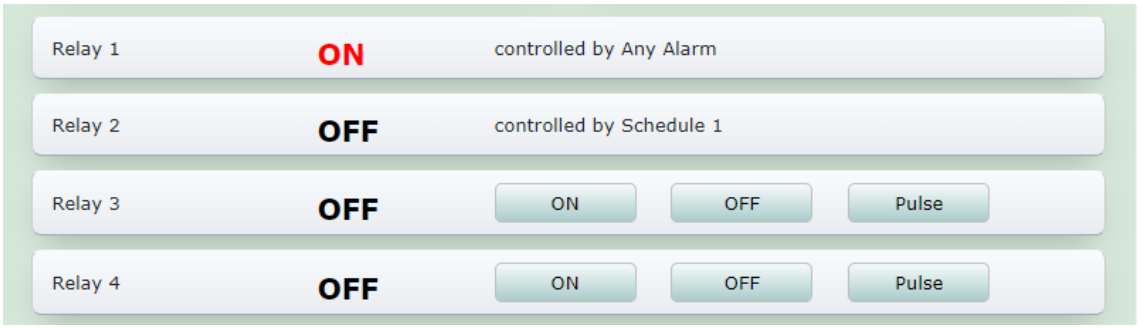
  In section I/O Setup->Analog inputs for every analog input can be set 6 variables – “Unit”, “Lower range”, “Upper range”, “Hysteresis”, “Multiplier” and “Offset”.

- **Digital Inputs section** can be used for monitoring sensors in “dry contact”(volt free) mode - motion PIR sensors, Water Leak sensors, Water level sensors, Door/window sensors, Smoke detectors, AC voltage detection sensors.



  Name, Low and High level descriptions can be changed on “I/O Setup->Digital Inputs” web page.

- **Outputs section** displays the relays current state. Through buttons can be changed the relays status.



  Each relay can be activated by the WEB interface, HTTP API, SNMP or from the status of a 1-Wire sensor, analog voltage, dry contact or scheduled task.

  Every relay has buttons “On”, “Off” and “Pulse”.

  When the relay is controlled from parameter, description of this parameter is displayed rather than button.

  From “I/O Setup->Local Relay outputs” can be setup Name descriptions, Relay control and Pulse duration.

- **Sensors section** can be used for monitoring up to 5x 1-Wire sensors for temperature and humidity.

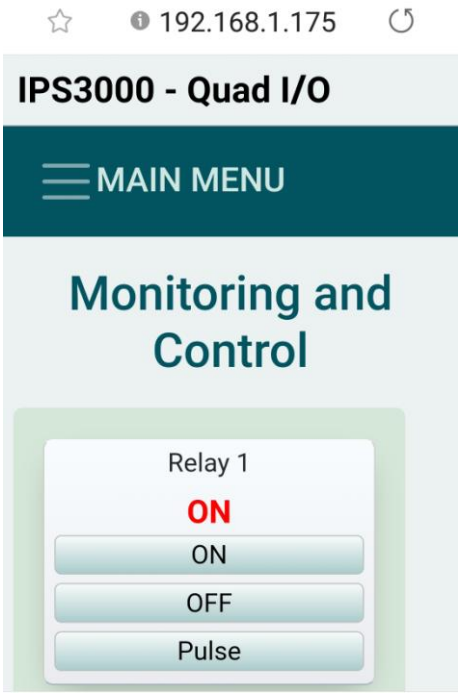
Sensors can be find trough button “Find sensors” or after power on.

The information for each sensor includes a description, ID and value. Temperature and humidity sensor STH102 has 2 value parameters – the left column is parameter temperature and the right column is humidity.

To check the sensor in a specific position, check the box to the right of the sensor.

For every sensor can be setup Name, Lower Range, Upper Range and Hysteresis on the page “I/O Setup->Sensors”. The number of sensors to be displayed on the monitoring page can also be setup on the page I/O Setup->Sensors”.

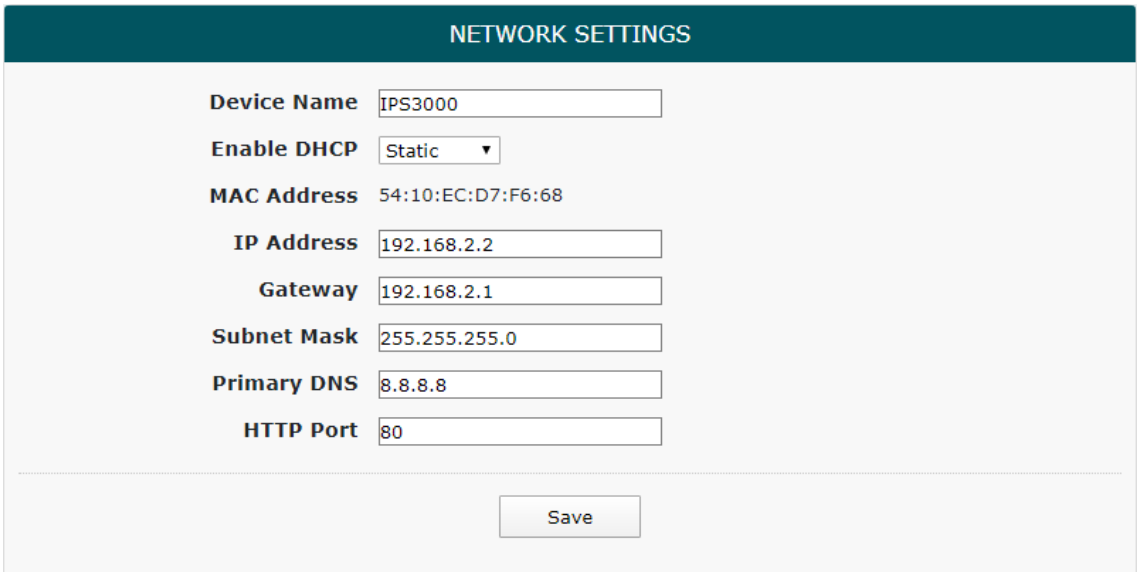
State-of-the-art technologies have been used to allow the controller web site to automatically resize according to the screen. Below is a screenshot from the smartphone web browser.



3.3. General Setup

3.3.1. Network page

On this page can be configured the network settings to make the IPS3000 controller accessible on your network.



Device Name

Device name length is up to 15 characters. This name is shown in our software tool Ethernet Device Discoverer when searched.

Enable DHCP (Dynamic Host Control Protocol)

The default setting is “Static”. In this case the IP address is fixed and does not change dynamically.

If DHCP is enabled, the IPS3000 will request an IP address from the DHCP server each time it is powered on.

IP Address

The IP address is specific to the network where the controller will be installed and must be obtained from the network administrator. The default setting is: 192.168.2.2

Gateway

This specifies the IP address of the gateway router. This can be obtained from the network administrator. The default setting for this field is: 192.168.2.1

Subnet Mask

The subnet mask defines the size of the local network. This can be obtained from the network administrator. The default setting is: 255.255.255.0

Primary DNS

The IP address of the Primary DNS server. It is recommended to use public DNS server rather than gateway. The default setting is: 8.8.8.8.

Without a correctly configured DNS server, the following functions will not work:

- the time synchronization (SNTP), used in emails and SNMP traps
- the email services(SMTP)

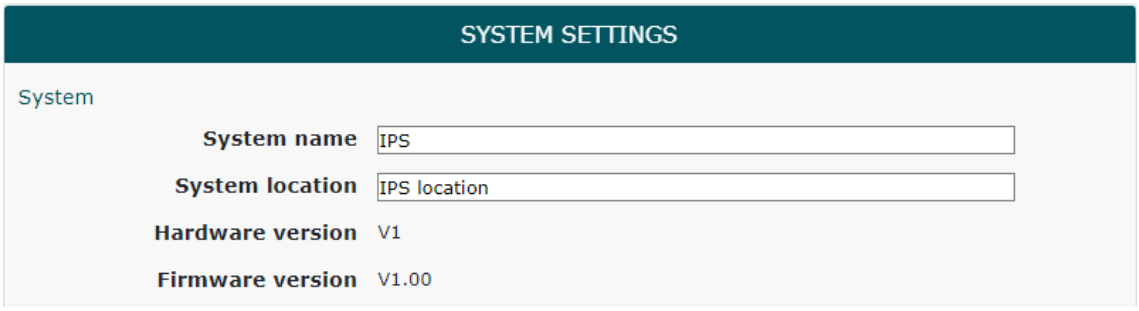
HTTP port

The TCP port used for unencrypted HTTP communication with the controller. The default setting for this port is 80. Value of HTTP port can be changed from 80 to 65500.

3.3.2. System page

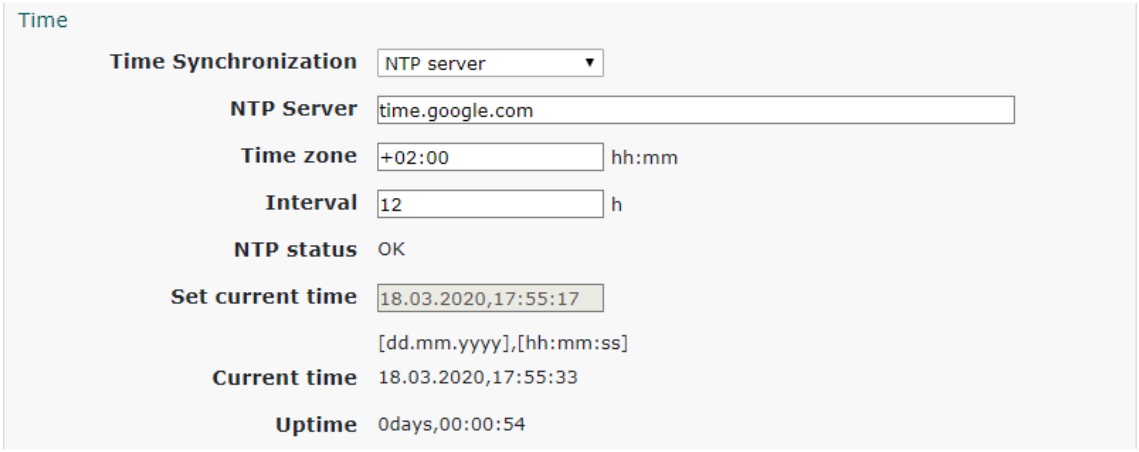
3.3.2.1. System section

In this section can be setup parameters for identification of device.



3.3.2.2. Time section

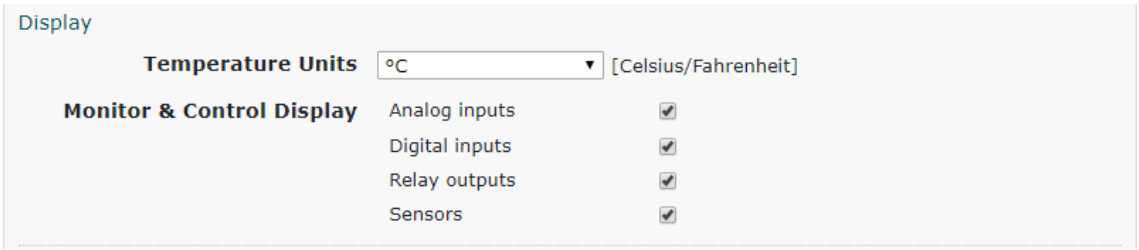
In this section can be set internal Real Time Clock trough automatic synchronization by NTP protocol or manual. NTP synchronization is disabled by default , NTP server – time.google.com, Time zone +00:00 and synchronization interval - 12 hours.



3.3.2.3. Display section

In this section can be setup Celsius or Fahrenheit temperature units for all observed temperatures.

