control solutions
TERACOM

TCW220 Ethernet data logger

Revision 4.22 / October 2024

USER MANUAL

For pricing or any further information, please contact Omni Instruments Ltd.

Measurement and data acquisition solutions

Contact Details:
Tel: +44 1382 443000
Email: info@omni.uk.com

Website: www.omniinstruments.co.uk

Mailing Address: Unit 1, 14 Nobel Road,
Wester Gourdie Industrial Estate,
Dundee, DD2 4UH.

Ethernet data logger TCW220

1. Introduction

The TCW220 is a versatile, stand-alone Ethernet data logger designed for data acquisition applications. It features two 10-bit resolution analog inputs and two digital "dry contact" inputs. Additionally, it supports up to 8 Teracom 1-Wire sensors through its 1-Wire interface, enabling measurement of parameters such as temperature, humidity, CO2 levels, current, 4/20mA signals, and galvanically isolated analog voltages.

The TCW220 includes two relays with both normally open and normally closed contacts. These relays can be activated either remotely (via web interface, SNMP, HTTP API, Modbus TCP/IP, etc.) or locally based on the status of a monitored parameter (1-Wire sensor, analog voltage, or dry contact input).

All monitored data can be logged into the device’s internal flash memory, either at pre-set time intervals or when alarm conditions are triggered. Logged data can be periodically uploaded to a designated server via HTTP/HTTPS POST, and the stored information can be viewed in graphical form across four different graph pages.

The TCW220 can also periodically upload an XML or JSON file containing all monitored parameters to a server via HTTP/HTTPS POST.

For each monitored parameter, the device can send alarm notifications via email or SNMP traps to up to 5 recipients. Alarm alerts can also be transmitted using HTTP/HTTPS POST, along with the XML/JSON status file.

2. Features

- 10/100 Mbps Ethernet connectivity with Auto-MDIX support
- Ethernet data logging with storage for up to 70,000 records
- Graphical display of logged data
- Password-protected web-based configuration and control
- 2 digital "dry contact" inputs
- 2 analog inputs (0 to 10VDC range) with 10-bit resolution
- Configurable multiplier, offset, and unit of measurement for analog inputs
- 2 relays with normally open (NO) and normally closed (NC) contacts
- 1-Wire interface supporting up to 8 Teracom sensors
- SNMP v2 support
- SNMP traps for alarm alerts, sent to up to 5 recipients
- Modbus TCP/IP support
- Support for TLS 1.0, 1.1, and 1.2
- SMTP with TLS security
- Periodic HTTP/HTTPS POST of XML/JSON status files for client-server systems
- Email alerts to up to 5 recipients for alarms
- HTTP API command support
- Dynamic DNS support with DynDNS, No-IP, and DNS-O-Matic
- NTP (Network Time Protocol) support
- Configurable ports for HTTP, SNMP, and Modbus TCP/IP
- Wide power supply voltage range
- Remote firmware update capability

3. Applications

The TCW220 is well-suited for a wide range of applications, including industrial automation, data acquisition, environmental monitoring, and local control of both electrical and non-electrical parameters. It is also an excellent choice for building automation systems.

The device can operate as a standalone unit controlled via a web browser, or it can be integrated into small or large industrial control systems, such as SCADA (Supervisory Control and Data Acquisition systems).

Some example applications include:

- Stand-alone or system-integrated Ethernet data logger
- Industrial process automation
- Industrial environmental monitoring
- Environmental monitoring and control in mushroom farms and wineries
- Integration into small to large SCADA systems
- Ethernet data logging for storage facilities.

4. Specifications

- Physical characteristics
  Dimensions: 115 x 90 x 40 mm
  Weight: 170 g

- Environmental limits
  Operating temperature range: -20 to 55°C
  Storage temperature range: -25 to 60°C
  Operating relative humidity range: 5 to 85% (non-condensing)

- Warranty
  Warranty period: 3 years

- Power supply
  Operating voltage range (including -15/+20% according to IEC 62368-1): 10 to 28 VDC
  Current consumption: 240 mA @ 12 VDC (with all relays ON)

- Digital inputs
  Isolation: Non isolated
  Mode: Dry contact
  Maximum input voltage: +5.5VDC
  Sampling rate: 10mS
  Digital filtering time interval: 30mS

- Analog inputs
  Isolation: Non isolated
  Type: Single ended
  Resolution: 10 bits
  Mode: Voltage
  Input Range: 0 to 10 VDC
  Accuracy: ±1%
  Sampling Rate: 500mS per channel (averaged value of 500 samples)
  Input Impedance: 150 kilo-ohms (min.)

- Relay outputs
  Type: Form C (N.O. and N.C. contacts)
  Contact current rating: 3 A @ 24 VDC/30 VAC (resistive load)
  Initial insulation resistance: 100 mega-ohms (min.) @ 500 VDC
  Mechanical endurance: 10 000 000 operations
  Electrical endurance: 100 000 operations @ 3 A resistive load
  Contact resistance: 50 milli-ohms max. (initial value)
  Minimum pulse output: 1 Hz at rated load

  ⚠ **Caution:** The device does not contain any internal overcurrent protection facilities on the relay's contact lines. External fuses or short circuit current limiting circuit breakers, rated to 3 Amps, are to be used for overcurrent protection of the connecting lines.

- 1-Wire interface
  Output voltage (+VW): 5.0 ± 0.3 VDC
  Maximum output current (+VW): 0.2 A

- Internal FLASH memory
  Endurance: 100 000 cycles (Every settings change is a memory cycle.)

- Lithium battery
  Type: CR1220

  ⚠ **Caution:** Replacing the battery with an incorrect type may result in an explosion.

5. LED indicators

The TCW220 controller includes the following LED indicators to display its status:

- **Relay1-Relay2** (green) – Illuminates when the corresponding relay is activated (NO contact closed, NC contact open);
- **PWR** (red) – Steady ON in normal operation; blinks in sync with the STS LED in the event of a hardware error;
- **STS** (yellow) – Flashes to indicate that the controller's main program is running;
- **NET** (orange) – Indicates network status; remains ON when a network link is established, and blinks to show network activity.

6. Installation and setup

Qualified personnel must install the device. It shouldn't be installed outside directly.

The installation process involves mounting the device, connecting it to an IP network, attaching inputs and outputs, supplying power, and configuring it through a web browser.

6.1. Mounting

TCW220 must be installed in a clean, dry, and non-flammable location. Ventilation is recommended for high ambient temperature environments.

To mount the device, use two plastic dowels (e.g. Würth GmbH 0912 802 002) and two dowel screws (e.g. Würth GmbH 0157 06 70) to secure it to a wall. Refer to fig.1 in Appendix A for mechanical details.

Leave 50 mm of space on all sides for ventilation and electrical isolation. Refer to fig.2 in Appendix A.

The device can also be attached to a standard DIN rail (35mm by 7.55mm) by hooking the back of the enclosure onto the rail and snapping the bottom into place.

6.2. Connection

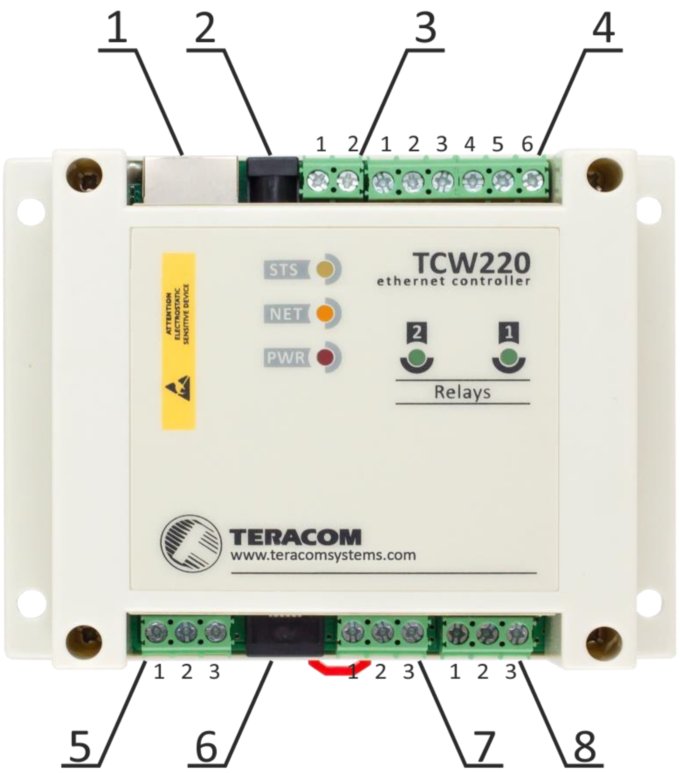
Warning! Power off before wiring.

Follow these steps for correct wiring:

- Turn off power;
- Connect wires to terminals;
- Turn on the power.

Ensure that wires are securely attached to terminals and tightened. Improper wiring or configuration can cause permanent damage to TCW220 or connected equipment.

.



Connector 1	Ethernet - RJ45
Connector 2	Power - 2.1x5.5mm connector, central positive
Connector 3	Pin1 – Power positive
	Pin2 – Power negative
Connector 4	Pin1 – NC Relay2
	Pin2 – COM Relay2
	Pin3 – NO Relay2
	Pin4 – NC Relay1
	Pin5 – COM Relay1
	Pin6 – NO Relay1
Connector 5	Pin1 – 1-Wire GND
	Pin2 – 1-Wire Data
	Pin3 – 1-Wire +VDD
Connector 6	Pin1 – GND (most left)
	Pin2 – GND
	Pin3 – 1-Wire Data
	Pin4 – 1-Wire GND
	Pin5 – 1-Wire +VDD
	Pin6 – 1-Wire +VDD (most right)
Connector 7	Pin1 – Digital In 1
	Pin2 – GND
	Pin3 – Digital In 2
Connector 8	Pin1– Analog In 1
	Pin2 – GND
	Pin3 – Analog In 2

6.2.1. Power supply

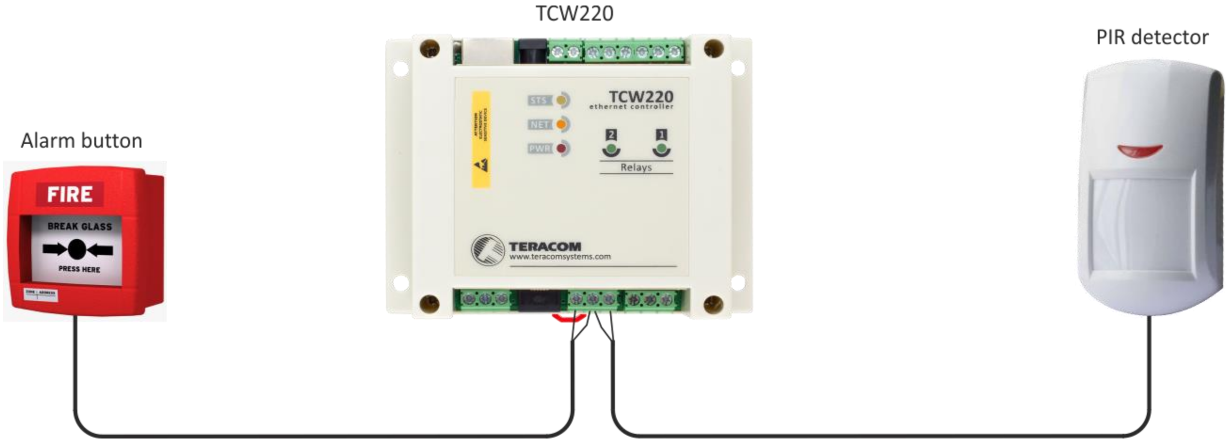
TCW220 must be powered by the adapter SYS1308(N)-2412-W2E or equivalent, suitable for overvoltage category II and certified for safety compliance. The power supply device should be able to withstand short circuits and secondary circuit overloads. Ensure the equipment is easily accessible for disconnecting from the power supply during use.

6.2.2. Digital inputs

Note that all inputs are not isolated from the power supply.

These inputs are designed for monitoring devices with "dry contact" outputs, such as door contact switches, push buttons, and PIR detectors.

The diagram below demonstrates how to connect a dry contact switch to the TCW220 input. One side of the contact is connected to the "Digital In" terminal, while the other side is connected to the "GND" terminal.



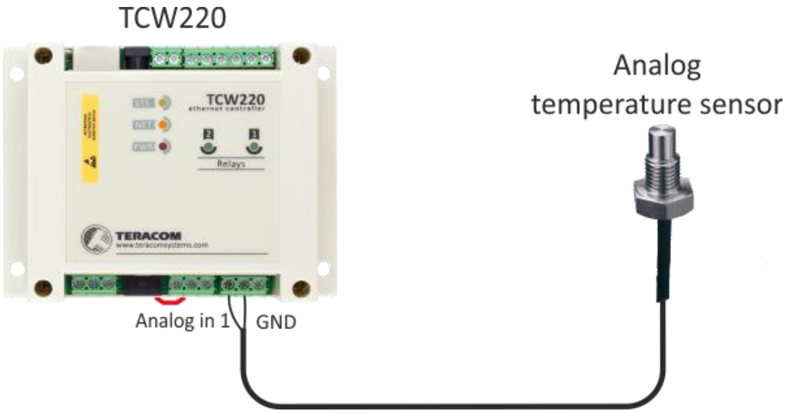
The maximum cable length should not exceed 30 meters.

6.2.3. Analog inputs

Note that all inputs are not isolated from the power supply.