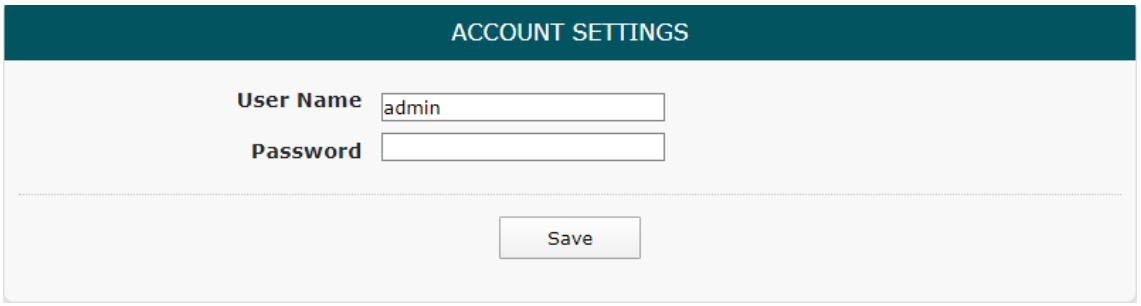
All sections which can be displayed on the Monitor & Control page can setup here.



3.3.3. Account page

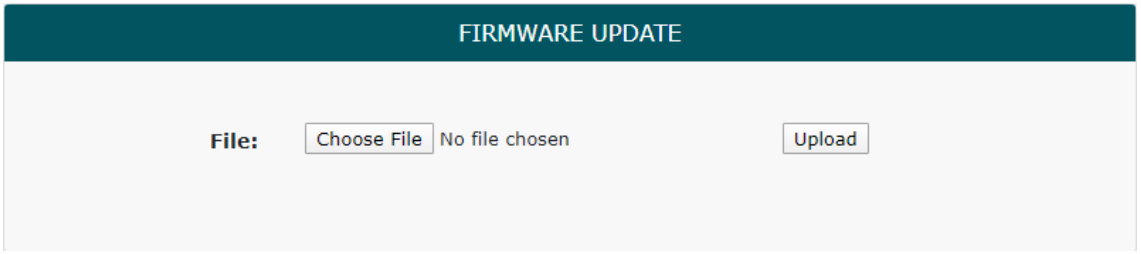
The IPS3000 supports one user with administrative rights.

The username and password length is up to 15 characters long.



3.3.4. Update page

Firmware update of IPS3000 can be done by this page.

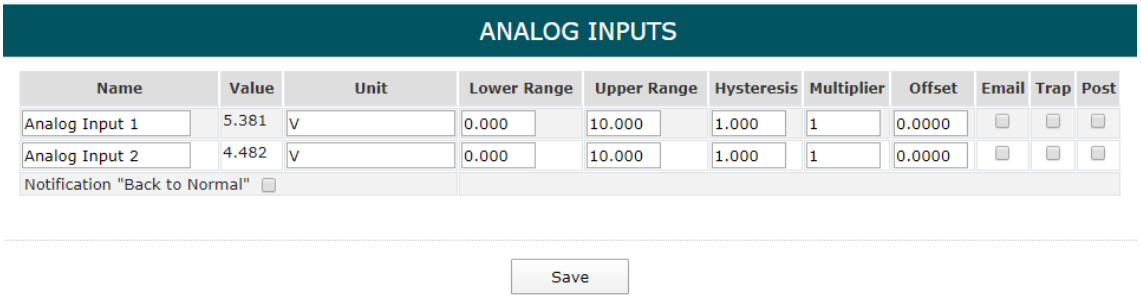


3.4. I/O Setup

3.4.1. Analog inputs

IPS3000 has 2 analog inputs. For every analog input can be configured Name, Units, Lower Range, Upper Range, Hysteresis, Multiplier and Offset.

The Name can be long up to 15 chars.

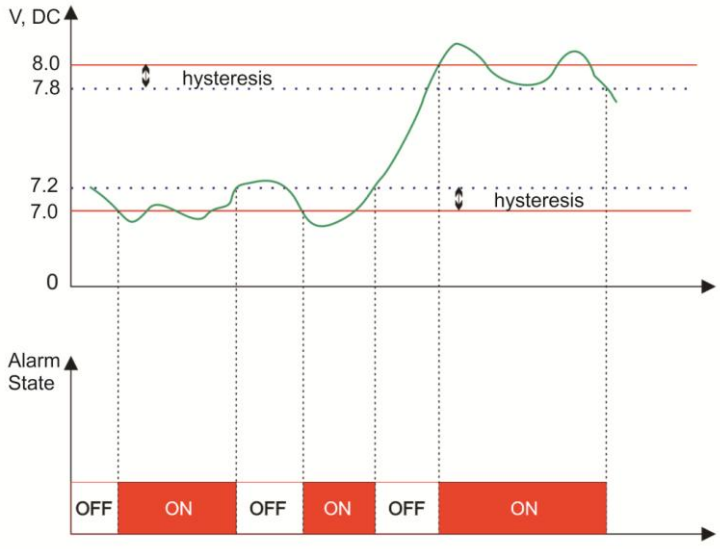


When a sensor with analog output is connected to analog input, the Units, Multiplier and Offset can be used to convert measured raw voltage in a sensor value.

The Email, Trap and Post checkboxes can be enabled to generate Notification when parameter is in alarm condition.

The checkbox “Back to Normal” can be enabled to generate notification when analog value returns in the range.

Use hysteresis value to avoid numerous false alerts whenever the reading value fluctuates around the threshold!

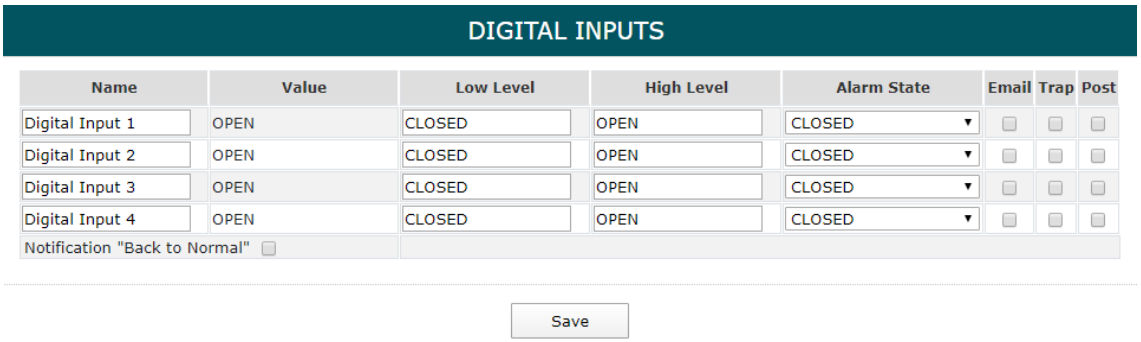


3.4.2. Digital inputs

For digital inputs can be setup Name, Low level, High level and Alarm state.

The Email, Trap and Post checkboxes can be enabled to generate notification when parameter is in alarm condition.

The checkbox “Back to Normal” can be enabled to generate notification when digital value returns in the range.



DIGITAL INPUTS

Name	Value	Low Level	High Level	Alarm State	Email	Trap	Post
Digital Input 1	OPEN	CLOSED	OPEN	CLOSED			
Digital Input 2	OPEN	CLOSED	OPEN	CLOSED			
Digital Input 3	OPEN	CLOSED	OPEN	CLOSED			
Digital Input 4	OPEN	CLOSED	OPEN	CLOSED			
Notification "Back to Normal"							

Save

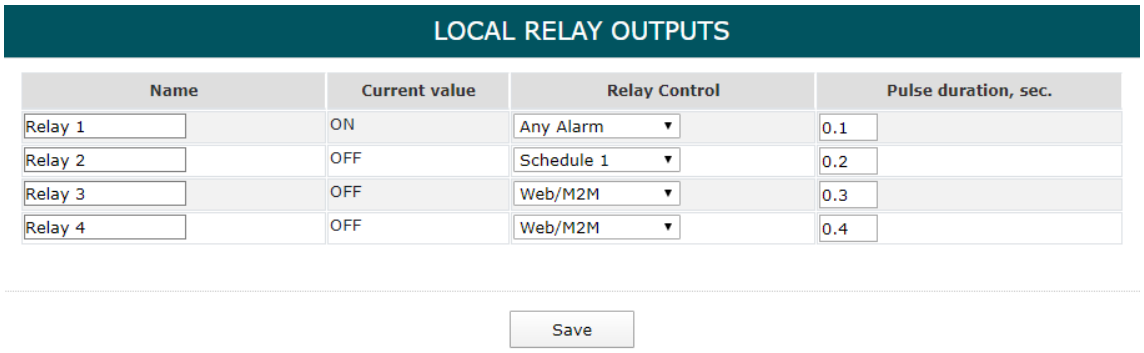
3.4.3. Local Relay Outputs

From the Local Relay Output settings can select whether the local relay to be activated through Web/M2M or automatic from the monitored parameter. By default the relays are activated from the web interface of the Monitor & Control page.

Only one parameter can be assigned for relay activation, at the same time.

When the relay is triggered automatically by a parameter then the button for Web/M2M operation on the Monitor & Control page disappears.

For every relay control can be set different time for pulse duration. The resolution is 0.1 second.



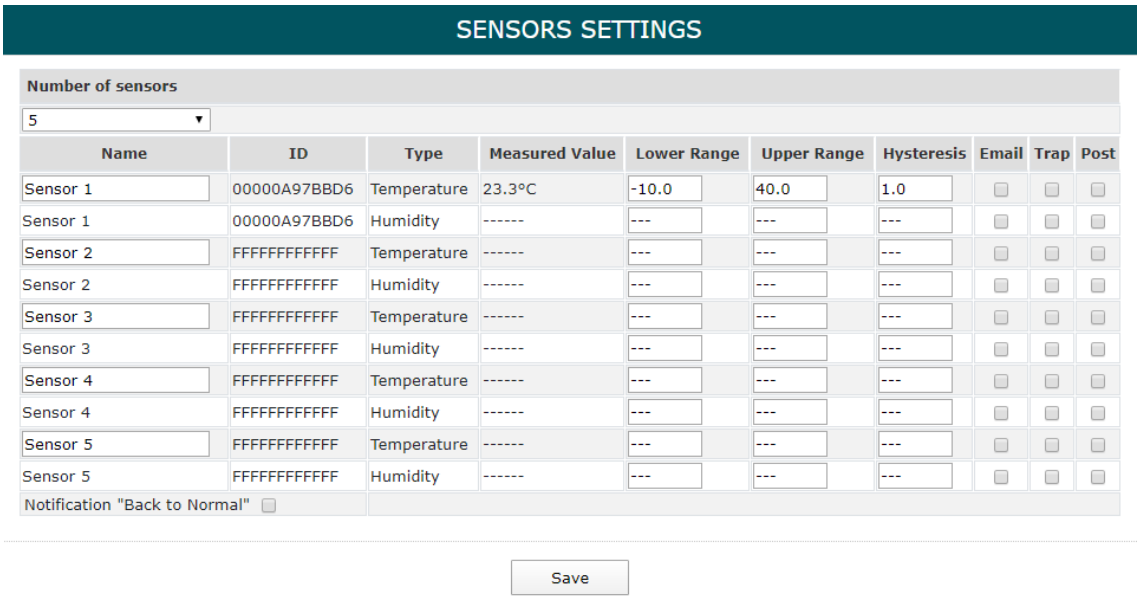
LOCAL RELAY OUTPUTS

Name	Current value	Relay Control	Pulse duration, sec.
Relay 1	ON	Any Alarm	0.1
Relay 2	OFF	Schedule 1	0.2
Relay 3	OFF	Web/M2M	0.3
Relay 4	OFF	Web/M2M	0.4

Save

3.4.4. Sensors

For every sensor can be setup Name, Lower Range, Upper Range, Hysteresis, Email and Trap alert. The number of sensors to be displayed on the monitoring page can also be setup from this page.