The analog inputs support a standard industrial range of 0 to 10VDC and can be directly connected to analog sensors with outputs of either 0 to 5VDC or 0 to 10VDC. These inputs are factory-calibrated to ensure better accuracy. The device offers built-in functions for each analog input, such as "Multiplier", "Offset", and "Dimension" allowing it to be used with a wide variety of analog sensors. This functionality ensures that the directly measured parameters are displayed accurately. Additionally, the TCW220 can monitor voltages higher than 10VDC by using external resistive dividers.

The diagram below demonstrates how to connect a high-temperature sensor to the analog input of the TCW220. The sensor's output is connected to the “Analog In” terminal, and the shield is connected to the “GND” terminal.



The maximum cable length should not exceed 30 meters.

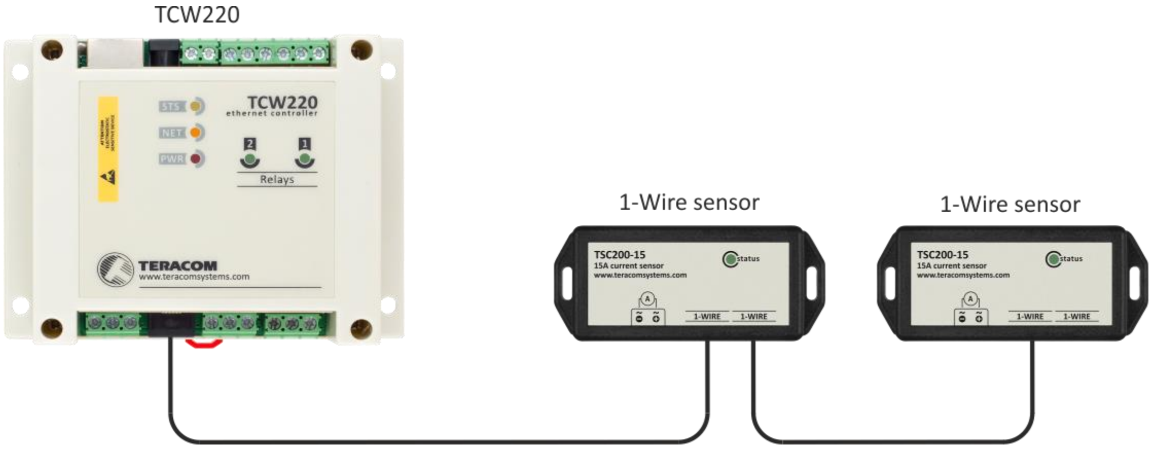
6.2.4. 1-Wire interface

The TCW220 supports the connection of up to eight 1-Wire sensors. Compatible sensors include those for temperature, temperature/humidity, CO2, DC current, AC current, 4/20mA signals, galvanically isolated analog voltage, barometric pressure, and more. Connected sensors are automatically detected, and the appropriate measurement units are assigned.

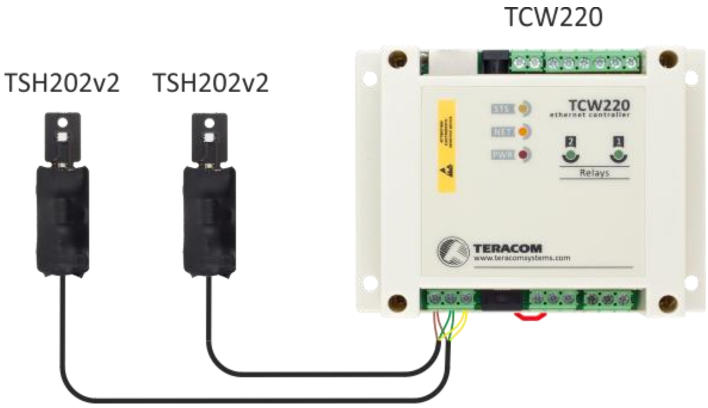
1-Wire, a registered trademark of Analog devices Inc., is designed for connecting multiple sensors over short wiring distances. It is not suitable for long distances or environments with significant electromagnetic interference (EMC). For best practices, refer to Guidelines for Reliable Long-Line 1-Wire Networks.

The sensors typically have three wires: positive voltage (+VDD), ground (GND), and bidirectional data (Data). The specific wire colors for each sensor are provided in the sensor's user manual.

For reliable operation with multiple sensors, it is strongly recommended to use a "daisy-chained" (linear) topology:



A "star" topology should only be used as a last resort, supporting a maximum of 4 sensors and a total cable length of up to 10 meters:



Connections to the 1-Wire interface can be made either through a screw terminal connector or a standard RJ-11 connector.

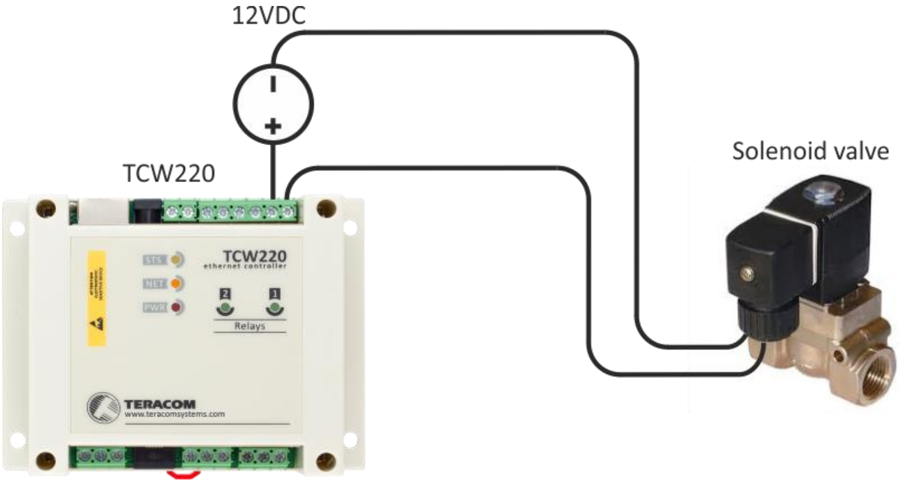
Several factors influence the maximum allowable length of the wires, including the type of cable used, the number of sensors, ambient electromagnetic noise, and the network topology of the sensor connections.

It is strongly recommended to use only UTP (Unshielded Twisted Pair) or FTP (Foiled Twisted Pair) cables, with a total cable length limited to 30 meters. While longer distances may still function, error-free operation cannot be guaranteed beyond the recommended length.

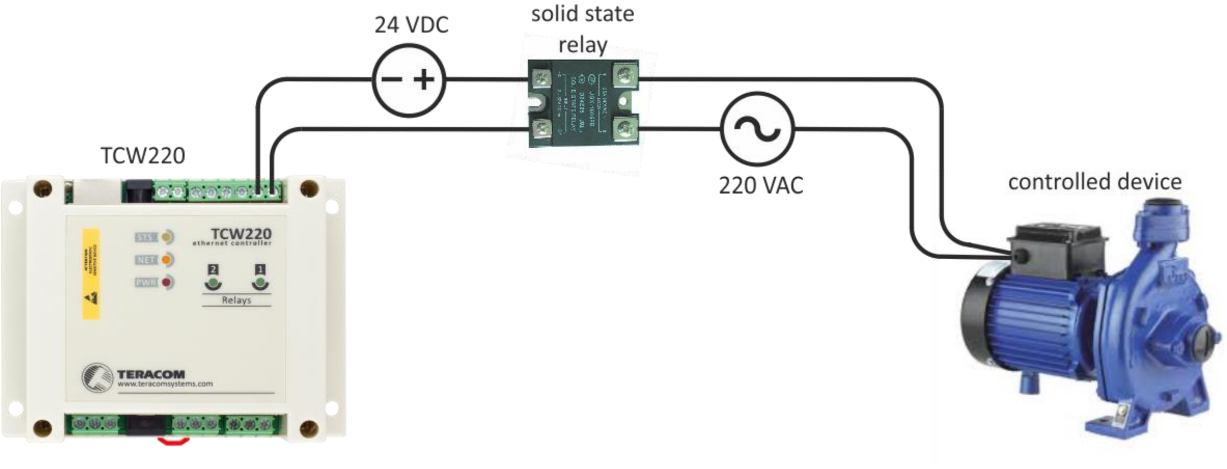
For optimal performance, Teracom guarantees proper operation only when using Teracom 1-Wire sensors.

6.2.5. Relays

The relay contacts are directly connected to the terminal connectors. Each relay provides normally open (NO), normally closed (NC), and common (COM) contacts for easy integration.



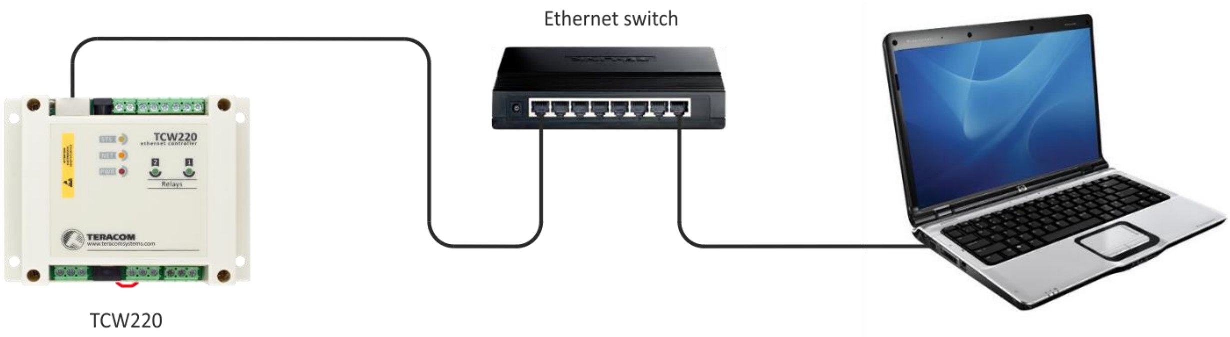
For loads that exceed the specified current or voltage limits, an external relay should be used.



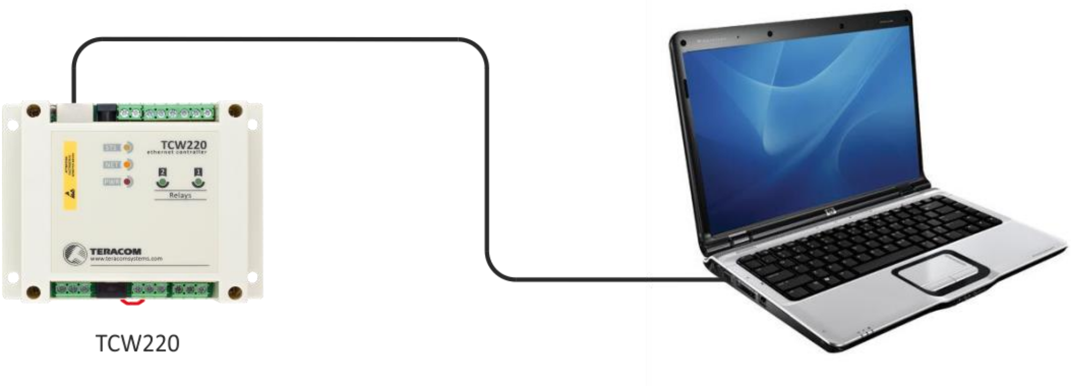
When switching inductive loads, such as motors, transformers, or other relays, mechanical relays are subject to contact arcing each time the contacts open. This can lead to wear and reduced relay lifespan. To mitigate this, it is recommended to use relay contact protection devices when switching inductive loads.

6.2.6. Network connection

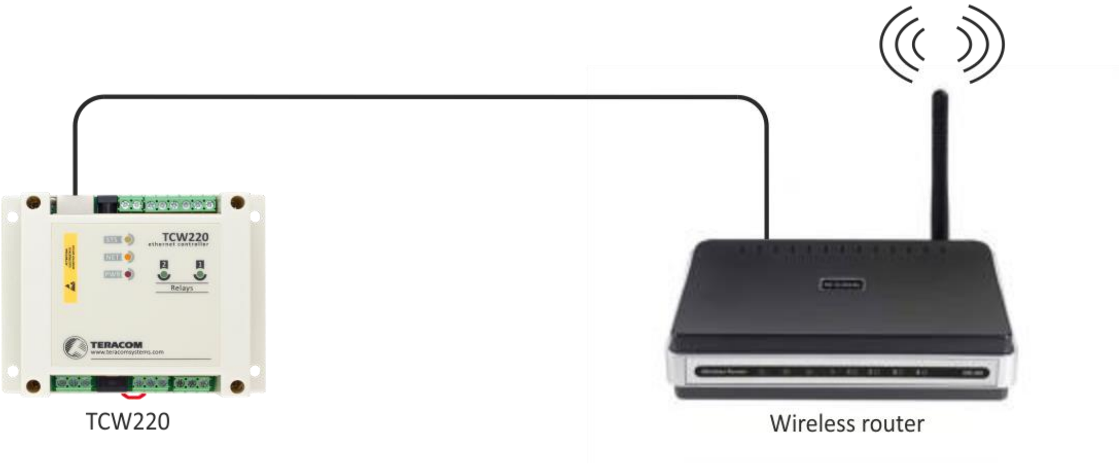
The Ethernet port of TCW220 should be connected to 10/100 Base-T Ethernet hub, switch or router.



For configuration purposes, the TCW220 can be directly connected to the Ethernet port of a computer. The device supports Auto-MDIX, so either a standard "straight-through" cable or a "crossover" cable can be used.



TCW220 can also be integrated into a wireless network by connecting it through a wireless router.

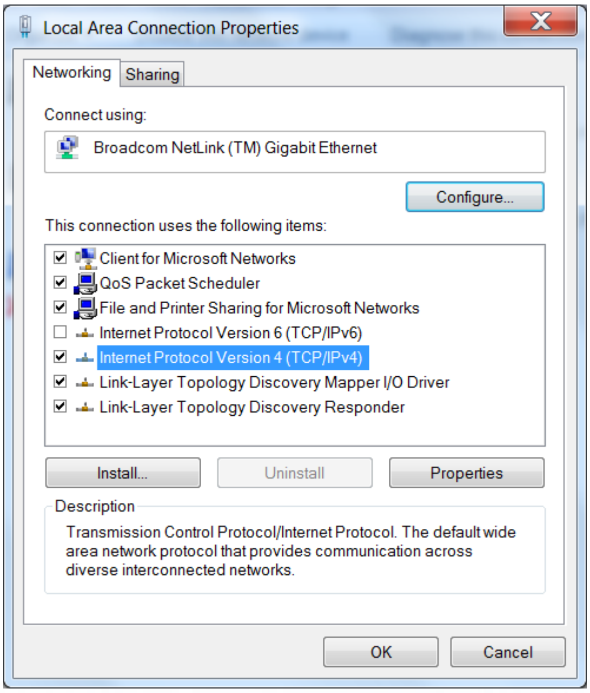


6.3. Communication setup

The TCW220 is shipped with the following default network settings:

- IP Address: 192.168.1.2
- Subnet Mask: 255.255.255.0
- Default Gateway: 192.168.1.1

To establish communication with the TCW220, you can assign a temporary IP address to your computer. For computers running Windows OS, this can be done through the “Local Area Connection Properties”:



Connect using:

Broadcom NetLink (TM) Gigabit Ethernet

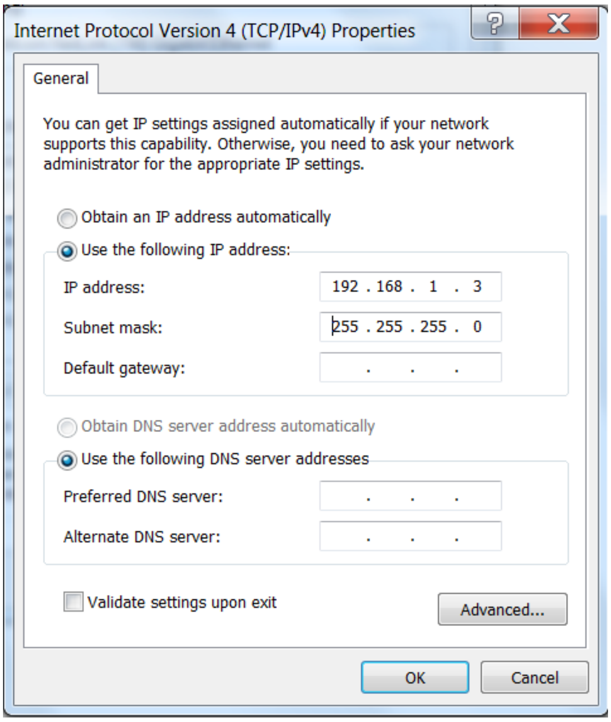
This connection uses the following items:

- ☑ Client for Microsoft Networks
- ☑ QoS Packet Scheduler
- ☑ File and Printer Sharing for Microsoft Networks
- □ Internet Protocol Version 6 (TCP/IPv6)
- ☑ Internet Protocol Version 4 (TCP/IPv4)
- ☑ Link-Layer Topology Discovery Mapper I/O Driver
- ☑ Link-Layer Topology Discovery Responder

Description

Transmission Control Protocol/Internet Protocol. The default wide area network protocol that provides communication across diverse interconnected networks.

The temporary IP address should be on the same network; for example, you can use 192.168.1.3:



Internet Protocol Version 4 (TCP/IPv4) Properties

General

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically

● Use the following IP address:

IP address:	192 . 168 . 1 . 3
Subnet mask:	255 . 255 . 255 . 0
Default gateway:	. . .

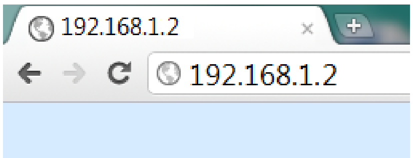
○ Obtain DNS server address automatically

● Use the following DNS server addresses

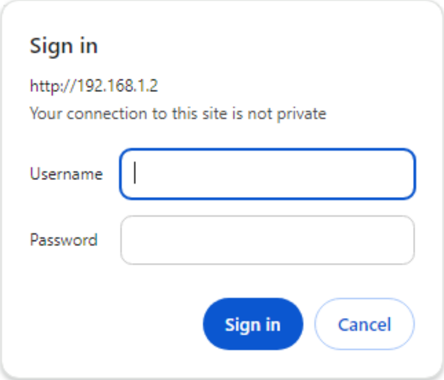
Preferred DNS server:	. . .
Alternate DNS server:	. . .

□ Validate settings upon exit

To access the web interface, enter http://192.168.1.2 into your browser's address bar:



If the network settings are configured correctly, a login pop-up window will appear:



Sign in

http://192.168.1.2

Your connection to this site is not private

Username

Password

The default authorization credentials are as follows: username: admin and password: admin.

It is highly recommended to change both the username and password to prevent unauthorized access.

All TCWxxx controllers connected to the LAN can be easily located using the free tool “TCW Discoverer.” This tool is available for both Windows and Mac operating systems and can be downloaded from www.teracomsystems.com

7. Web interface

The web interface enables configuration, monitoring, and control of the TCW220. All pages are UTF-8 encoded. Please note that the device supports HTTP only; HTTPS is not available.

The controller supports multiple active sessions simultaneously.

7.1. Monitoring

The Monitoring menu provides access to all parameters in two formats: data view (Monitoring -> Data) and graphical view (Monitoring -> Graph).

The Data page shows the current state of the TCW220 and is divided into four sections: Sensors, Digital Inputs, Analog Inputs, and Relays. Each section can be independently added or removed from the monitoring page by adjusting the settings in the "Setup -> System -> Display" menu.

Each parameter (sensor, input, relay) has a description of up to 15 characters. These default descriptions can be modified in the "Setup -> Input/Output" section.

The Monitoring page can automatically refresh at intervals between 0 and 253 seconds, with 0 meaning no automatic refresh. This interval is configurable in the "Setup -> System -> Monitoring Page Automatic Refresh" section. The default refresh interval is 1 second.

The Graph pages display all logged data in a 2D format. There are four graph pages available, and each graph page can display up to four parameters with up to two different dimensions.

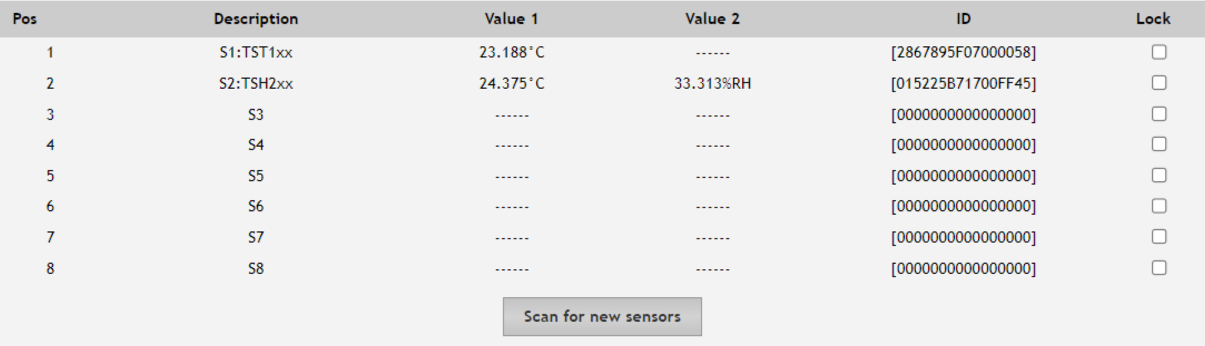
7.1.1. Sensors section

All detected 1-Wire sensors are displayed in this section. Detection can occur either upon powering on the device or by clicking the “Scan for New Sensors” button. The found sensors are listed in ascending order based on their unique ID numbers.

For each sensor, information is provided, including a description, value, and ID. The readings from Teracom 1-Wire sensors are shown in the Value 1 column. For dual sensors, such as the TSH2xx temperature/humidity sensors, the second parameter is displayed in the Value 2 column.

It is possible to lock sensors in a specific position. To do this, each sensor must be added individually. After adding a sensor, a new scan should be performed, and the newly detected sensor should be locked in its position. Once all sensors are locked, removing one from the middle will not affect the positions of the other sensors after a reset. This feature is particularly useful when the TCW220 is integrated into a monitoring and control system managed via SNMP or HTTP API commands.

For certain sensors, options for “Unit”, “Multiplier”, and “Offset” can be configured in the “Setup-Input/Output” section.

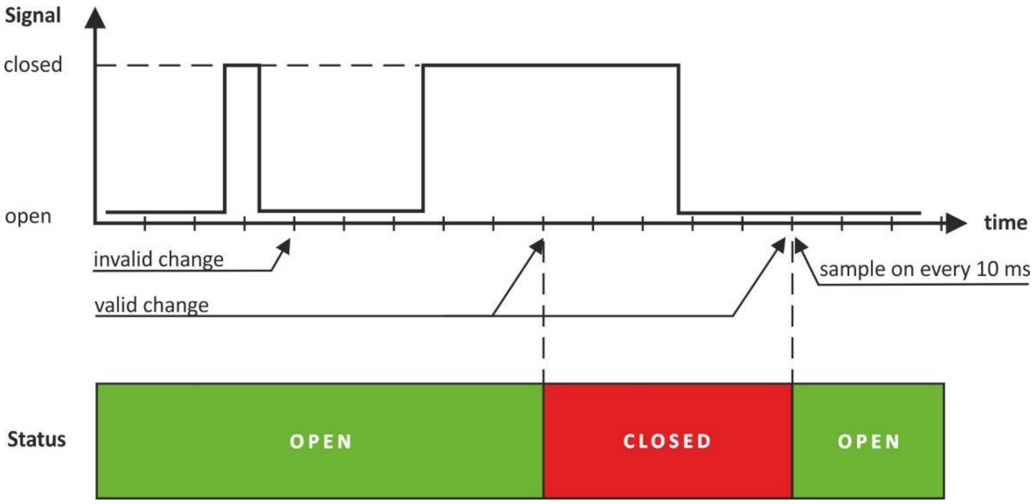


Pos	Description	Value 1	Value 2	ID	Lock
1	S1:TST1xx	23.188°C	------	[2867895F07000058]	☐
2	S2:TSH2xx	24.375°C	33.313%RH	[015225B71700FF45]	☐
3	S3	------	------	[0000000000000000]	☐
4	S4	------	------	[0000000000000000]	☐
5	S5	------	------	[0000000000000000]	☐
6	S6	------	------	[0000000000000000]	☐
7	S7	------	------	[0000000000000000]	☐
8	S8	------	------	[0000000000000000]	☐

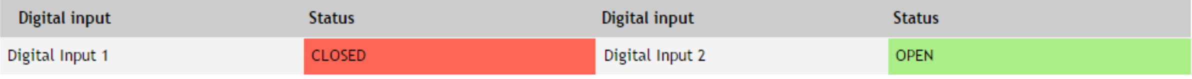
7.1.2. Digital inputs section

Digital inputs can be utilized to monitor the status of discrete devices, such as motion sensors, door contacts, relay contacts, and alarm outputs. Please note that all digital inputs are not galvanically isolated.

Each contact is connected to the "Digital In" pin on one side and the "GND" pin on the other. Digital inputs are sampled every 10 milliseconds. An input status change is considered valid if the same value is recorded in three consecutive samples (30 milliseconds), provided that the low-to-high and high-to-low delays (configured in Setup -> Conditions) are set to zero.



The status of each input is indicated by both text and color, with red representing an alarm condition.



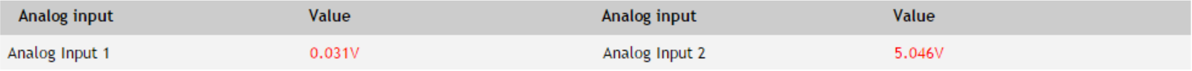
Digital input	Status	Digital input	Status
Digital Input 1	CLOSED	Digital Input 2	OPEN

Default descriptions can be modified on the “Setup -> Input/Output” page.

7.1.3. Analog inputs section

The Analog Inputs section allows monitoring of analog sensors with 0 to 5V or 0 to 10V voltage outputs.

Please note that none of the analog inputs are galvanically isolated, so take precautions when connecting sensors to avoid electrical interference or ground loops.

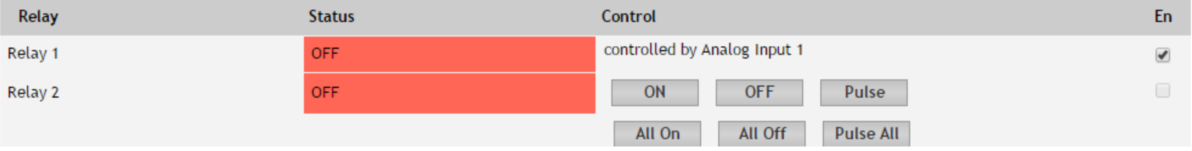


Analog input	Value	Analog input	Value
Analog Input 1	0.031V	Analog Input 2	5.046V

For each analog input, three variables can be configured in the “Setup -> Input/Output” section: “Unit”, “Multiplier”, and “Offset”.