SENSORS SETTINGS

Number of sensors: 5

Name	ID	Type	Measured Value	Lower Range	Upper Range	Hysteresis	Email	Trap	Post
Sensor 1	00000A97BBD6	Temperature	23.3°C	-10.0	40.0	1.0			
Sensor 1	00000A97BBD6	Humidity	------	---	---	---			
Sensor 2	FFFFFFFFFFFF	Temperature	------	---	---	---			
Sensor 2	FFFFFFFFFFFF	Humidity	------	---	---	---			
Sensor 3	FFFFFFFFFFFF	Temperature	------	---	---	---			
Sensor 3	FFFFFFFFFFFF	Humidity	------	---	---	---			
Sensor 4	FFFFFFFFFFFF	Temperature	------	---	---	---			
Sensor 4	FFFFFFFFFFFF	Humidity	------	---	---	---			
Sensor 5	FFFFFFFFFFFF	Temperature	------	---	---	---			
Sensor 5	FFFFFFFFFFFF	Humidity	------	---	---	---			
Notification "Back to Normal"									

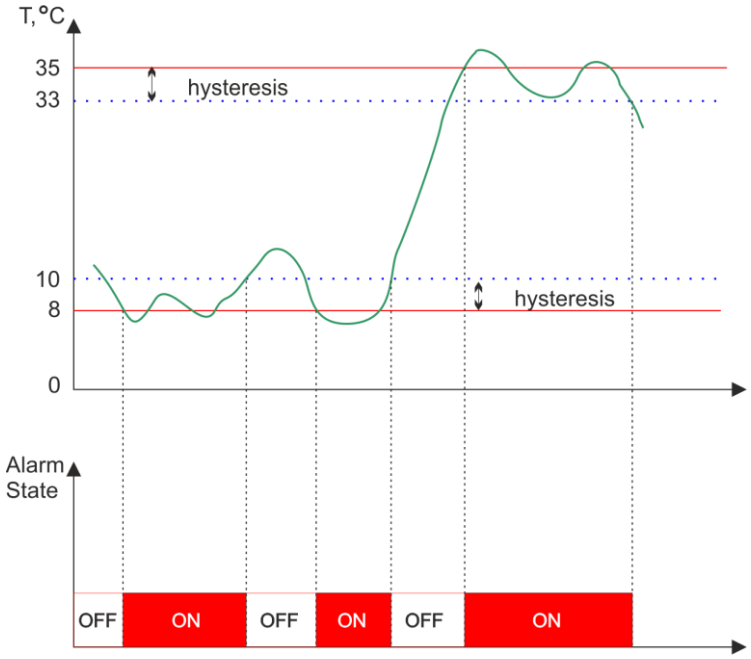
Save

The Email, Trap and Post checkboxes can be enabled to generate notification when parameter is in alarm condition.

The checkbox “Back to Normal” can be enabled to generate notification when digital value returns in the range.

The working range for the observed parameter can be setup through Lower, Upper Range and Hysteresis.

Hysteresis is used to avoid multiple alarms when the measured value fluctuates around the threshold!.



3.5. Services

3.5.1. SNMP

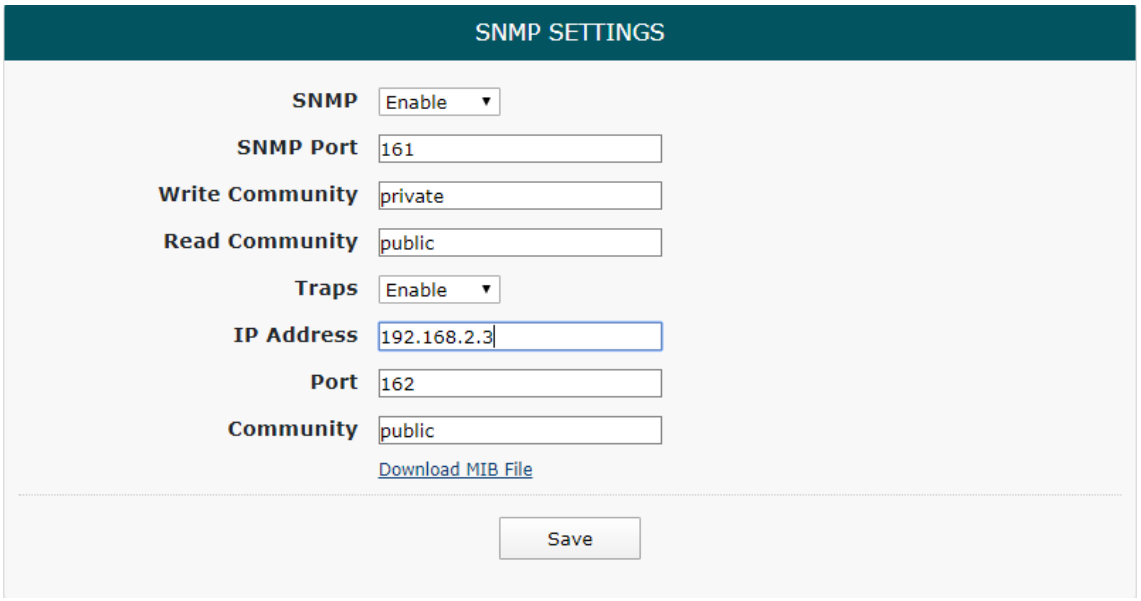
The IPS3000 can be configured and monitored through SNMP.

Controller supports SNMP V2 protocol. By default SNMP is disabled, the port is 161, write community is “private” and read community is “public”.

Traps are sent in following conditions:

- when measured values on analog inputs goes outside the range
- when dry contact inputs changed their status
- when measured sensor values goes outside the range

MIB file is necessary for management and control from SNMP. MIB file can be downloading from this page by click on the link “Download MIB file”.



3.5.2. Email

On this page can be make the Email settings.

The IPS3000 send Email alerts based on any sensor or input, such as temperature, humidity, digital input or analog input.

SMTP server address can be set by name or IP address.

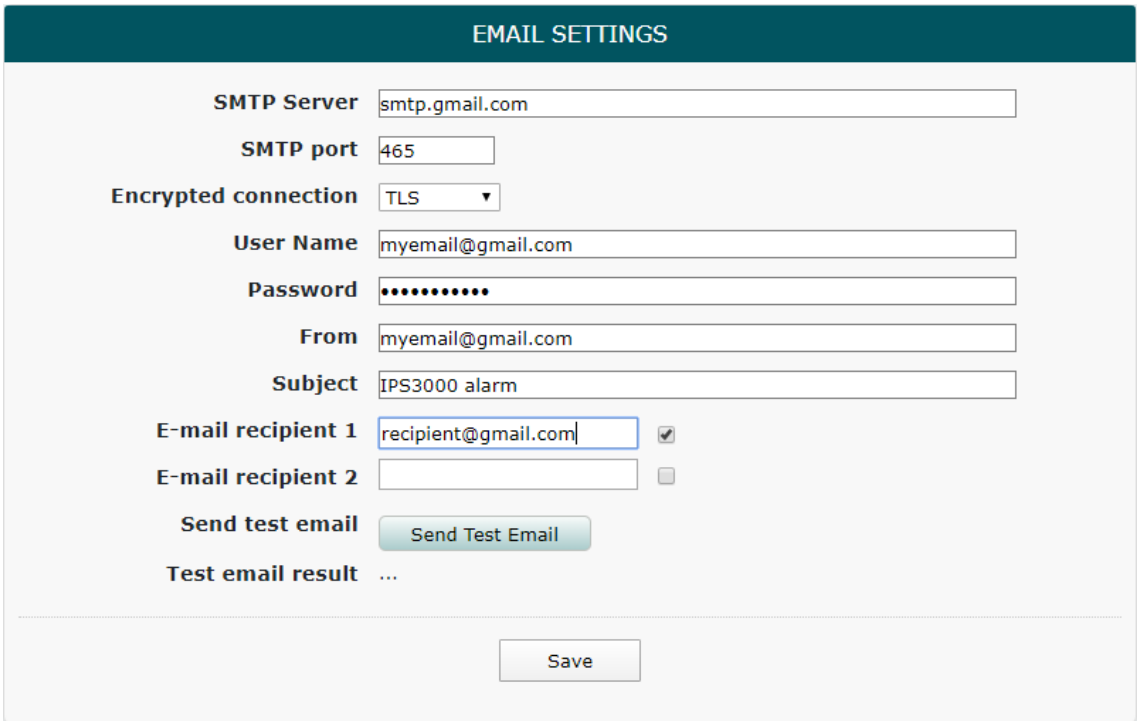
IPS3000 supports TLS 1.0, TLS 1.1 and TLS 1.2 encryption.

This ensures successful send emails with most of the public email servers.

When TLS encrypted connection is selected, the connection to the SMTP server will be securely. TLS generally requires the use of port 465.

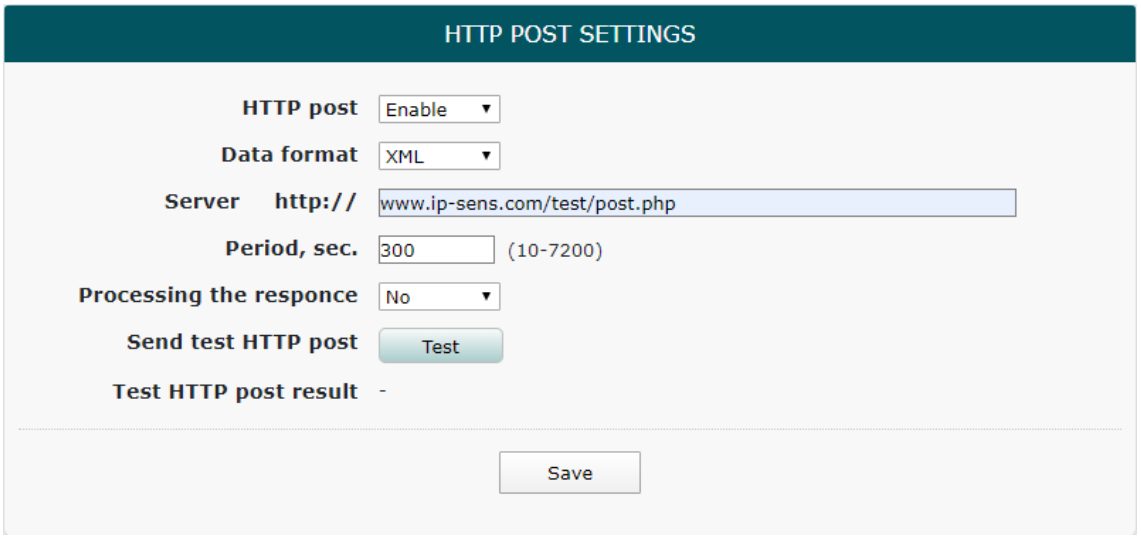
By default SMTP port is 25, without encrypted connection, Sender e-mail, username and password are standard authentication details.

There is a button for test email settings with a feedback.



3.5.3. HTTP Post

The controller IPS3000 can be setup on this page to send XML/JSON file to dedicated server through HTTP Post. The post period is between 10 and 7200 seconds.

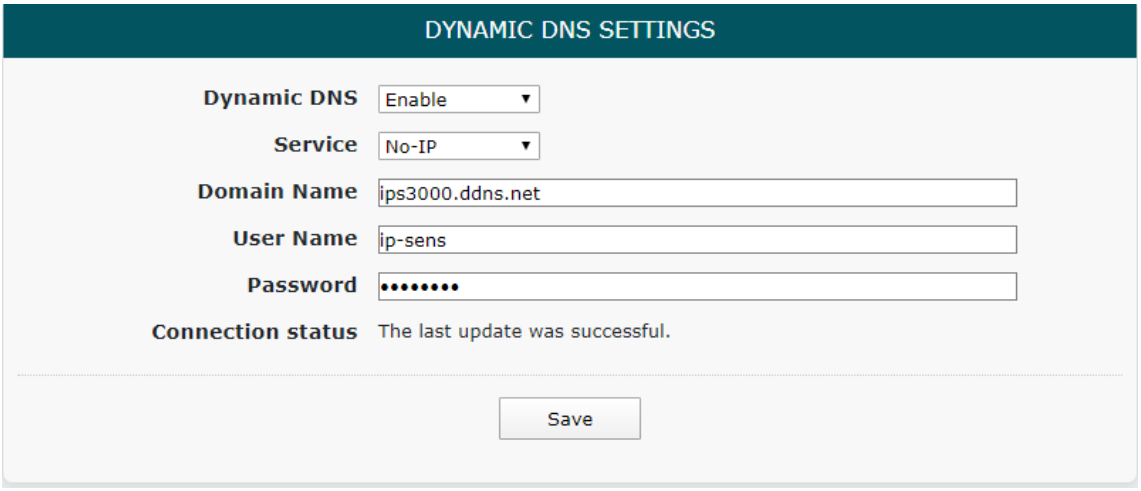


When option “Processing the response” is enabled, the controller will process the response HTTP API commands from the remote server.