7.1.4. Relays section

This section displays the current status of the relays and provides buttons to change their states.



Relay	Status	Control	En
Relay 1	OFF	controlled by Analog Input 1	☑
Relay 2	OFF	ON OFF Pulse	☐
		All On All Off Pulse All	

Each relay can be activated remotely via the web interface, HTTP API, or SNMP, or locally based on the status of a monitored parameter (such as a 1-Wire sensor, analog voltage, or dry contact).

Local control of the relay can be configured using either a single parameter or any "alarm" condition.

For web control, each relay features “On”, “Off”, and “Pulse” buttons. Additionally, there are “All On”, “All Off”, and “Pulse All” buttons for collective control of the relays. The pulse duration, in seconds, can be configured individually for each relay in the “Setup -> Input/Output -> Relay Outputs” section.

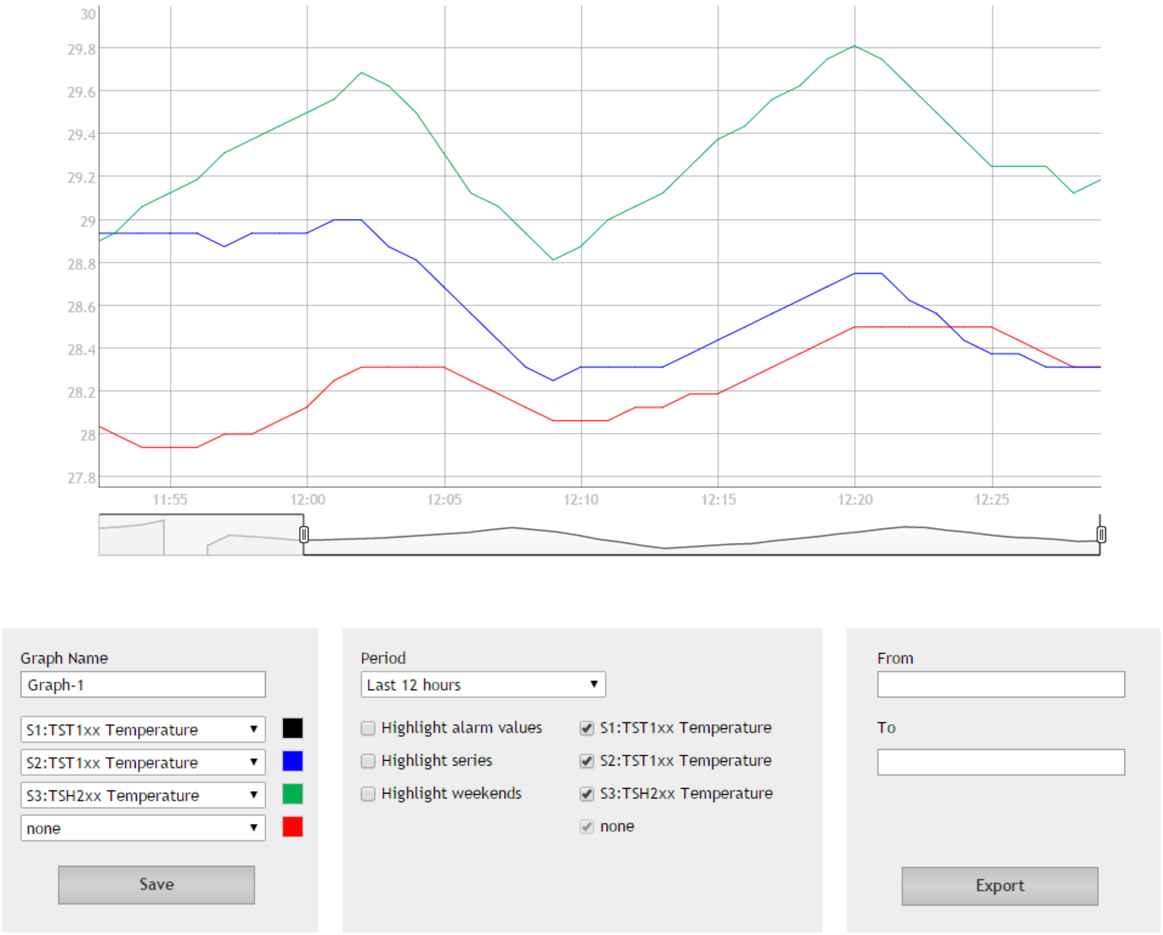
For relays activated locally, a text description of the controlling parameter is displayed instead of buttons. The parameters for local relay activation can be set in the “Setup -> Input/Output -> Relay Outputs” section. Relay control adheres to the conditions established in the “Setup -> Alarm Conditions” section.

Each locally activated relay features an "En" checkbox, which allows for the temporary disabling of automatic control. This enables manual adjustments via buttons, after which the system can return to automatic control. By default, this checkbox is turned off.

7.1.5. Graph

Each Graph page can display up to four parameters (including relay status) with up to two different dimensions.

You can assign a different color for each parameter to distinguish them easily. Additionally, there are several checkboxes available for customizing the display.



Graph Name
Graph-1

S1:TST1xx Temperature ▾ ■
S2:TST1xx Temperature ▾ 🟦
S3:TSH2xx Temperature ▾ 🟩
none ▾ 🟥

Period
Last 12 hours ▾

□ Highlight alarm values
□ Highlight series
□ Highlight weekends

☑ S1:TST1xx Temperature
☑ S2:TST1xx Temperature
☑ S3:TSH2xx Temperature
☑ none

From

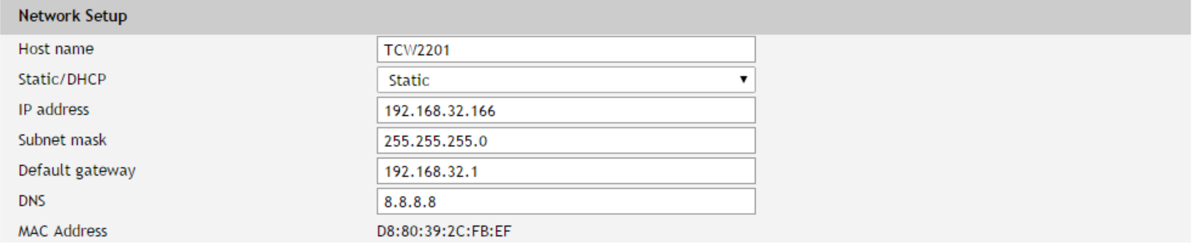
To

Monitored data can also be exported directly from the graph page for further analysis or record-keeping.

7.2. Setup

7.2.1. Network

Network parameters are configured in this section.



Network Setup	
Host name	TCW2201
Static/DHCP	Static
IP address	192.168.32.166
Subnet mask	255.255.255.0
Default gateway	192.168.32.1
DNS	8.8.8.8
MAC Address	D8:80:39:2C:FB:EF

The controller supports both static and dynamic IP addresses. It is advisable to change the default IP address of the controller immediately after the first power-on to prevent potential IP address conflicts when multiple devices are connected to the same network.

Each time a new device is connected to the network, it may be necessary to clear the ARP cache. This can be done by entering arp -d in the command prompt on the computer.

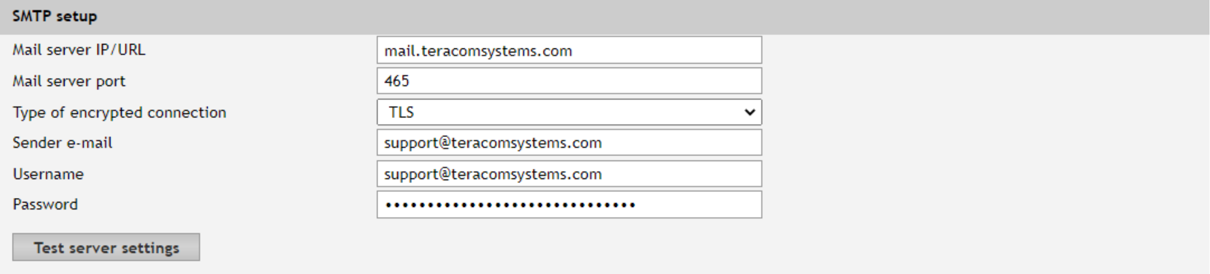
The "Hostname" can be up to 15 characters long and will appear in the search results of the TCW Discoverer tool.

Public DNS servers (such as 8.8.8.8 or 8.8.4.4) can be used instead of the default gateway.

7.2.2. SMTP

This page allows you to input the valid SMTP settings for email alerts and specify the recipients' addresses.

7.2.2.1. SMTP setup



SMTP setup	
Mail server IP/URL	mail.teracomsystems.com
Mail server port	465
Type of encrypted connection	TLS
Sender e-mail	support@teracomsystems.com
Username	support@teracomsystems.com
Password	••••••••••••••••••••••••••••••

The mail server address can be configured using either a hostname (e.g., mail.teracomsystems.com) or an IP address. By default, the SMTP port is set to 25 for non-encrypted connections. If the default port does not work, please consult your Internet Service Provider (ISP).

The sender's email, username, and password are standard authentication details, each allowing for up to 128 characters in length.

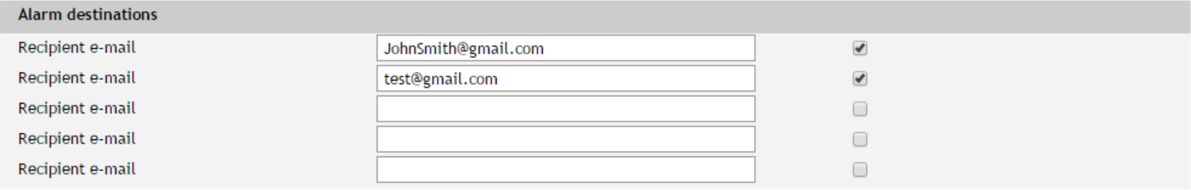
A button is available to test the server settings, providing feedback on the results. In this test, the sender and recipient email addresses are the same.

For secure communication with mail servers, the Transport Layer Security (TLS) protocol is employed. The TCW220 supports TLS versions 1.0, 1.1, and 1.2, using RSA for key exchange, agreement, and authentication. This ensures compatibility with nearly all public servers.

Please note that STARTTLS is not supported.

7.2.2.2. Alarm destination

You can configure up to five email recipients, each of whom can be activated independently using a checkbox.

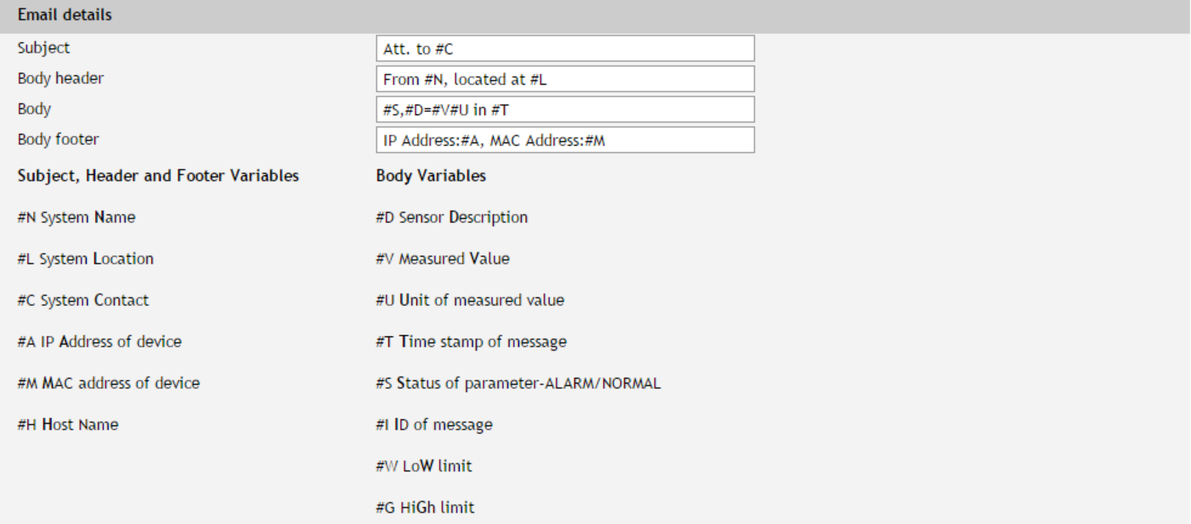


Alarm destinations

Recipient e-mail	JohnSmith@gmail.com	☑
Recipient e-mail	test@gmail.com	☑
Recipient e-mail		□
Recipient e-mail		□
Recipient e-mail		□

7.2.2.3. E-mail details

The subject, body header, body, and footer of the email can be customized using a predefined set of keys. All available keys are described on the page.



Email details

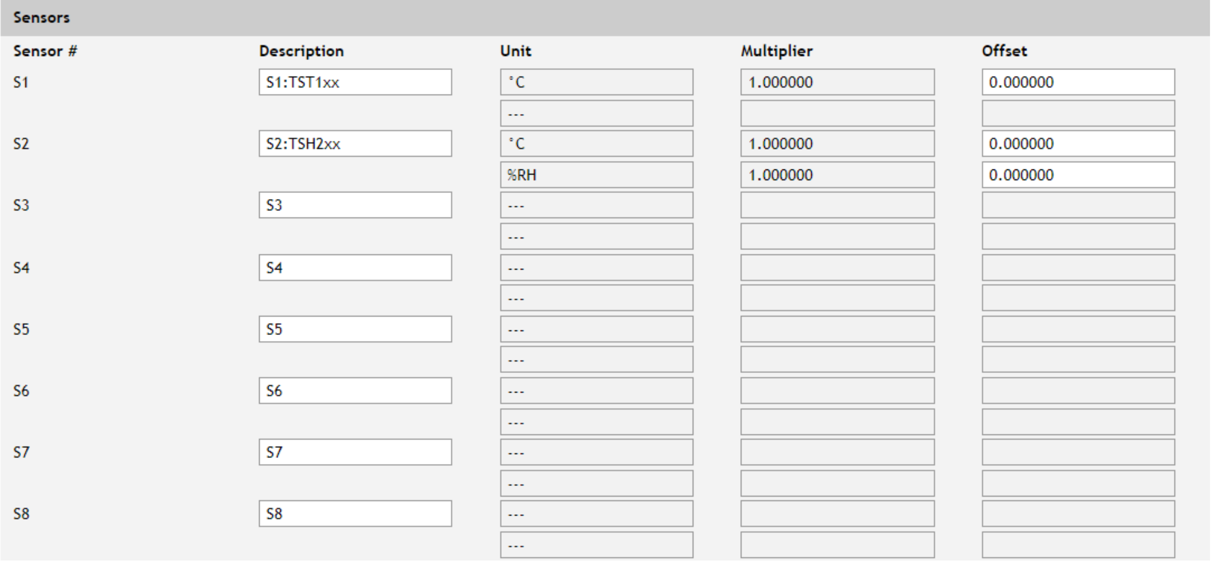
Subject	Att. to #C
Body header	From #N, located at #L
Body	#S,#D=#V#U in #T
Body footer	IP Address:#A, MAC Address:#M
Subject, Header and Footer Variables	**Body Variables**
#N System Name	#D Sensor Description
#L System Location	#V Measured Value
#C System Contact	#U Unit of measured value
#A IP Address of device	#T Time stamp of message
#M MAC address of device	#S Status of parameter-ALARM/NORMAL
#H Host Name	#I ID of message
	#W LoW limit
	#G HiGh limit

7.2.3. Input/Output

7.2.3.1. 1-Wire sensors

A description of up to 15 characters can be assigned to each 1-Wire sensor.

The "Offset" field is available for all sensors, enabling simple adjustments to the displayed value. Furthermore, for specific sensors (such as TSA200, TSV200, etc.), the "Unit" and "Multiplier" fields are also accessible.



Sensors

Sensor #	Description	Unit	Multiplier	Offset
S1	S1:TST1xx	°C	1.000000	0.000000

S2	S2:TSH2xx	°C	1.000000	0.000000
		%RH	1.000000	0.000000
S3	S3	---		

S4	S4	---		

S5	S5	---		

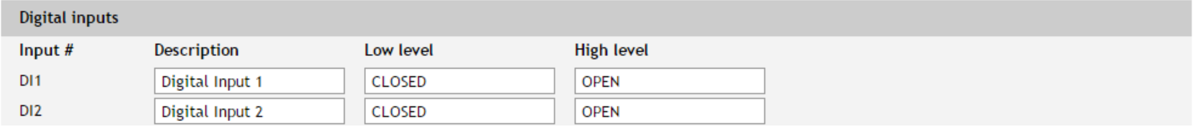
S6	S6	---		

S7	S7	---		

S8	S8	---		

7.2.3.2. Digital inputs

Each digital input can be assigned a description of up to 15 characters. The text entered in the "Low Level" and "High Level" fields will be displayed on the monitoring page for that input, with each field also allowing up to 15 characters.

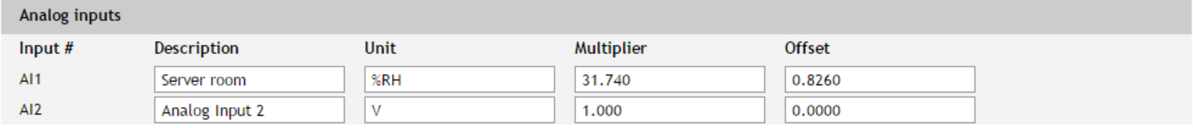


Digital inputs

Input #	Description	Low level	High level
DI1	Digital Input 1	CLOSED	OPEN
DI2	Digital Input 2	CLOSED	OPEN

7.2.3.3. Analog inputs

Each analog input can have a description of up to 15 characters assigned to it.



Analog inputs

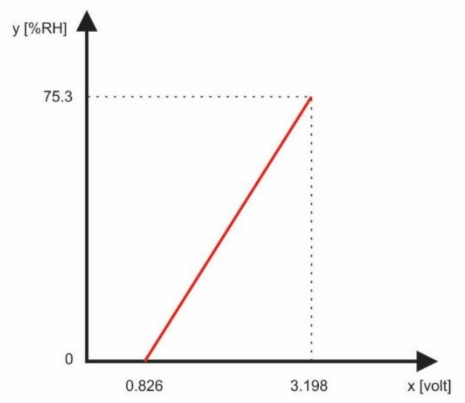
Input #	Description	Unit	Multiplier	Offset
AI1	Server room	%RH	31.740	0.8260
AI2	Analog Input 2	V	1.000	0.0000

For every analog input, the "Unit", "Multiplier", and "Offset" fields are available to convert the raw voltage or current into meaningful engineering units. The scaled value is calculated using the following formula:

SV[Un] = (RV - OF) * MU
Where:
SV – scaled (displayed) value;
Un – unit;
RV – raw voltage from the source;
MU – multiplier;
OF – offset.

Example:
For the humidity sensor HIH-4000-003, the following data is available from the datasheet:

VOUT = 0.826	at 0% RH
VOUT = 3.198	at 75.3% RH



The sensor outputs raw voltage values, but our goal is to convert these into corresponding relative humidity (RH) values. To achieve this, we use a multiplier and an offset. These parameters enable us to calculate the relative humidity for any voltage within the sensor's operational range.

Calculation of the multiplier (MU)

The multiplier (MU) is determined by the ratio of the change in relative humidity (ΔRH%) to the change in voltage (ΔV). Geometrically, this resembles finding the slope of a line. For this sensor, the line is represented by the equation ΔRH%/ΔV. We can calculate the multiplier as follows:

MU = (75.3 - 0) / (3.198 - 0.826) = 75.3 / 2.372 = 31.745 %RH/V

Calculation of the offset (OF)

The offset (OF) is calculated using the multiplier and one of the known points. By substituting the scaled value (SV) and the corresponding raw value (RV) into the equation $SV = (RV - OF) * MU$, we can solve for the offset:

$$OF = RV - SV/MU$$

Using the point where SV = 0 and RV = 0.826, we find:

$$OF = 0.826 - (0 / 31.745) = 0.826 - 0 = 0.826$$

We can also calculate the offset using the other known point, where SV = 75.3 and RV = 3.198:

$$OF = 3.198 - (75.3 / 31.745) = 3.198 - 2.372 = 0.826$$

Final formula

Thus, the formula for this sensor become:

$$SV = (RV - 0.826) * 31.745$$

Verification

To verify the accuracy of this formula, let's check the case where VOUT = 0.826 V (0%RH):

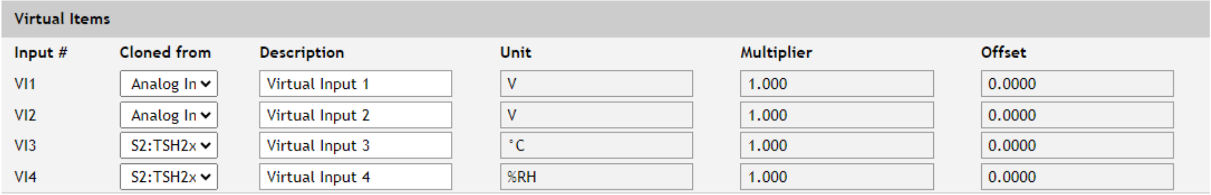
$$SV = (0.826 - 0.826) * 31.745 = 0 * 31.745 = 0 \%RH$$

This confirms that the formula accurately converts voltage readings to their corresponding relative humidity values.

By default, and following the “Factory Default Settings” procedure, the settings are as follows:

Unit	- V
Offset	- 0.00
Multiplier	- 1.00

7.2.3.4. Virtual inputs



Virtual Items

Input #	Cloned from	Description	Unit	Multiplier	Offset
VI1	Analog In	Virtual Input 1	V	1.000	0.0000
VI2	Analog In	Virtual Input 2	V	1.000	0.0000
VI3	S2:TSH2x	Virtual Input 3	°C	1.000	0.0000
VI4	S2:TSH2x	Virtual Input 4	%RH	1.000	0.0000

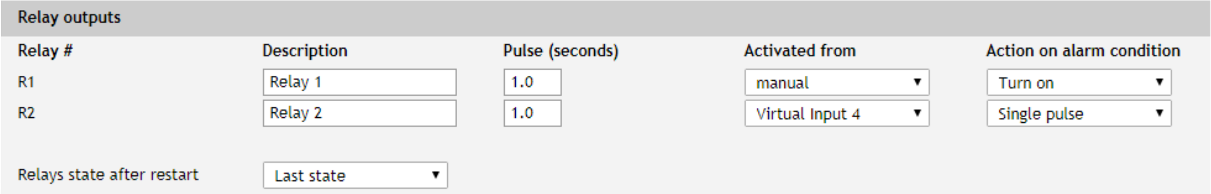
A virtual item is a useful feature that allows you to clone a monitored parameter, such as an analog input or a 1-Wire sensor. You can set different alarm thresholds for the virtual item compared to the original, enabling you to establish multiple alarm notifications for the same parameter.

The values for Unit, Multiplier, and Offset are displayed for informational purposes only. They are inherited from the original parameter and cannot be modified.

Virtual items can be utilized for alarm notifications, incorporated into Functions, and used for local relay activation.

7.2.3.5. Relay outputs

For every relay, a description up to 15 characters can be set.



Relay outputs

Relay #	Description	Pulse (seconds)	Activated from	Action on alarm condition
R1	Relay 1	1.0	manual	Turn on
R2	Relay 2	1.0	Virtual Input 4	Single pulse

Relays state after restart: Last state

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

Every relay can be activated remotely or locally – by the value of the monitored parameter.

By default, all relays are activated remotely and in the field “Activated from” is written “manual”.

For local activation, alarm conditions for different sources are used. They are set up in section “Setup-Alarm conditions”. To assign a parameter to relay, following choices are possible:

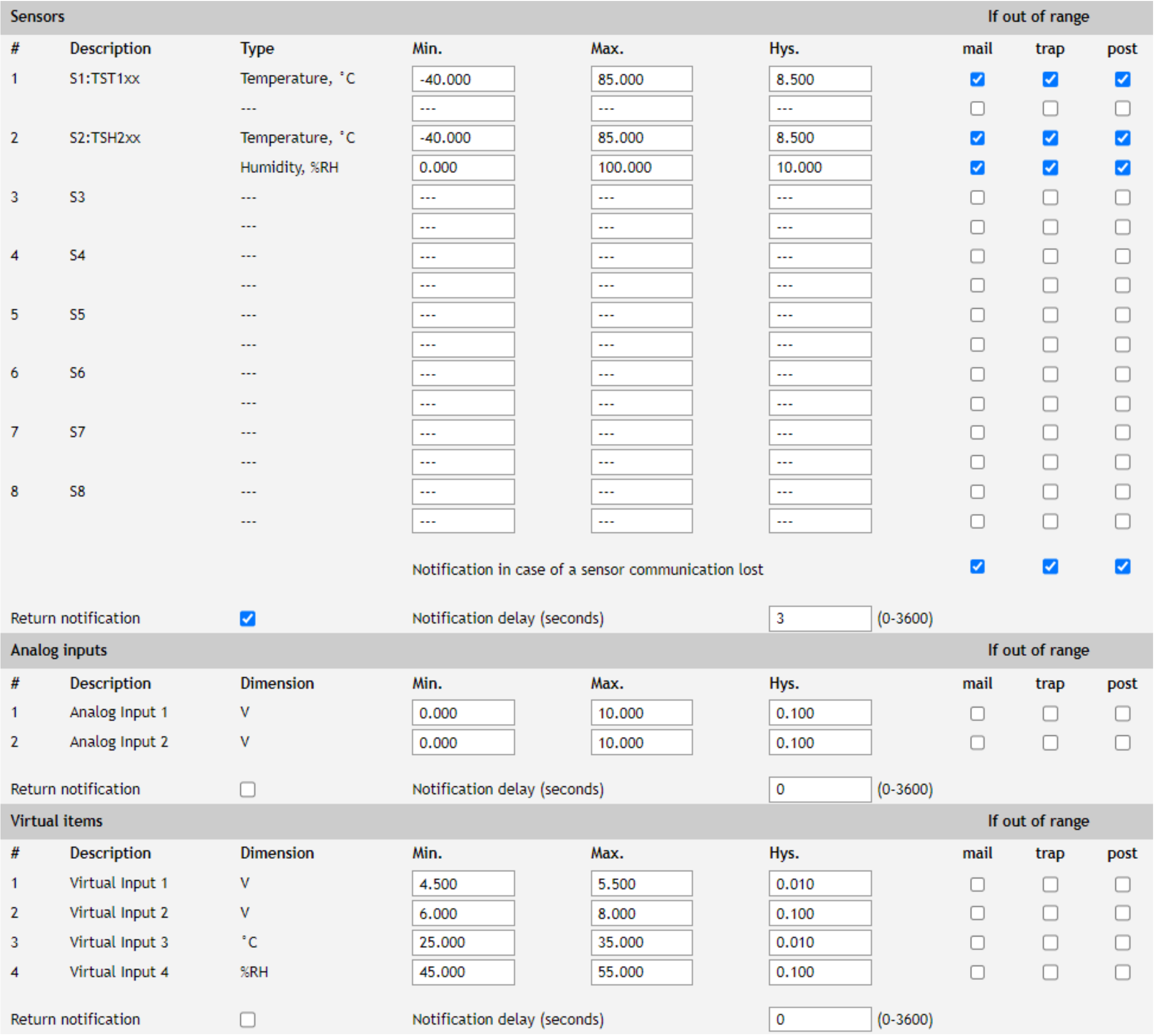
- Sxy – “S” represents a “1-Wire Sensor”, “x” is a number from 1 to 8, and “y” is a number from 1 to 2. The relay activates based on the value measured from the specified 1-Wire sensor, following the rules defined in “Setup->Alarm Conditions”;
- “Analog input z” - The relay activates from the value measured by the specified analog input, with rules based on the ranges defined in “Setup->Alarm Conditions”. Here, “z” is a number from 1 to 2;
- “Virtual input z” - The relay activates from the value of the specified virtual item (a cloned analog input or 1-Wire sensor), governed by the rules established in “Setup->Alarm Conditions”. Again, “z” is a number from 1 to 4;
- “Digital input z” - The relay follows the state of the specified digital input, where “z” is a number from 1 to 2;
- Any alarm - The relay activates based on any of the defined alarm conditions.