3.5.4. DynDNS

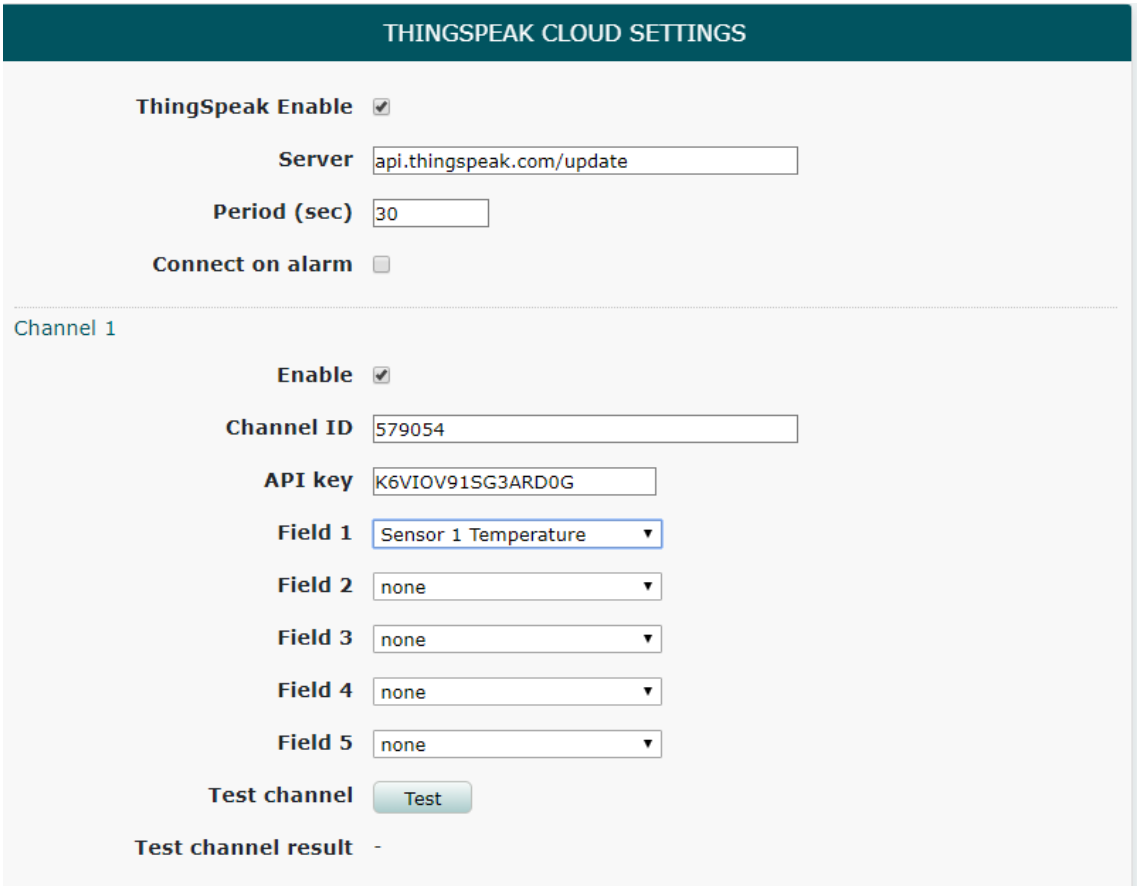
Dynamic DNS service lets you assign a fixed host and domain name to a dynamic Internet IP address.

IPS3000 supports Dynamic DNS services from DynDNS and No-IP.



3.5.5. ThingSpeak

ThingSpeak is analytics platform service that allows you to aggregate, visualize and analyze live data streams in the cloud. Platform provides instant visualizations of data posted by IoT devices to ThingSpeak.



ThingSpeak Features:

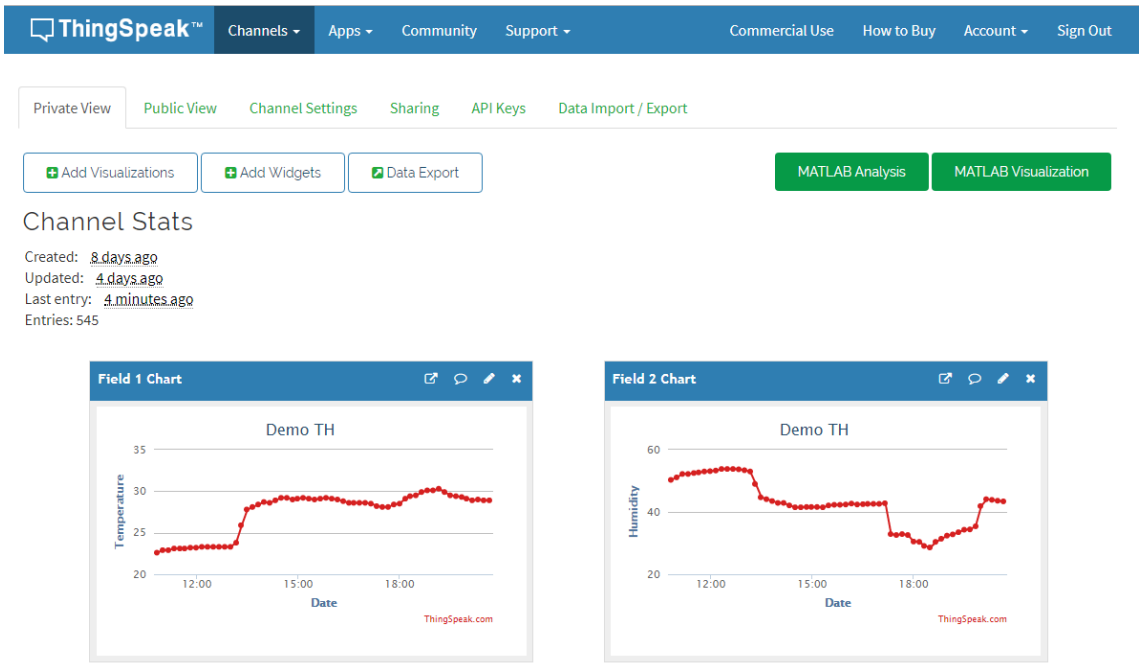
- Collect data in private channels
- Share data with public channels
- Visualize your sensor data in real-time as charts.
- Prototype and build IoT systems without setting up servers or developing web software
- MATLAB® analytics and visualizations
- Smartphone client applications

The main element of ThingSpeak is the channel.

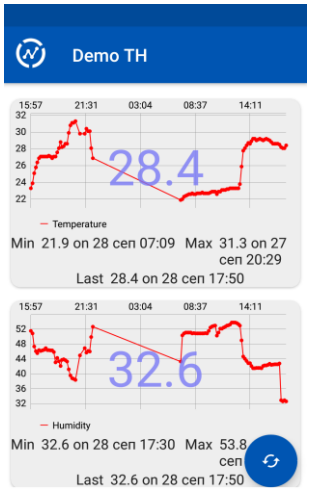
Every channel contains channel ID, API key and Fields.

Controller IPS3000 support 2 channels and 5 fields for every channel. Every field can be related with any sensor, analog input, digital input or relay.

Bellow is an example screenshot with charts for one ThingSpeak channel captured from website:



Bellow is ann example screenshot with charts for one ThingSpeak channel captured from mobile application ThingView:



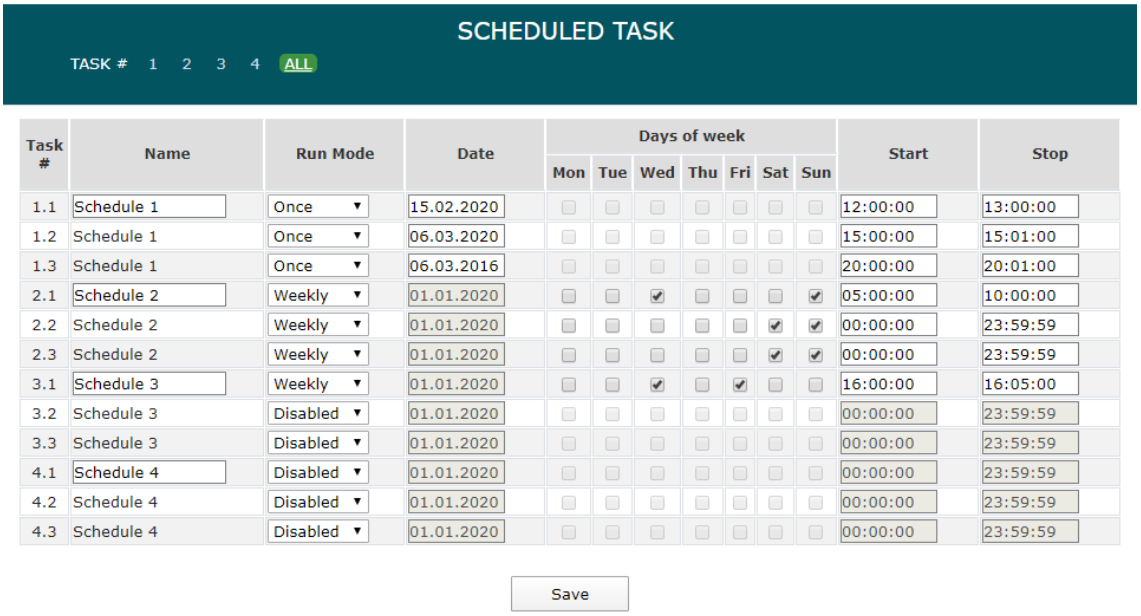
3.5.6. Scheduled Task

Scheduled tasks run at a specific time and on specific days of the week and are used to control the relays.

Each Scheduled task can initiate up to three Actions.

Every action has two Run modes:

- Single task for a time period –> Run mode = Once
- Weekly task for a time period -> Run mode = Weekly



Task #	Name	Run Mode	Date	Days of week							Start	Stop
				Mon	Tue	Wed	Thu	Fri	Sat	Sun		
1.1	Schedule 1	Once	15.02.2020	☐	☐	☐	☐	☐	☐	☐	12:00:00	13:00:00
1.2	Schedule 1	Once	06.03.2020	☐	☐	☐	☐	☐	☐	☐	15:00:00	15:01:00
1.3	Schedule 1	Once	06.03.2016	☐	☐	☐	☐	☐	☐	☐	20:00:00	20:01:00
2.1	Schedule 2	Weekly	01.01.2020	☐	☐	☑	☐	☐	☐	☑	05:00:00	10:00:00
2.2	Schedule 2	Weekly	01.01.2020	☐	☐	☐	☐	☐	☑	☑	00:00:00	23:59:59
2.3	Schedule 2	Weekly	01.01.2020	☐	☐	☐	☐	☐	☑	☑	00:00:00	23:59:59
3.1	Schedule 3	Weekly	01.01.2020	☐	☐	☑	☐	☑	☐	☐	16:00:00	16:05:00
3.2	Schedule 3	Disabled	01.01.2020	☐	☐	☐	☐	☐	☐	☐	00:00:00	23:59:59
3.3	Schedule 3	Disabled	01.01.2020	☐	☐	☐	☐	☐	☐	☐	00:00:00	23:59:59
4.1	Schedule 4	Disabled	01.01.2020	☐	☐	☐	☐	☐	☐	☐	00:00:00	23:59:59
4.2	Schedule 4	Disabled	01.01.2020	☐	☐	☐	☐	☐	☐	☐	00:00:00	23:59:59
4.3	Schedule 4	Disabled	01.01.2020	☐	☐	☐	☐	☐	☐	☐	00:00:00	23:59:59

Save

Example to set Scheduled tasks to operate at Once.

Task #	Name	Run Mode	Date	Days of week							Start	Stop
				Mon	Tue	Wed	Thu	Fri	Sat	Sun		
1.1	Schedule 1	Once	10.04.2020	☐	☐	☐	☐	☐	☐	☐	08:00:00	08:30:00

Example to set Scheduled tasks to operate Weekly.

Task #	Name	Run Mode	Date	Days of week							Start	Stop
				Mon	Tue	Wed	Thu	Fri	Sat	Sun		
1.1	Schedule 1	Weekly	01.01.2020	☐	☐	☐	☐	☐	☑	☑	10:00:00	13:00:00

Section 4: Protocols and API

4.1. HTTP API

4.1.1. HTTP Get

Custom computer applications may be created to monitor and control the IPS3000. This method does not use a web browser.

HTTP Get can be used to monitor IPS3000 via XML or JSON files. The format is:

- http:// device_ip_address /status.xml
- http:// device_ip_address /status.json

HTTP API commands can be used to control the relays. Before that is required basic authentication .

HTTP command	**Description**
http://device_ip_address /status.xml?r**x**=1	Relay **x** = ON, where **x** is 1,2,3,4
http://device_ip_address /status.xml?r**x**=0	Relay **x** = OFF, where **x** is 1,2,3,4
http://device_ip_address /status.xml?tg**x**=1	Toggle Relay **x** state, where **x** is 1,2,3,4
http://device_ip_address /status.xml?p**x**=1	Pulse Relay **x,** where **x** is 1,2,3,4

Example to send HTTP API command at the same time as authentication, where user=admin and password=admin:

http://device_ip_address /status.xml?a=admin:admin&r1=1

4.1.1.1. Status XML structure

```
<Monitor>
    <Device>IPS3000</Device>
    <HostName>IPS3000</HostName>
    <MAC>54:10:EC:D7:F6:68</MAC>
    <SysName>IPS</SysName>
    <Location>IPS location</Location>
    <aIn1>
       <aIn1_name>Analog Input 1</aIn1_name>
       <aIn1_val>6.200V</aIn1_val>
       <aIn1_min>5.000</aIn1_min>
       <aIn1_max>10.000</aIn1_max>
       <aIn1_hys>1.000</aIn1_hys>
       <aIn1_mult>40.000</aIn1_mult>
       <aIn1_offset>0.0000</aIn1_offset>
       <aIn1_alarm>0</aIn1_alarm>
    </aIn1>
    <aIn2>
       <aIn2_name>Analog Input 2</aIn2_name>
       <aIn2_val>4.955V</aIn2_val>
       <aIn2_min>5.000</aIn2_min>
       <aIn2_max>10.000</aIn2_max>
       <aIn2_hys>1.000</aIn2_hys>
       <aIn2_mult>40.000</aIn2_mult>
       <aIn2_offset>0.0000</aIn2_offset>
       <aIn2_alarm>1</aIn2_alarm>
    </aIn2>
    <dIn1>
       <dIn1_name>Digital Input 1</dIn1_name>
       <dIn1_val>OPEN</dIn1_val>
       <dIn1_alst>CLOSED</dIn1_alst>
       <dIn1_alarm>0</dIn1_alarm>
```

```
</dIn1>
<dIn2>
   <dIn2_name>Digital Input 2</dIn2_name>
   <dIn2_val>OPEN</dIn2_val>
   <dIn2_alst>CLOSED</dIn2_alst>
   <dIn2_alarm>0</dIn2_alarm>
</dIn2>
<dIn3>
   <dIn3_name>Digital Input 3</dIn3_name>
   <dIn3_val>OPEN</dIn3_val>
   <dIn3_alst>CLOSED</dIn3_alst>
   <dIn3_alarm>0</dIn3_alarm>
</dIn3>
<dIn4>
   <dIn4_name>Digital Input 4</dIn4_name>
   <dIn4_val>OPEN</dIn4_val>
   <dIn4_alst>CLOSED</dIn4_alst>
   <dIn4_alarm>0</dIn4_alarm>
</dIn4>
<Out1>
   <Out1_name>Relay 1</Out1_name>
   <Out1_val>OFF</Out1_val>
   <Out1_pw>0.1</Out1_pw>
   <Out1_contr>Web/M2M</Out1_contr>
</Out1>
   <Out2>
   <Out2_name>Relay 2</Out2_name>
   <Out2_val>ON</Out2_val>
   <Out2_pw>0.2</Out2_pw>
   <Out2_contr>Web/M2M</Out2_contr>
</Out2>
<Out3>
   <Out3_name>Relay 3</Out3_name>
   <Out3_val>ON</Out3_val>
   <Out3_pw>0.3</Out3_pw>
   <Out3_contr>Web/M2M</Out3_contr>
</Out3>
<Out4>
   <Out4_name>Relay 4</Out4_name>
   <Out4_val>ON</Out4_val>
   <Out4_pw>0.4</Out4_pw>
   <Out4_contr>Web/M2M</Out4_contr>
</Out4>
<S1>
   <S1_name>Sensor 1</S1_name>
   <S1_id>00000A97BBD6</S1_id>
   <S1T_val>23.1</S1T_val>
   <S1T_unit>°C</S1T_unit>
   <S1T_alarm>0</S1T_alarm>
   <S1T_min>-10.0</S1T_min>
   <S1T_max>40.0</S1T_max>
   <S1T_hys>1.0</S1T_hys>
   <S1H_val>---</S1H_val>
   <S1H_unit>---</S1H_unit>
   <S1H_alarm>0</S1H_alarm>
   <S1H_min>---</S1H_min>
   <S1H_max>---</S1H_max>
   <S1H_hys>---</S1H_hys>
   <S1_st>1</S1_st>
</S1>
<S2>
   <S2_name>Sensor 2 </S2_name>
   <S2_id>FFFFFFFFFFFF</S2_id>
   <S2T_val>---</S2T_val>
   <S2T_unit>---</S2T_unit>
   <S2T_alarm>0</S2T_alarm>
```

```
      <S2T_min>---</S2T_min>
      <S2T_max>---</S2T_max>
      <S2T_hys>---</S2T_hys>
      <S2H_val>---</S2H_val>
      <S2H_unit>---</S2H_unit>
      <S2H_alarm>0</S2H_alarm>
      <S2H_min>---</S2H_min>
      <S2H_max>---</S2H_max>
      <S2H_hys>---</S2H_hys>
      <S2_st>0</S2_st>
   </S2>
   <S3>
      <S3_name>Sensor 3</S3_name>
      <S3_id>FFFFFFFFFFFF</S3_id>
      <S3T_val>---</S3T_val>
      <S3T_unit>---</S3T_unit>
      <S3T_alarm>0</S3T_alarm>
      <S3T_min>---</S3T_min>
      <S3T_max>---</S3T_max>
      <S3T_hys>---</S3T_hys>
      <S3H_val>---</S3H_val>
      <S3H_unit>---</S3H_unit>
      <S3H_alarm>0</S3H_alarm>
      <S3H_min>---</S3H_min>
      <S3H_max>---</S3H_max>
      <S3H_hys>---</S3H_hys>
      <S3_st>0</S3_st>
   </S3>
   <S4>
      <S4_name>Sensor 4</S4_name>
      <S4_id>FFFFFFFFFFFF</S4_id>
      <S4T_val>---</S4T_val>
      <S4T_unit>---</S4T_unit>
      <S4T_alarm>0</S4T_alarm>
      <S4T_min>---</S4T_min>
      <S4T_max>---</S4T_max>
      <S4T_hys>---</S4T_hys>
      <S4H_val>---</S4H_val>
      <S4H_unit>---</S4H_unit>
      <S4H_alarm>0</S4H_alarm>
      <S4H_min>---</S4H_min>
      <S4H_max>---</S4H_max>
      <S4H_hys>---</S4H_hys>
      <S4_st>0</S4_st>
   </S4>
   <S5>
      <S5_name>Sensor 5</S5_name>
      <S5_id>FFFFFFFFFFFF</S5_id>
      <S5T_val>---</S5T_val>
      <S5T_unit>---</S5T_unit>
      <S5T_alarm>0</S5T_alarm>
      <S5T_min>---</S5T_min>
      <S5T_max>---</S5T_max>
      <S5T_hys>---</S5T_hys>
      <S5H_val>---</S5H_val>
      <S5H_unit>---</S5H_unit>
      <S5H_alarm>0</S5H_alarm>
      <S5H_min>---</S5H_min>
      <S5H_max>---</S5H_max>
      <S5H_hys>---</S5H_hys>
      <S5_st>0</S5_st>
   </S5>
<System>
<Time>
   <Date>06.04.2020</Date>
      <Time>10:23:23</Time>
```

```
    </Time>
    <HTTPPost>
      <PostPeriod>60</PostPeriod>
    </HTTPPost>
  </System>
</Monitor>
```

4.1.1.2. Status JSON structure

```
{
    "Monitor":{
        "Device": "IPS3000",
        "HostName": "IPS3000",
        "MAC": "54:10:EC:D7:F6:68",
        "SysName": "IPS",
        "Location": "IPS location"
    },
    "aIn1": {
        "aIn1_name": "Analog Input 1",
        "aIn1_val": "5.292V",
        "aIn1_min": "5.000",
        "aIn1_max": "10.000",
        "aIn1_hys": "1.000",
        "aIn1_mult": "40.000",
        "aIn1_offset": "0.0000",
        "aIn1_alarm": "1"
    },
    "aIn2": {
        "aIn2_name": "Analog Input 2",
        "aIn2_val": "4.155V",
        "aIn2_min": "5.000",
        "aIn2_max": "10.000",
        "aIn2_hys": "1.000",
        "aIn2_mult": "40.000",
        "aIn2_offset": "0.0000",
        "aIn2_alarm": "1"
    },
    "dIn1": {
        "dIn1_name": "Digital Input 1",
        "dIn1_val": "OPEN",
        "dIn1_alst": "CLOSED",
        "dIn1_alarm": "0"
    },
    "dIn2": {
        "dIn2_name": "Digital Input 2",
        "dIn2_val": "OPEN",
        "dIn2_alst": "CLOSED",
        "dIn2_alarm": "0"
    },
    "dIn3": {
        "dIn3_name": "Digital Input 3",
        "dIn3_val": "OPEN",
        "dIn3_alst": "CLOSED",
        "dIn3_alarm": "0"
    },
    "dIn4": {
        "dIn4_name": "Digital Input 4",
        "dIn4_val": "OPEN",
        "dIn4_alst": "CLOSED",
        "dIn4_alarm": "0"
    },
    "Out1": {
        "Out1_name": "Relay 1",
        "Out1_val": "OFF",
        "Out1_pw": "0.1",
        "Out1_contr": "Web\/M2M"
```

```
},
"Out2": {
    "Out2_name": "Relay 2",
    "Out2_val": "OFF",
    "Out2_pw": "0.2",
    "Out2_contr": "Web\/M2M"
},
"Out3": {
    "Out3_name": "Relay 3",
    "Out3_val": "OFF",
    "Out3_pw": "0.3",
    "Out3_contr": "Web\/M2M"
},
"Out4": {
    "Out4_name": "Relay 4",
    "Out4_val": "OFF",
    "Out4_pw": "0.4",
    "Out4_contr": "Web\/M2M"
},
"S1": {
    "S1_name": "Sensor 1",
    "S1_id": "00000A97BBD6",
    "S1T_val": "22.4",
    "S1T_unit": "0",
    "S1T_alarm": "0",
    "S1T_min": "-10.0",
    "S1T_max": "40.0",
    "S1T_hys": "1.0",
    "S1H_val": "---",
    "S1H_unit": "---",
    "S1H_alarm": "0",
    "S1H_min": "---",
    "S1H_max": "---",
    "S1H_hys": "---",
    "S1_st": "1"
},
"S2": {
    "S2_name": "Sensor 2 ",
    "S2_id": "FFFFFFFFFFFF",
    "S2T_val": "---",
    "S2T_unit": "0",
    "S2T_alarm": "0",
    "S2T_min": "---",
    "S2T_max": "---",
    "S2T_hys": "---",
    "S2H_val": "---",
    "S2H_unit": "---",
    "S2H_alarm": "0",
    "S2H_min": "---",
    "S2H_max": "---",
    "S2H_hys": "---",
    "S2_st": "0"
},
"S3": {
    "S3_name": "Sensor 3",
    "S3_id": "FFFFFFFFFFFF",
    "S3T_val": "---",
    "S3T_unit": "0",
    "S3T_alarm": "0",
    "S3T_min": "---",
    "S3T_max": "---",
    "S3T_hys": "---",
    "S3H_val": "---",
    "S3H_unit": "---",
    "S3H_alarm": "0",
    "S3H_min": "---",
```