7.2.4. Conditions

This section is dedicated to configuring the trigger and alert conditions for 1-Wire sensors, analog inputs, virtual items, and digital inputs.

7.2.4.1. 1-Wire sensors, analog inputs and virtual items

For each sensor, there are two types of fields: one for setting trigger conditions (“Min”, “Max”, and “Hys.”) and another for alert notifications (“If out of range”).



Sensors						If out of range		
#	Description	Type	Min.	Max.	Hys.	mail	trap	post
1	S1:TST1xx	Temperature, °C	-40.000	85.000	8.500	☑	☑	☑
		---	---	---	---	☐	☐	☐
2	S2:TSH2xx	Temperature, °C	-40.000	85.000	8.500	☑	☑	☑
		Humidity, %RH	0.000	100.000	10.000	☑	☑	☑
3	S3	---	---	---	---	☐	☐	☐
		---	---	---	---	☐	☐	☐
4	S4	---	---	---	---	☐	☐	☐
		---	---	---	---	☐	☐	☐
5	S5	---	---	---	---	☐	☐	☐
		---	---	---	---	☐	☐	☐
6	S6	---	---	---	---	☐	☐	☐
		---	---	---	---	☐	☐	☐
7	S7	---	---	---	---	☐	☐	☐
		---	---	---	---	☐	☐	☐
8	S8	---	---	---	---	☐	☐	☐
		---	---	---	---	☐	☐	☐
			Notification in case of a sensor communication lost			☑	☑	☑
Return notification		☑	Notification delay (seconds)		3 (0-3600)			

Analog inputs						If out of range		
#	Description	Dimension	Min.	Max.	Hys.	mail	trap	post
1	Analog Input 1	V	0.000	10.000	0.100	☐	☐	☐
2	Analog Input 2	V	0.000	10.000	0.100	☐	☐	☐
Return notification		☐	Notification delay (seconds)		0 (0-3600)			

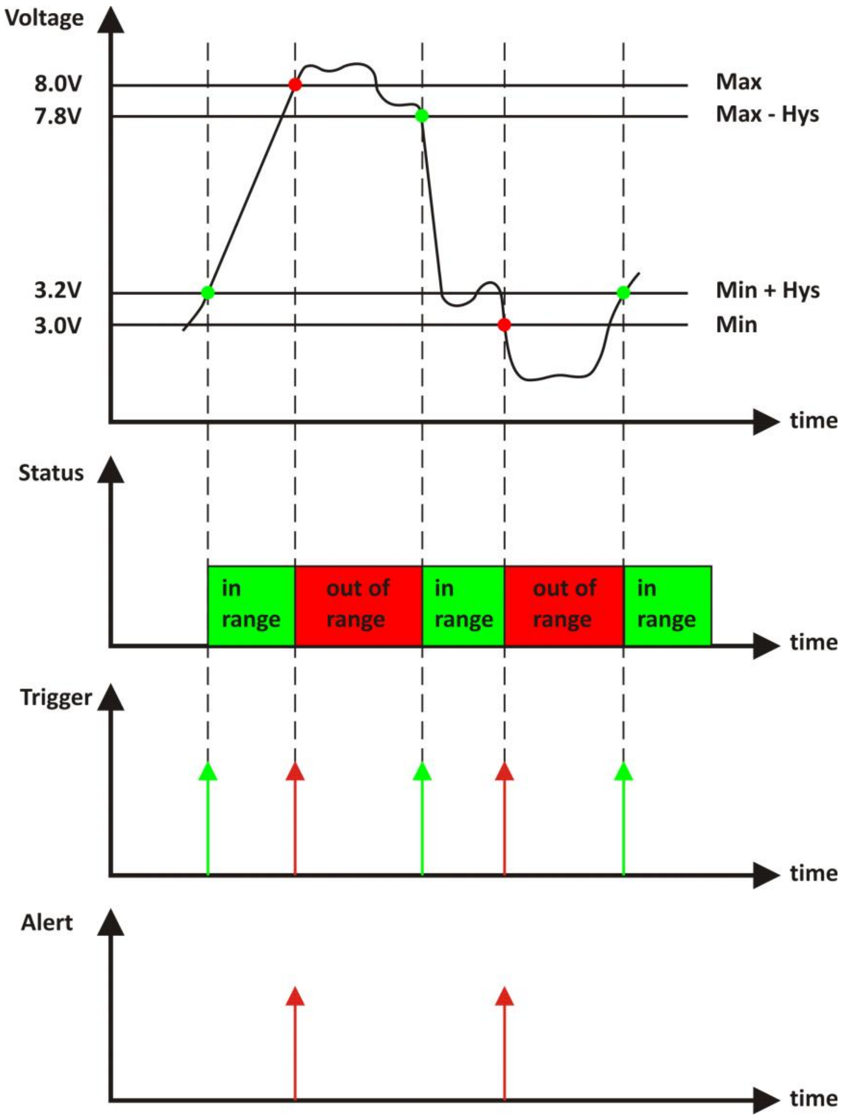
Virtual items						If out of range		
#	Description	Dimension	Min.	Max.	Hys.	mail	trap	post
1	Virtual Input 1	V	4.500	5.500	0.010	☐	☐	☐
2	Virtual Input 2	V	6.000	8.000	0.100	☐	☐	☐
3	Virtual Input 3	°C	25.000	35.000	0.010	☐	☐	☐
4	Virtual Input 4	%RH	45.000	55.000	0.100	☐	☐	☐
Return notification		☐	Notification delay (seconds)		0 (0-3600)			

“Min” and “Max” define the operational range for the monitored parameter.

A “Max” trigger condition is activated when the value exceeds the specified upper limit.

A “Min” trigger condition is activated when the value falls below the specified lower limit. In both instances, the monitored parameter is considered out of range.

Returning to the operational range is recognized when the value exceeds (Min + Hys) or falls below (Max – Hys). The hysteresis (“Hys”) is implemented to minimize unnecessary triggering caused by fluctuations around the trigger points.

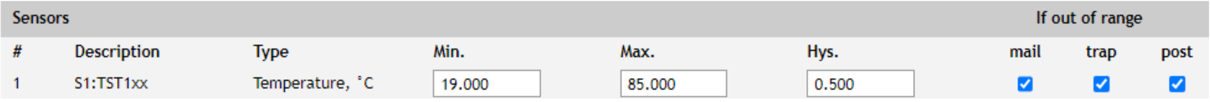


Example:

In this scenario, a TCW220 controller, a TST100 sensor, and an appropriate heater are used to manage room temperature. The desired minimum temperature is set at 19°C, while the initial temperature is 17°C. The TST100 sensor is assigned as the first 1-Wire sensor.

Local activation for Relay1 is configured based on Sensor1 with the following parameters:

Min = 19, Max = 85, and Hys = 0.5.

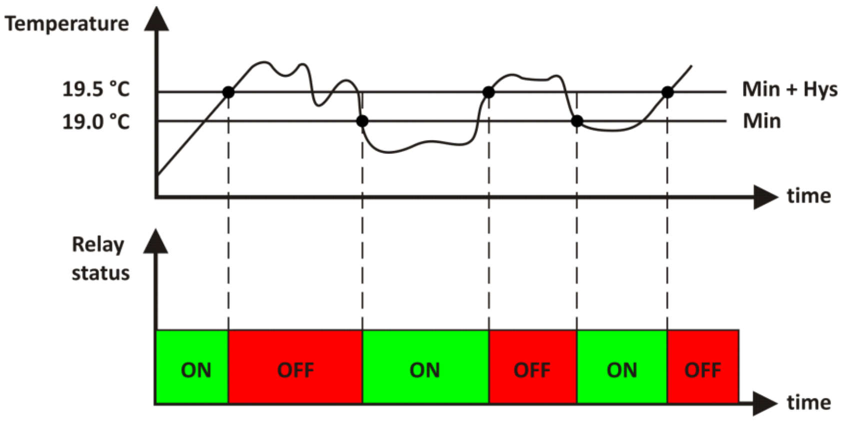


Sensors						If out of range		
#	Description	Type	Min.	Max.	Hys.	mail	trap	post
1	S1:TST1xx	Temperature, ˚C	19.000	85.000	0.500	☑	☑	☑

When the controller is powered on, Relay1 is immediately activated because the monitored temperature is below the acceptable range. This action turns on the heater, causing the temperature to rise.

As the temperature increases and reaches 19.5°C (19.0 + 0.5), it falls within the defined range (trigger condition), and Relay1 is deactivated, turning the heater off.

However, as the temperature begins to drop and reaches 19°C, it goes out of range (trigger and alert conditions), which reactivates the relay, turning the heater back on. Additionally, an email notification is sent to alert about the temperature change.



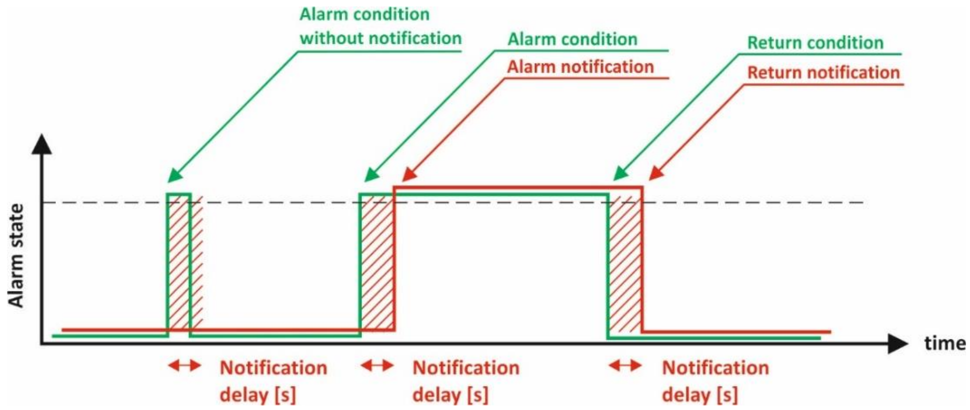
The “Max” value should be set sufficiently high above the desired temperature to prevent unnecessary trigger and alert conditions.

Each sensor or analog input has three independent alert methods available when an alarm condition occurs: email notifications, SNMP traps, and HTTP/HTTPS POSTs (XML file). Each method can be activated via a checkbox.

The “Max” value should be set sufficiently high above the desired temperature to prevent unnecessary trigger and alert conditions. Each sensor or analog input has three independent alert methods available when an alarm condition occurs: email notifications, SNMP traps, and HTTP/HTTPS POSTs (XML file). Each method can be activated via a checkbox.

There is a global checkbox labeled “Return notification” for all sensors and analog inputs. When selected, notifications will also be sent when a parameter returns to within the defined range.

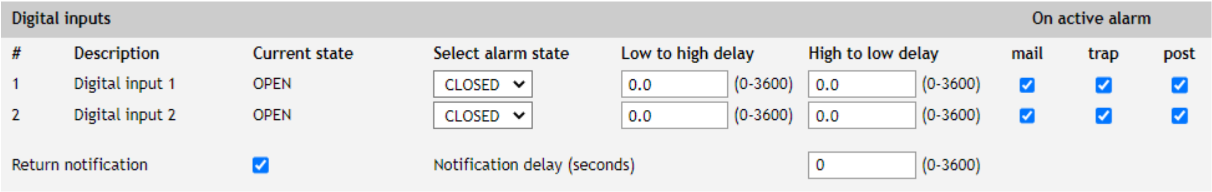
Additionally, a “Notification delay” parameter is available globally for all sensors and analog inputs. This feature acts as a filter for short alarm conditions, helping to reduce false alerts.



7.2.4.2. Digital inputs

For each digital input, you need to select the alarm state—either Open or Closed. When the input transitions to an alarm state, three independent alert methods can be activated: email notifications, SNMP traps, and HTTP/HTTPS POSTs (XML file).