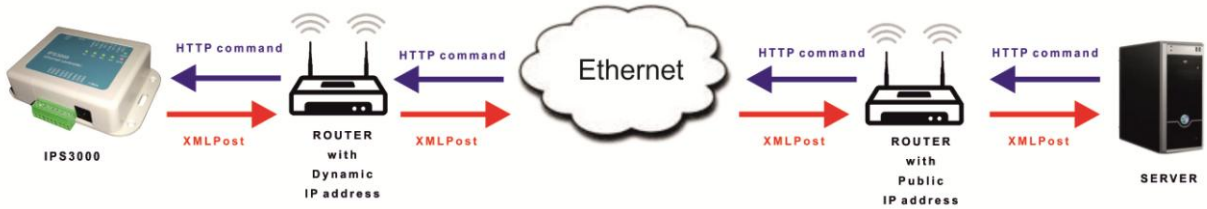
```
            "S3H_max": "---",
            "S3H_hys": "---",
            "S3_st": "0"
        },
        "S4": {
            "S4_name": "Sensor 4",
            "S4_id": "FFFFFFFFFFFF",
            "S4T_val": "---",
            "S4T_unit": "0",
            "S4T_alarm": "0",
            "S4T_min": "---",
            "S4T_max": "---",
            "S4T_hys": "---",
            "S4H_val": "---",
            "S4H_unit": "---",
            "S4H_alarm": "0",
            "S4H_min": "---",
            "S4H_max": "---",
            "S4H_hys": "---",
            "S4_st": "0"
        },
        "S5": {
            "S5_name": "Sensor 5",
            "S5_id": "FFFFFFFFFFFF",
            "S5T_val": "---",
            "S5T_unit": "0",
            "S5T_alarm": "0",
            "S5T_min": "---",
            "S5T_max": "---",
            "S5T_hys": "---",
            "S5H_val": "---",
            "S5H_unit": "---",
            "S5H_alarm": "0",
            "S5H_min": "---",
            "S5H_max": "---",
            "S5H_hys": "---",
            "S5_st": "0"
        },
        "System": {
            "Time": {
                "Date": "06.04.2020",
                "Time": "10:49:39"
            },
            "HTTPPost": {
                "PostPeriod": "60"
            }
        }
    }
```


4.1.2. HTTP Post

HTTP Post functionality is very useful when the controller without public IP address must to send data to server with public IP address.

HTTP Post can be send periodically. As response the server can send HTTP Get with Command.

For more info please contact us.



4.2. SNMP

SNMP offers the capability to poll network device and monitor data. SNMP is also capable changing the configurations on the host, allowing the remote management of the network device.

The protocol uses a community string for authentication from the SNMP client to the SNMP agent on the managed device.

SNMP parameters for IPS3000 can be controlled with any SNMP compatible program. To obtain a valid OID number in table bellow it is necessary to replace the “x” symbol with ”1.3.6.1.4.1.17095”.

product

OID	Name	Access	Description	Syntax
X.3.1.1.0	name	read-only	Device name	String
X.3.1.2.0	version	read-only	Firmware version	String
X.3.1.3.0	dateTime	read-only	Current date and time	String

setup -> Network

OID	Name	Access	Description	Syntax
X.3.2.1.1.0	deviceID	read-only	Device ID (default MAC address)	String
X.3.2.1.2.0	hostName	read-only	Hostname	String
X.3.2.1.3.0	deviceIP	read-write	Device IP address	IpAddress

setup -> analogSetup -> analog1setup

OID	Name	Access	Description	Syntax
X.3.2.2.1.1.0	analogIn1Name	read-write	Analog input 1 Name	DisplayString
X.3.2.2.1.2.0	analogIn1Upper	read-write	Analog input 1 upper range in integer (decimal x 1000) format.	Integer32
X.3.2.2.1.3.0	analogIn1Lower	read-write	Analog input 1 low range in integer (decimal x 1000) format.	Integer32
X.3.2.2.1.4.0	analogIn1Hyst	read-write	Analog input 1 hysteresis	Integer32

setup -> analogSetup -> analog2setup

OID	Name	Access	Description	Syntax
X.3.2.2.2.1.0	analogIn2Name	read-write	Analog input 2 name	DisplayString
X.3.2.2.2.2.0	analogIn2Upper	read-write	Analog input 2 upper range in integer (decimal x 1000) format.	Integer32
X.3.2.2.2.3.0	analogIn2Lower	read-write	Analog input 2 low range in integer (decimal x 1000) format.	Integer32
X.2.2.3.2.4.0	analogIn2Hyst	read-write	Analog input 2 hysteresis	Integer32

setup -> digitalSetup

OID	Name	Access	Description	Syntax
X.3.2.3.1.0	digitalIn1description	read-write	Digital Input 1 description	DisplayString
X.3.2.3.2.0	digitalIn2description	read-write	Digital Input 2 description	DisplayString
X.3.2.3.3.0	digitalIn3description	read-write	Digital Input 3 description	DisplayString
X.3.2.3.4.0	digitalIn4description	read-write	Digital Input 4 description	DisplayString

setup -> sensorsSetup -> sensor1setup

OID	Name	Access	Description	Syntax
X.2.2.4.1.1.0	sens1Name	read-write	Sensor 1 Name	String

setup -> sensorsSetup -> sensor1Setup -> sensor11setup

OID	Name	Access	Description	Syntax
X.3.2.4.1.2.1.0	s11Upper	read-write	S11 upper range x1000 in Integer format	Integer32
X.3.2.4.1.2.2.0	s11Lower	read-write	S11 lower range x1000 in Integer format	Integer32
X.3.2.4.1.2.3.0	s11Hyst	read-write	S11 hysteresis value x1000 in Integer format	Integer32

setup -> sensorsSetup -> sensor1Setup -> sensor12setup

OID	Name	Access	Description	Syntax
X.3.2.4.1.3.1.0	s12Upper	read-write	S12 upper range x1000 in Integer format	Integer32
X.3.2.4.1.3.2.0	s12Lower	read-write	S12 lower range x1000 in Integer format	Integer32
X.3.2.4.1.3.3.0	s12Hyst	read-write	S12 hysteresis value x1000 in Integer format	Integer32

setup -> sensorsSetup -> sensor2setup

OID	Name	Access	Description	Syntax
X.3.2.4.2.1.0	sens2Name	read-write	Sensor 2 Name	String

setup -> sensorsSetup -> sensor2Setup -> sensor21setup

OID	Name	Access	Description	Syntax
X.3.2.4.2.2.1.0	s21Upper	read-write	S21 upper range x1000 in Integer format	Integer32
X.3.2.4.2.2.2.0	s21Lower	read-write	S21 lower range x1000 in Integer format	Integer32
X.3.2.4.2.2.3.0	s21Hyst	read-write	S21 hysteresis value x1000 in Integer format	Integer32

setup -> sensorsSetup -> sensor2Setup -> sensor22setup

OID	Name	Access	Description	Syntax
X.3.2.4.2.3.1.0	s22Upper	read-write	S22 upper range x1000 in Integer format	Integer32
X.3.2.4.2.3.2.0	s22Lower	read-write	S22 lower range x1000 in Integer format	Integer32
X.3.2.4.2.3.3.0	s22Hyst	read-write	S22 hysteresis value x1000 in Integer format	Integer32

setup -> sensorsSetup -> sensor3setup

OID	Name	Access	Description	Syntax
X.3.2.4.3.1.0	sens3Name	read-write	Sensor 3 Name	String

setup -> sensorsSetup -> sensor3Setup -> sensor31setup

OID	Name	Access	Description	Syntax
X.3.2.4.3.2.1.0	s31Upper	read-write	S31 upper range x1000 in Integer format	Integer32
X.3.2.4.2.2.2.0	s31Lower	read-write	S31 lower range x1000 in Integer format	Integer32
X.3.2.4.2.2.3.0	s31Hyst	read-write	S31 hysteresis value x1000 in Integer format	Integer32

setup -> sensorsSetup -> sensor3Setup -> sensor32setup

OID	Name	Access	Description	Syntax
X.3.2.4.3.3.1.0	s32Upper	read-write	S32 upper range x1000 in Integer format	Integer32
X.3.2.4.3.3.2.0	s32Lower	read-write	S32 lower range x1000 in Integer format	Integer32
X.3.2.4.3.3.3.0	s32Hyst	read-write	S32 hysteresis value x1000 in Integer format	Integer32

setup -> sensorsSetup -> sensor4Setup

OID	Name	Access	Description	Syntax
X.3.2.4.4.1.0	sens4Name	read-write	Sensor 4 Name	String

setup -> sensorsSetup -> sensor4Setup -> sensor41setup

OID	Name	Access	Description	Syntax
X.3.2.4.4.2.1.0	s41Upper	read-write	S41 upper range x1000 in Integer format	Integer32
X.3.2.4.4.2.2.0	s41Lower	read-write	S41 lower range x1000 in Integer format	Integer32
X.3.2.4.4.2.3.0	s41Hyst	read-write	S41 hysteresis value x1000 in Integer format	Integer32

setup -> sensorsSetup -> sensor4Setup -> sensor42setup

OID	Name	Access	Description	Syntax
X.3.2.4.4.3.1.0	s42Upper	read-write	S42 upper range x1000 in Integer format	Integer32
X.3.2.4.4.3.2.0	s42Lower	read-write	S42 lower range x1000 in Integer format	Integer32
X.3.2.4.4.3.3.0	s42Hyst	read-write	S42 hysteresis value x1000 in Integer format	Integer32

setup -> sensorsSetup -> sensor5setup

OID	Name	Access	Description	Syntax
X.3.2.4.5.1.0	sens5Name	read-write	Sensor 5 Name	String

setup -> sensorsSetup -> sensor5Setup -> sensor51setup

OID	Name	Access	Description	Syntax
X.3.2.4.5.2.1.0	s51Upper	read-write	S51 upper range x1000 in Integer format	Integer32
X.3.2.4.5.2.2.0	s51Lower	read-write	S51 lower range x1000 in Integer format	Integer32
X.3.2.4.5.2.3.0	s51Hyst	read-write	S51 hysteresis value x1000 in Integer format	Integer32

setup -> sensorsSetup -> sensor5Setup -> sensor52setup

OID	Name	Access	Description	Syntax
X.3.2.4.5.3.1.0	s52Upper	read-write	S52 upper range x1000 in Integer format	Integer32
X.3.2.4.5.3.2.0	s52Lower	read-write	S52 lower range x1000 in Integer format	Integer32
X.3.2.4.5.3.3.0	s52Hyst	read-write	S52 hysteresis value x1000 in Integer format	Integer32