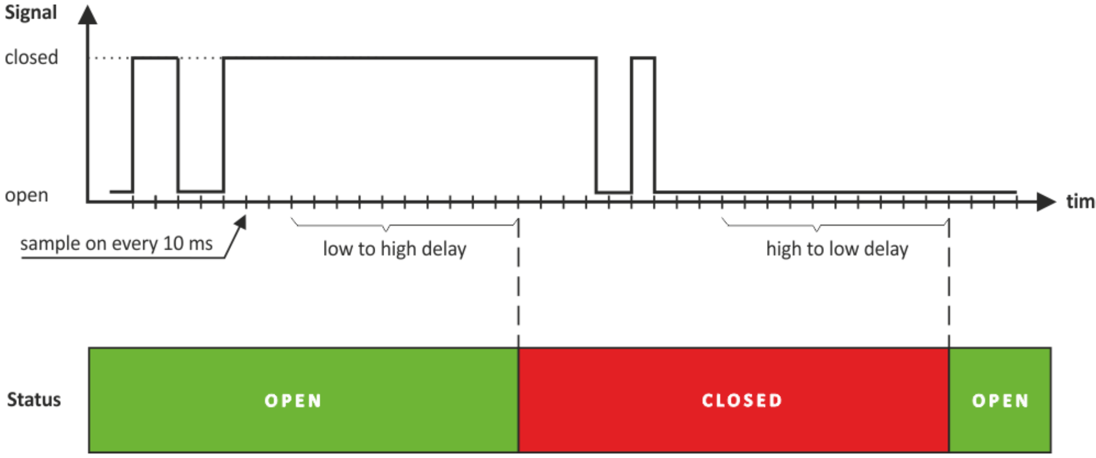
There is a global checkbox labeled “Return notification” for all digital inputs. When selected, notifications will also be sent when the parameter returns to its normal state. Additionally, a “Notification delay” parameter is available globally for all digital inputs, which serves as a filter for short alarm conditions.



Digital inputs								On active alarm		
#	Description	Current state	Select alarm state	Low to high delay		High to low delay		mail	trap	post
1	Digital input 1	OPEN	CLOSED	0.0	(0-3600)	0.0	(0-3600)	☑	☑	☑
2	Digital input 2	OPEN	CLOSED	0.0	(0-3600)	0.0	(0-3600)	☑	☑	☑
Return notification		☑	Notification delay (seconds)			0	(0-3600)			

When an input is in an alarm state, the corresponding input on the Monitoring page will be highlighted in red.

There are two delay settings for changes in digital inputs: low-to-high and high-to-low. These delays are added to the standard delay of 30 milliseconds, with a resolution of 100 milliseconds and a default setting of zero. These options can be used to provide additional debouncing.



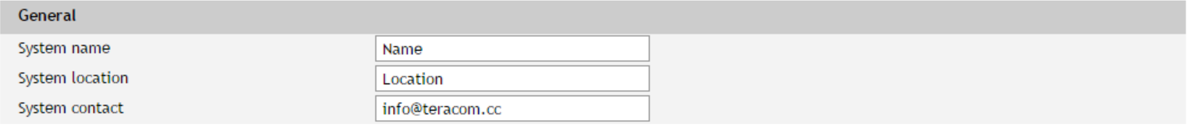
In the example shown above, both the low-to-high and high-to-low delays are set to 0.1 seconds.

7.2.5. System

This page allows you to configure various general settings for the system.

7.2.5.1. General

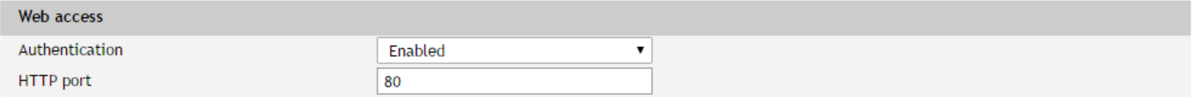
This section allows you to configure general parameters for device identification.



General	
System name	Name
System location	Location
System contact	info@teracom.cc

7.2.5.2. WEB access

This section allows you to deactivate WEB access authentication, which is enabled by default with the credentials of admin/admin. You can also change the HTTP port for WEB access, which can be useful for certain routers that do not support separate outside/inside ports for port forwarding. The default HTTP port is set to 80.



Web access	
Authentication	Enabled ▾
HTTP port	80

7.2.5.3. HTTP API

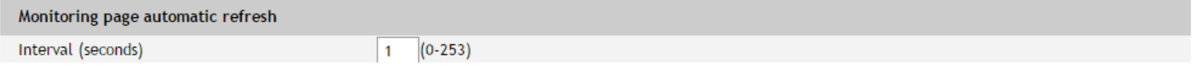
This section allows you to activate or deactivate HTTP API access authentication, which is enabled by default.

HTTP API	
Authentication	Enabled ▾

The authentication details are the same as for WEB access. The controller supports two types of authentications; refer to the explanation for HTTP API below for more information.

7.2.5.4. Monitoring page automatic refresh

The refresh interval for the monitoring page can be configured to range from 0 to 253 seconds. A setting of zero indicates that automatic refresh is disabled.

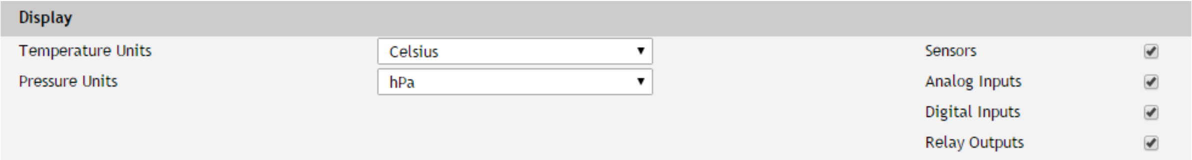


Monitoring page automatic refresh	
Interval (seconds)	1 (0-253)

7.2.5.5. Display

In this section, you can select the working units for temperature (Celsius or Fahrenheit) and pressure (hPa, mbar or mmhg).

Each section of the Monitoring Page has a checkbox; when checked, the corresponding section will be displayed.

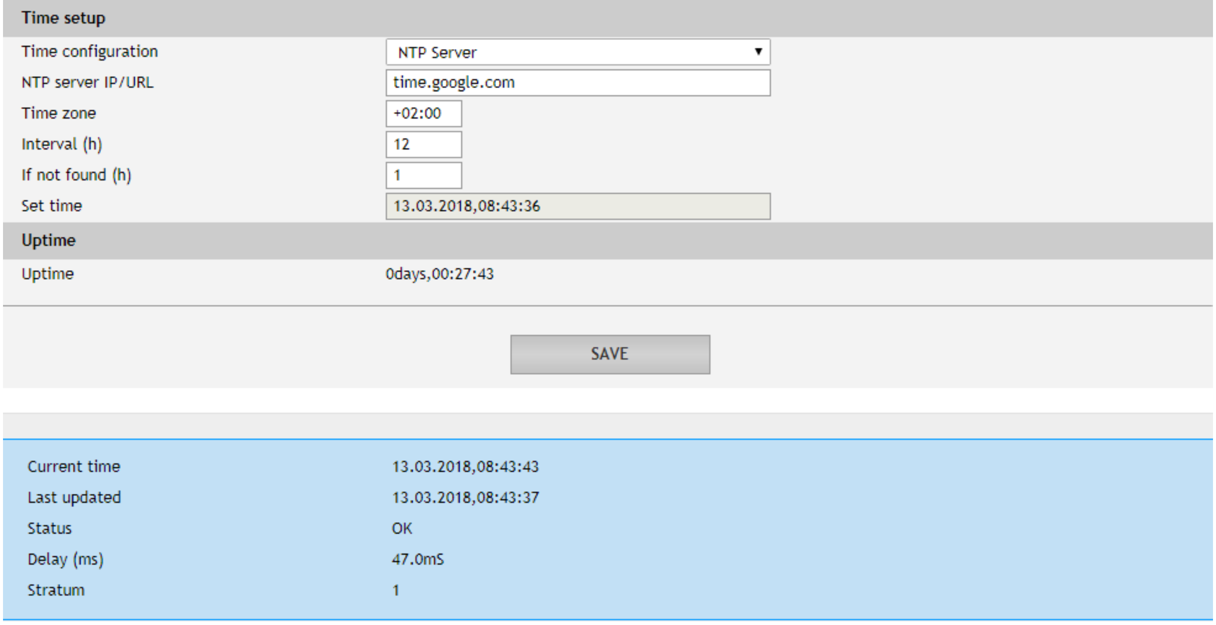


Display

Temperature Units	Celsius ▾	Sensors	☑
Pressure Units	hPa ▾	Analog Inputs	☑
		Digital Inputs	☑
		Relay Outputs	☑

7.2.6. NTP

The internal real-time clock (RTC) of the controller can be configured either manually or automatically.



Time setup

Time configuration	NTP Server ▾
NTP server IP/URL	time.google.com
Time zone	+02:00
Interval (h)	12
If not found (h)	1
Set time	13.03.2018,08:43:36

Uptime

Uptime	0days,00:27:43

Current time	13.03.2018,08:43:43
Last updated	13.03.2018,08:43:37
Status	OK
Delay (ms)	47.0mS
Stratum	1

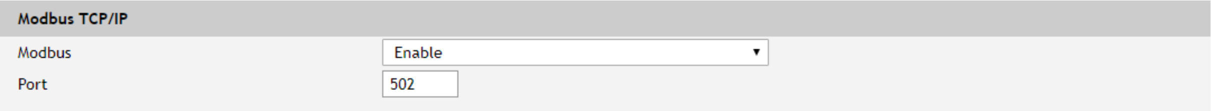
For automatic clock synchronization, the controller supports NTP (Network Time Protocol), and all necessary parameters for this synchronization are available in this section.

By default, NTP synchronization is disabled, with the server set to time.google.com, a time zone of +00:00, and an interval of 12 hours.

7.3. Services

7.3.1. Modbus TCP/IP

TCW220 supports Modbus TCP/IP via the Ethernet interface.



Modbus TCP/IP

Modbus	Enable ▾
Port	502

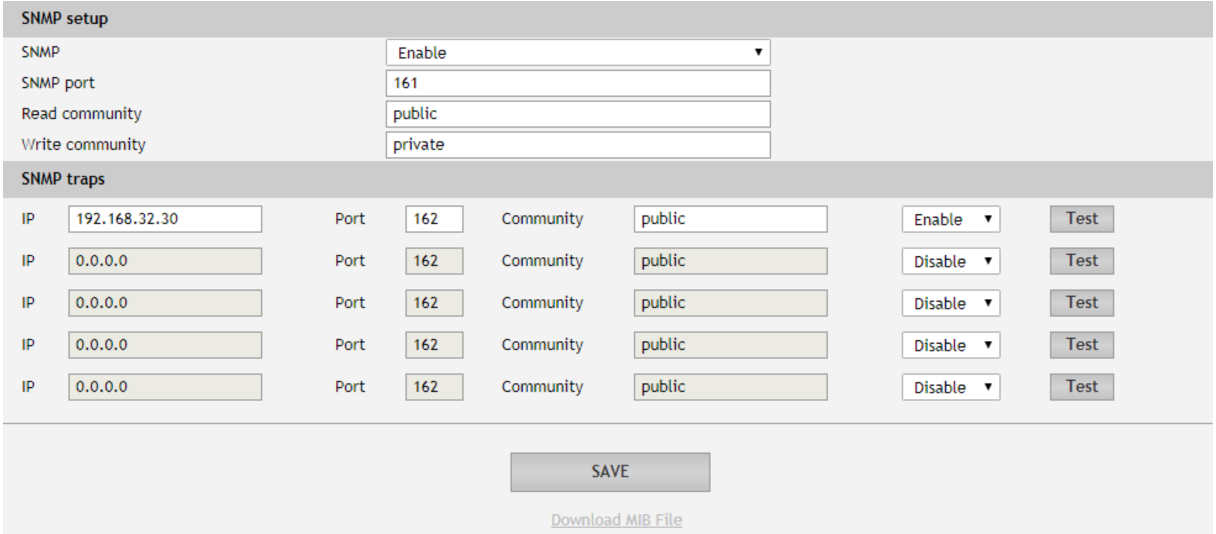
By default, Modbus TCP/IP functionality is disabled. The standard port for this protocol is 502.

A table containing the register addresses can be found in section, titled " Modbus TCP/IP”.

7.3.2. SNMP

The TCW220 supports SNMP v.2, allowing the device to integrate into monitoring and control systems using the SNMP protocol.

This section allows you to configure all necessary parameters for the proper operation of SNMP.



SNMP setup

SNMP	Enable ▾
SNMP port	161
Read community	public
Write community	private

SNMP traps

IP	192.168.32.30	Port	162	Community	public	Enable ▾	Test
IP	0.0.0.0	Port	162	Community	public	Disable ▾	Test
IP	0.0.0.0	Port	162	Community	public	Disable ▾	Test
IP	0.0.0.0	Port	162	Community	public	Disable ▾	Test
IP	0.0.0.0	Port	162	Community	public	Disable ▾	Test

By default, SNMP is disabled, with the port set to 161. The read community is set to "public" and the write community is set to "private." In the event of an alarm condition, SNMP traps can be sent to up to five independent recipients, each capable of having different ports and communities. A dedicated button is available for testing the SNMP trap functionality.