setup -> outsSetup -> out1setup

OID	Name	Access	Description	Syntax
X.3.2.5.1.1.0	out1Name	read-write	Name of relay 1 output	String
X.3.2.5.1.2.0	out1pulse	read-write	Out1 Pulse x100ms	Integer32
X.3.2.5.1.3.0	out1Control	read-write	out1Control	Integer { webm2m(0), analogIn1(1), analogIn2(2),digitalIn1 (3), digitalIn2(4), digitalIn3(5), digitalIn4(6), S11(7), S12(8), S21(9), S22(10), S31(11),S32(12),S41(13),S42 (14),S51(15),S52(16),anyAlar m (17), schedule1(18), schedule2(19), schedule3(20), schedule4(21)}}

setup -> outsSetup -> out2setup

OID	Name	Access	Description	Syntax
X.3.2.5.2.1.0	out2Name	read-write	Name of relay 2 output	String
X.3.2.5.2.2.0	out2pulse	read-write	Out2 Pulse x100ms	Integer32
X.3.2.5.2.3.0	out2Control	read-write	out2Control	Integer { webm2m(0), analogIn1(1), analogIn2(2),digitalIn1 (3), digitalIn2(4), digitalIn3(5), digitalIn4(6), S11(7), S12(8), S21(9), S22(10), S31(11),S32(12),S41(13),S42 (14),S51(15),S52(16),anyAlar m (17), schedule1(18), schedule2(19), schedule3(20), schedule4(21)}}

setup -> outsSetup -> out3setup

OID	Name	Access	Description	Syntax
X.3.2.5.3.1.0	out3Name	read-write	Name of relay 3 output	String
X.3.2.5.3.2.0	out3pulse	read-write	Out3 Pulse x100ms	Integer32
X.3.2.5.3.3.0	out3Control	read-write	out2Control	Integer { webm2m(0), analogIn1(1), analogIn2(2),digitalIn1 (3),

				digitalIn2(4), digitalIn3(5), digitalIn4(6), S11(7), S12(8), S21(9), S22(10), S31(11),S32(12),S41(13),S42(14),S51(15),S52(16),anyAlarm (17), schedule1(18), schedule2(19), schedule3(20), schedule4(21)}}

setup -> outsSetup -> out4setup

OID	Name	Access	Description	Syntax
X.3.2.5.4.1.0	out4Name	read-write	Name of relay 4 output	String
X.3.2.5.4.2.0	out4pulse	read-write	out4 Pulse x100ms	Integer32
X.3.2.5.4.3.0	out4Control	read-write	out4Control	Integer { webm2m(0), analogIn1(1), analogIn2(2),digitalIn1 (3), digitalIn2(4), digitalIn3(5), digitalIn4(6), S11(7), S12(8), S21(9), S22(10), S31(11),S32(12),S41(13),S42(14),S51(15),S52(16),anyAlarm (17), schedule1(18), schedule2(19), schedule3(20), schedule4(21)}}

control -> analog

OID	Name	Access	Description	Syntax
X.3.3.1.1.0	analog1Int	read only	Analog1 in integer (decimal x 1000) format	Integer32
X.3.3.1.2.0	analog2Int	read only	Analog2 in integer (decimal x 1000) format	Integer32

control -> digital

OID	Name	Access	Description	Syntax
X.3.3.2.1.0	digitalIn1State	read only	Digital1 Input State	Integer
X.3.3.2.2.0	digitalIn2State	read only	Digital2 Input State	Integer
X.3.3.2.3.0	digitalIn3State	read only	Digital3 Input State	Integer
X.3.3.2.4.0	digitalIn4State	read only	Digital4 Input State	Integer

control -> sensors -> sensor1

OID	Name	Access	Description	Syntax
X.3.3.3.1.1.0	s11Int	read only	Sensor1 Temperature in integer (decimal x 1000) format	Integer32
X.3.3.3.1.2.0	s12Int	read only	Sensor1 Humidity in integer (decimal x 1000) format	Integer32
X.3.3.3.1.3.0	s1ID	read only	S1 ID value	OCTET STRING

control -> sensors -> sensor2

OID	Name	Access	Description	Syntax
X.3.3.3.2.1.0	s21Int	read only	Sensor2 Temperature in integer (decimal x 1000) format	Integer32
X.3.3.3.2.2.0	s22Int	read only	Sensor2 Humidity in integer (decimal x 1000) format	Integer32
X.3.3.3.2.3.0	s2ID	read only	S2 ID value	OCTET STRING

control -> sensors -> sensor3

OID	Name	Access	Description	Syntax
X.3.3.3.3.1.0	s31Int	read only	Sensor3 Temperature in integer (decimal x 1000) format	Integer32
X.3.3.3.3.2.0	s32Int	read only	Sensor3 Humidity in integer (decimal x 1000) format	Integer32
X.3.3.3.3.3.0	s3ID	read only	S3 ID value	OCTET STRING

control -> sensors -> sensor4

OID	Name	Access	Description	Syntax
X.3.3.3.4.1.0	s41Int	read only	Sensor4 Temperature in integer (decimal x 1000) format	Integer32
X.3.3.3.4.2.0	s42Int	read only	Sensor4 Humidity in integer (decimal x 1000) format	Integer32
X.3.3.3.4.3.0	s4ID	read only	S4 ID value	OCTET STRING

control -> sensors -> sensor5

OID	Name	Access	Description	Syntax
X.3.3.3.5.1.0	s51Int	read only	Sensor5 Temperature in integer (decimal x 1000) format	Integer32
X.3.3.3.5.2.0	s52Int	read only	Sensor5 Humidity in integer (decimal x 1000) format	Integer32
X.3.3.3.5.3.0	s5ID	read only	S5 ID value	OCTET STRING

control -> outs -> out1

OID	Name	Access	Description	Syntax
X.3.3.4.1.1.0	out1State	read only	Out1 State	Integer
X.3.3.4.2.1.0	out1Pulse	read only	Out1 Pulse	Integer

control -> outs -> out2

OID	Name	Access	Description	Syntax
X.3.3.4.2.1.0	sut2State	read only	Out2 State	Integer
X.3.3.4.2.2.0	out2Pulse	read only	Out2 Pulse	Integer

control

OID	Name	Access	Description	Syntax
X.3.3.5.0	configSaved	read-write	Configuration save status YES/NO	Integer
X.3.3.6.0	restartDevice	read-write	Restart Device	Integer
X.3.3.7.0	tempUnit	read-write	Unit of the all temperature values	Integer

Section 5: Factory default settings

5.1. Default network parameters configuration:

IP address: 192.168.2.2

Network mask: 255.255.255.0

Default gateway: 192.168.2.1

IP configuration via DHCP: Disabled

User name: admin

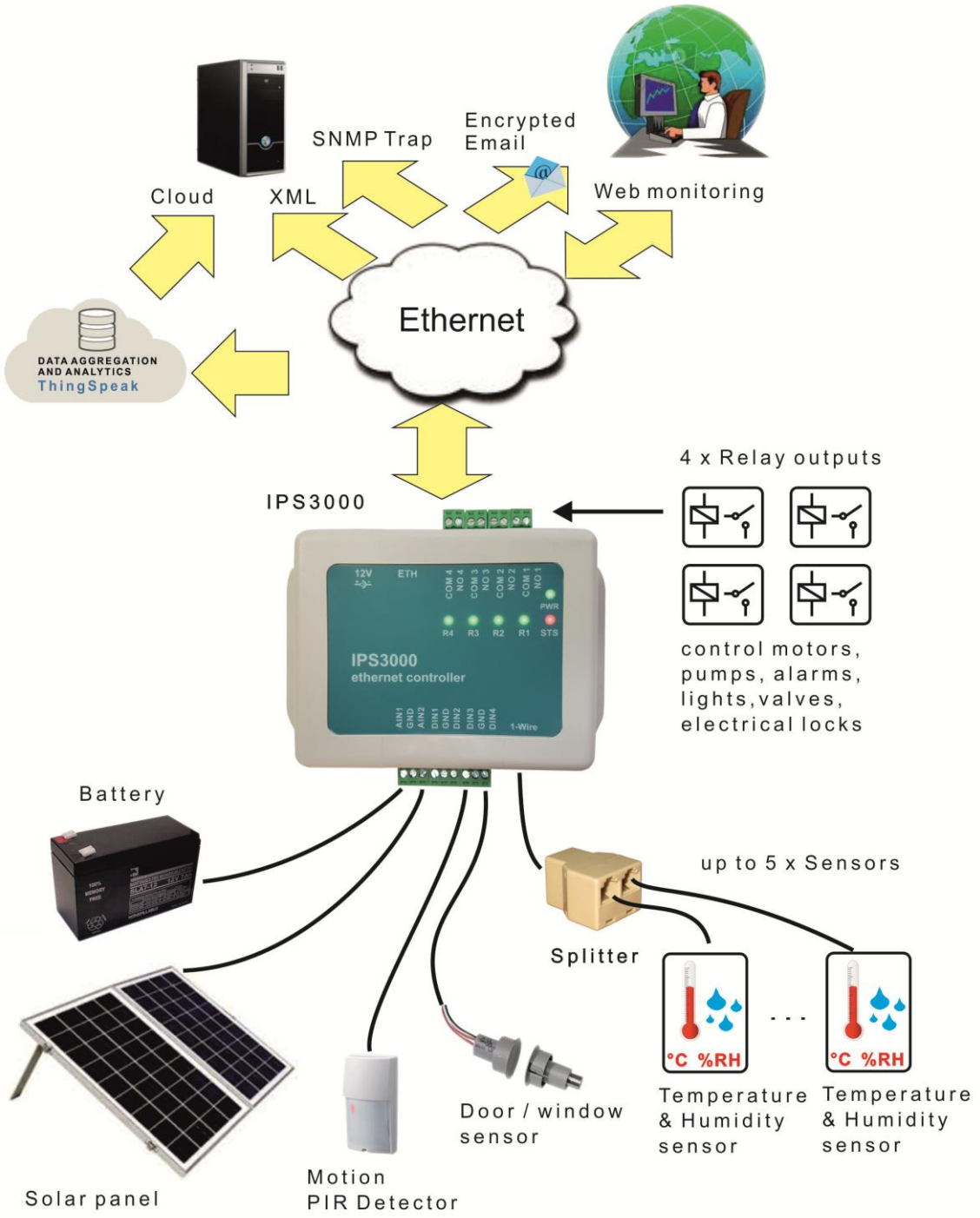
Password: admin

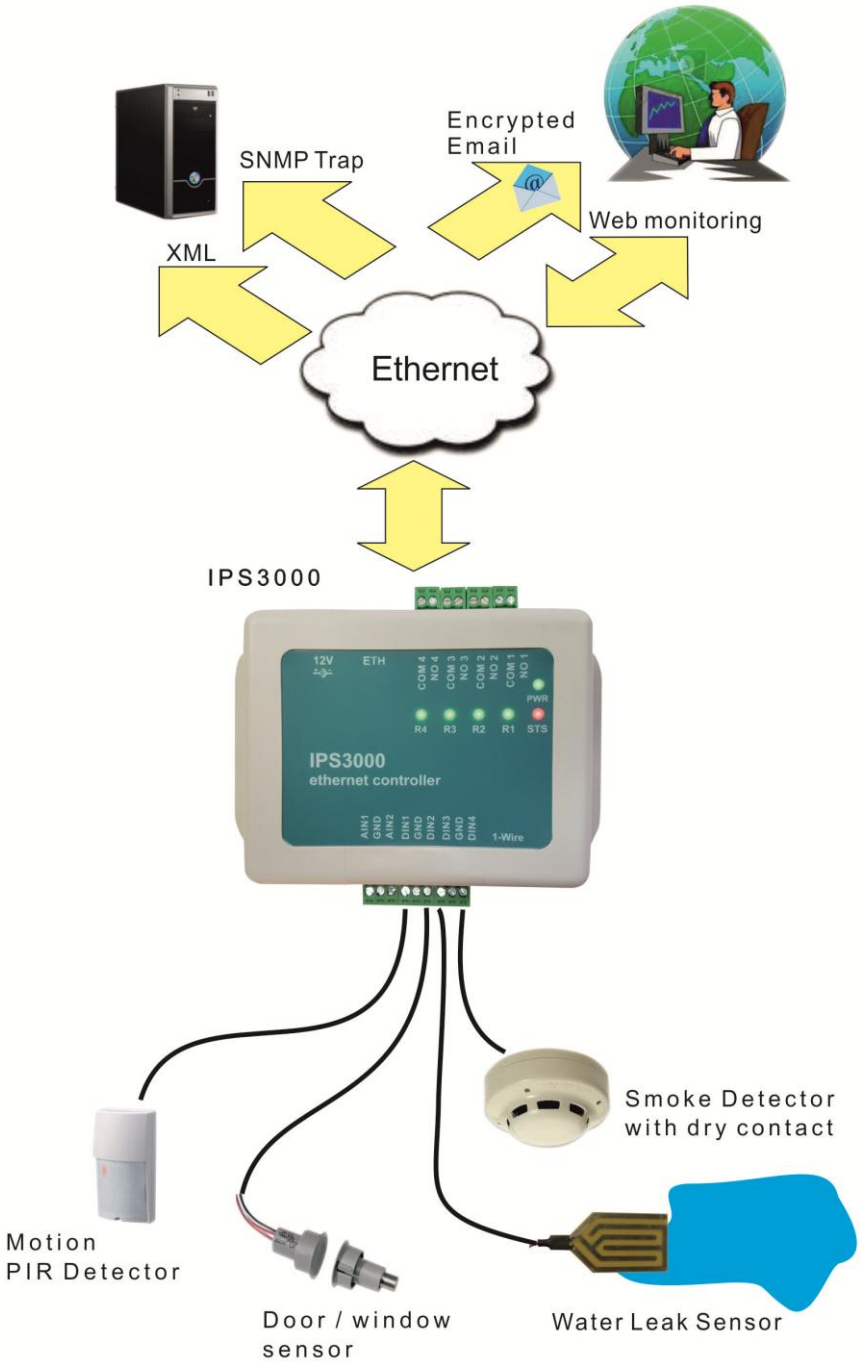
5.2. Restoring defaults factory configuration

- Press and hold the Reset button.
- Connect power supply and hold the button pressed.
- Green Led will start flashing. Release the button after Green Led stops flashing and starts to glow.

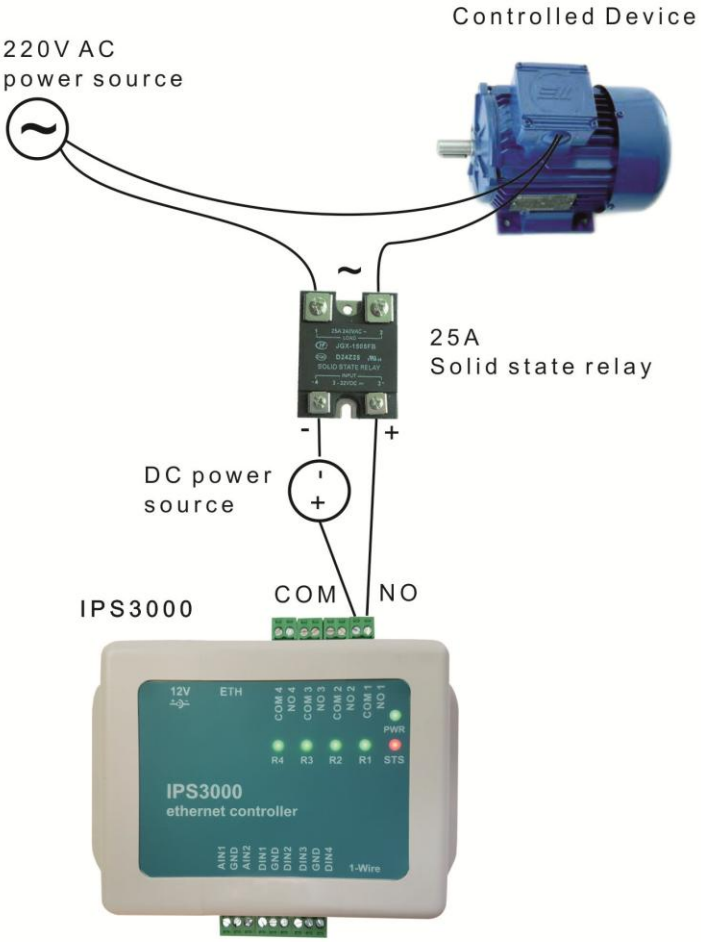
Section 6: Application Examples

6.1. Remote monitoring application

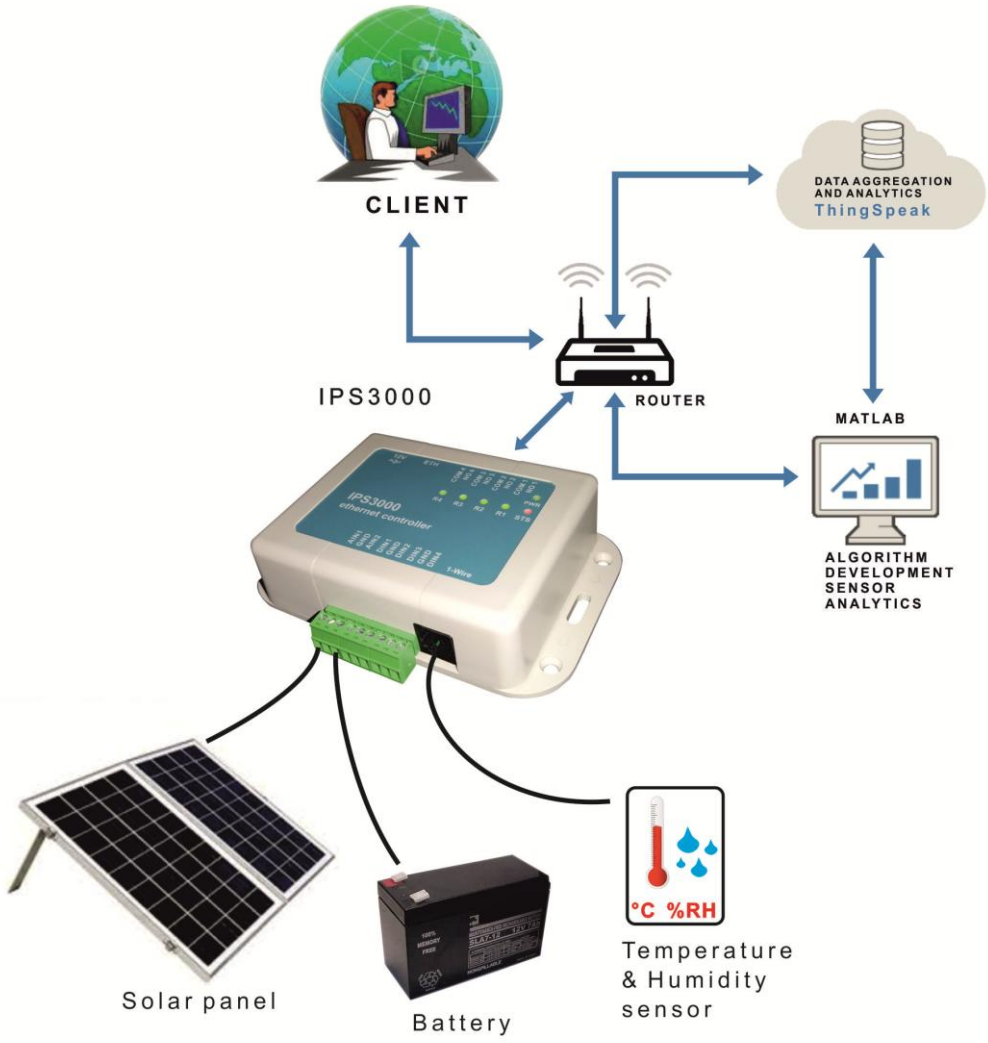



Encrypted
Email
SNMP Trap
Web monitoring
XML
Ethernet
IPS3000
12V
ETH
COM 4
NO 4
COM 3
NO 3
COM 2
NO 2
COM 1
NO 1
PWR
R4
R3
R2
R1
STS
IPS3000
ethernet controller
AIN1
GND
AIN2
DIN1
GND
DIN2
DIN3
GND
DIN4
1-Wire
Smoke Detector
with dry contact
Motion
PIR Detector
Door / window
sensor
Water Leak Sensor

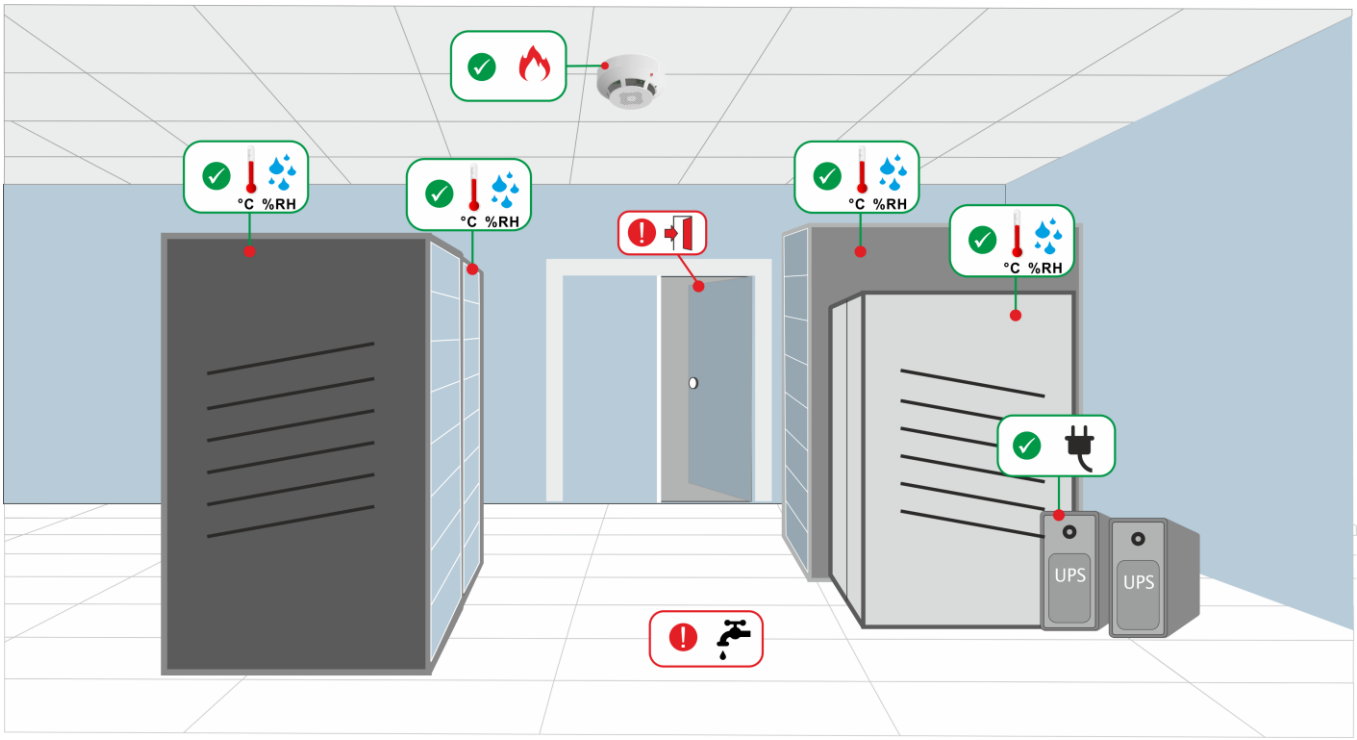
6.2. Control of 220V AC / 20 A motor using external relay



6.3. ThingSpeak cloud application



6.4. Monitoring data centers and server rooms



Section 7: Mechanical dimensions

Bottom view