SNMP traps can be triggered under the following conditions:

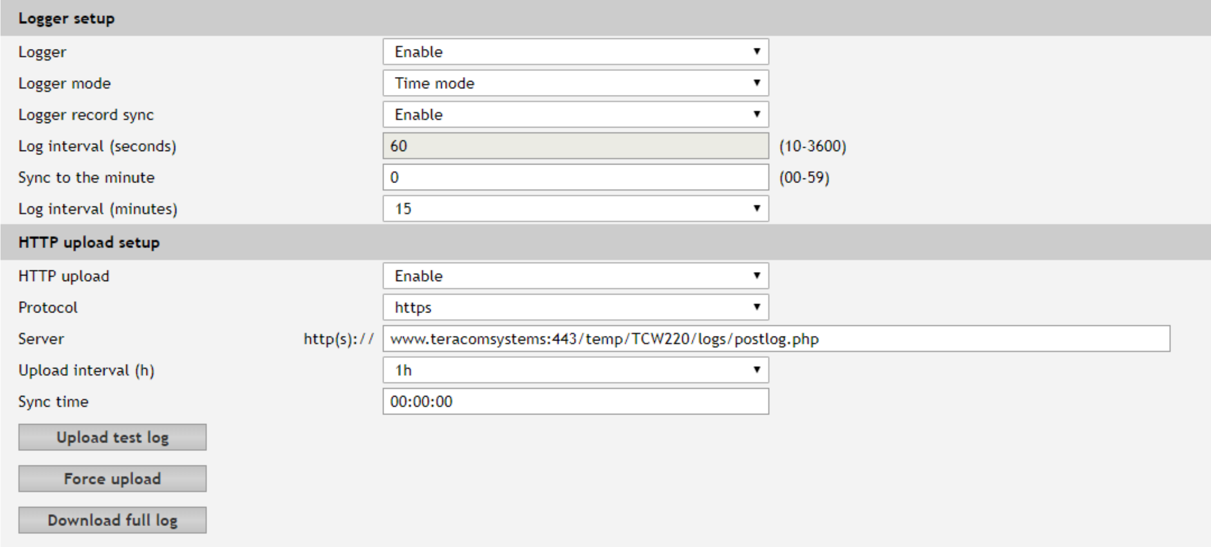
- An event occurs (status change) on Digital Inputs.
- A measured parameter on Analog Inputs falls outside the specified range.
- A measured parameter on the 1-Wire bus exceeds the defined limits.
- A restart condition is initiated.

An SNMP trap will be sent following a reset.

The current MIB file can be downloaded from here.

7.3.3. Logger

The TCW220 supports logger for all monitored parameters and the status of the relays.



Logger setup

Logger	Enable ▾	
Logger mode	Time mode ▾	
Logger record sync	Enable ▾	
Log interval (seconds)	60	(10-3600)
Sync to the minute	0	(00-59)
Log interval (minutes)	15 ▾	

HTTP upload setup

HTTP upload		Enable ▾
Protocol		https ▾
Server	http(s)://	www.teracomsystems:443/temp/TCW220/logs/postlog.php
Upload interval (h)		1h ▾
Sync time		00:00:00

The logger operates in three modes: Time, Alarm, and Time & Alarm. Each mode defines what triggers a record to be saved in the logger's memory.

- In Time mode, records are created at regular intervals defined by the "Log Interval" setting.
- In Alarm mode, records are generated whenever an alarm condition occurs.
- In Time & Alarm mode, both conditions are used to determine when records are logged.

The log interval specifies the duration between two log entries. It’s important to note that reducing the log interval increases the resolution of the data but also decreases the duration for which records can be retained.

The logger can be synchronized to a specific minute within each hour, which is particularly useful for monitoring utilities such as electricity, water, or gas meters. The log interval can be selected from a dropdown menu, allowing for values between 1 and 60 minutes. The "Sync to the Minute" field indicates which minute of each hour will be used for synchronization. While any minute can be selected, it is recommended to use the default value of 00 for optimal performance.

Example:

Current settings:

- Current time: 09:12
- Logger record sync: Enable;
- Sync to the minute: 00;
- Sync interval: 15 minutes.

With these settings, the logger will create four records per hour at HH:00, HH:15, HH:30, and HH:45.

When the device powers up, the first record will be created immediately at 09:12. Subsequent records will be logged at 09:15, 09:30, 09:45, 10:00, 10:15, and so on.

There are two methods to access the logger records:

- Download Full Log File: Use the “Download Full Log” option in the web interface*;*
- Periodic Upload: The last unsent records can be automatically uploaded to a designated HTTP server.

Records are uploaded in CSV file format using either the HTTP or HTTPS protocol. The HTTPS upload is secured using TLS 1.0, TLS 1.1, or TLS 1.2, with RSA used for key exchange and authentication.

The upload interval can be selected from the menu, ranging from 1 to 24 hours. If this service is enabled, ensure that the real-time clock (NTP service) is functioning correctly.

The HTTP server for uploads can be specified by either a domain name or an IP address, so be sure to configure the DNS settings properly.

The “Sync Time” setting determines the specific moment in the day when the upload period is synchronized.

Example:

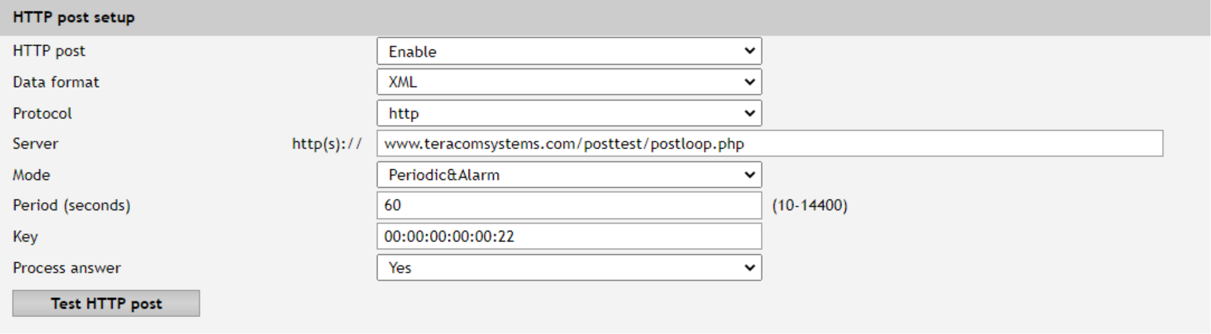
If the upload period is set to 3 hours and the sync time is 09:00, the upload schedule will be as follows: 09:00, 12:00, 15:00, 18:00, 21:00, 00:00, 03:00, and 06:00. If the current time is 19:31 and periodic upload is enabled, the first upload will occur at 21:00.

The “Force Upload” button allows you to manually initiate an upload of the recorded information between the last periodic upload and the current time.

By default, the logger is disabled. For additional information about the logger, please refer to the Data Logger section.

7.3.4. HTTP POST

TCW220 can periodically upload a file to a designated server using HTTP or HTTPS POST. The HTTPS connection utilizes TLS 1.0, TLS 1.1, and TLS 1.2 for secure key exchange and authentication via RSA. The posting interval can be set between 10 seconds and 14,400 seconds, and the supported file formats include XML and JSON.



HTTP post setup

HTTP post		Enable	
Data format		XML	
Protocol		http	
Server	http(s)://	www.teracomsystems.com/posttest/postloop.php	
Mode		Periodic&Alarm	
Period (seconds)		60	(10-14400)
Key		00:00:00:00:00:22	
Process answer		Yes	

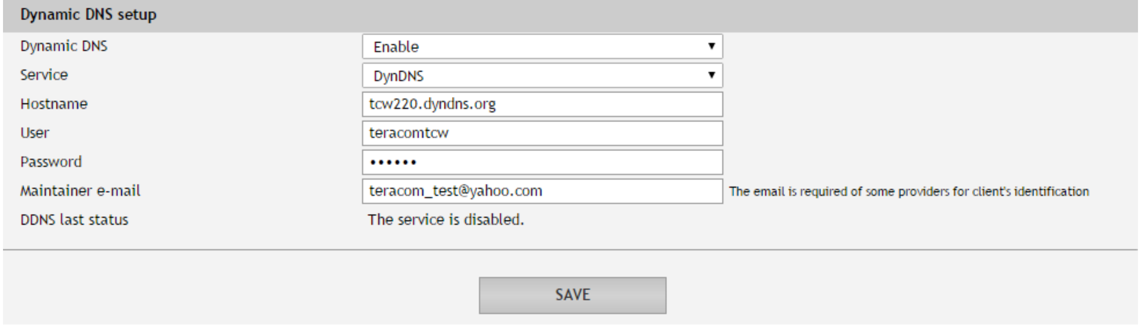
By default, the mode is set to "Periodic & Alarm." This allows for both periodic uploads and additional uploads during any alarm condition. If the "Periodic Only" mode is selected, only regular uploads will occur without any alarm-triggered POSTs. Conversely, if "Alarm Only" is chosen, only POSTs related to alarms will be executed, excluding periodic uploads.

The value in the "Key" field is included in the XML/JSON payload and can be utilized for device identification. If the "Process Answer" option is enabled, the TCW220 will handle responses from the remote server, with a list of valid commands available in the section on "HTTP API commands”.

7.3.5. Dynamic DNS

With dynamic DNS, the TCW220 can be accessed from the public Internet without the need for a broadband account with a static IP address.

TCW220 supports the following DNS services: DynDNS, No-IP, and DNS-O-Matric.



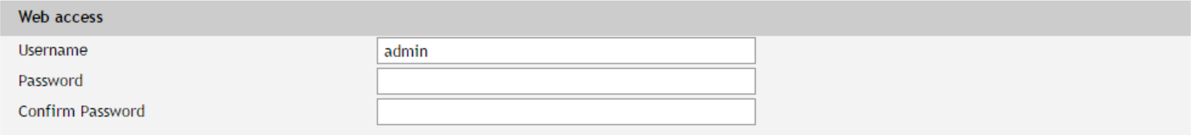
Dynamic DNS setup

Dynamic DNS	Enable	
Service	DynDNS	
Hostname	tcw220.dyndns.org	
User	teracomtcw	
Password	••••••	
Maintainer e-mail	teracom_test@yahoo.com	The email is required of some providers for client's identification
DDNS last status	The service is disabled.	

7.4. Administration

7.4.1. User/Password

The TCW220 supports a single user with administrative rights. Both the username and password can be up to 31 characters in length.

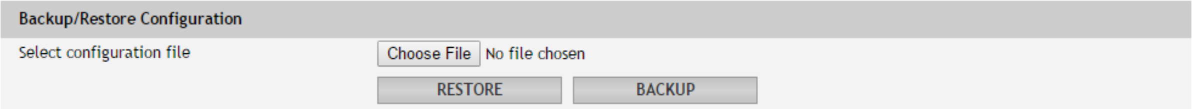


Web access

Username	admin
Password	
Confirm Password	

7.4.2. Backup/Restore

The TCW220 allows for the backup and restoration of all user settings. These settings are saved in an XML backup file, which can be utilized for restoring configurations on multiple devices. This feature is particularly useful for applying similar settings across a batch of controllers.

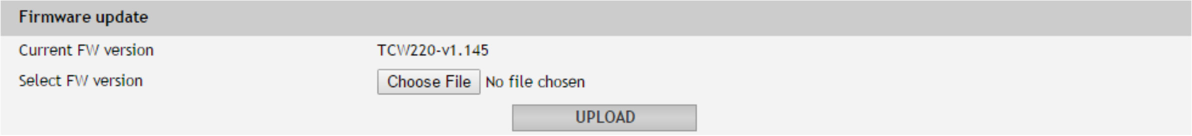


Backup/Restore Configuration

Select configuration file

7.4.3. FW update

The TCW220 can be updated through the web interface.



Firmware update

Current FW version TCW220-v1.145

Select FW version

To update the device, follow these steps:

- Go to www.teracomsystems.com and download the latest firmware;
- From Administration->FW update select downloaded .cod file and click “Upload” button;
- Once the firmware update is complete, the Login page will appear.

Important: Do not turn off the power supply during the update process, as this can damage the device.

7.5. Logout

The TCW220 supports multi-session access; however, it is recommended to log out after completing your tasks for security and best practices.

8. Protocols and API

8.1. SNMP

The Simple Network Management Protocol (SNMP) is a standard internet protocol used for managing devices on IP networks. In typical SNMP implementations, one or more administrative computers, referred to as managers, monitor and control devices on a Local Area Network (LAN). Each managed device continuously runs a software component known as an agent, which reports information via SNMP to the manager.

The TCW220 can be configured and monitored using SNMP. This can be achieved with any SNMP v.2 compatible program. The parameters that can be modified are organized by function in the tables below. To obtain a valid Object Identifier (OID) number, replace the “x” symbol with “1.3.6.1.4.1.38783.” To save the changes, set configurationSaved (OID x.2.3.5.0) to "1".

product

OID	Name	Access	Description	Syntax
x.2.1.1.0	name	read-only	Device name	DisplayString
x.2.1.2.0	version	read-only	Firmware version	DisplayString
x.2.1.3.0	date	read-only	Release date	DisplayString

setup -> network

OID	Name	Access	Description	Syntax
x.2.2.1.1.0	deviceID	read-only	Device ID (default MAC address)	MacAddress
x.2.2.1.2.0	hostName	read-only	Hostname	DisplayString
x.2.2.1.3.0	deviceIP	read-only	Device IP address	IpAddress

setup -> io -> sensorsSetup -> sensor1setup

OID	Name	Access	Description	Syntax
x.2.2.2.1.1.1.0	s1description	read-write	Sensor 1 description	DisplayString
x.2.2.2.1.1.2.1.0	s11MAXInt	read-write	S11 maximum value x1000 in Integer format	Integer32
x.2.2.2.1.1.2.2.0	s11MINInt	read-write	S11 minimum value x1000 in Integer format	Integer32
x.2.2.2.1.1.2.3.0	s11HYSTInt	read-write	S11 hysteresis value x1000 in Integer format	Integer32
x.2.2.2.1.1.3.1.0	s12MAXInt	read-write	S12 maximum value x1000 in Integer format	Integer32
x.2.2.2.1.1.3.2.0	s12MINInt	read-write	S12 minimum value x1000 in Integer format	Integer32
x.2.2.2.1.1.3.3.0	s12HYSTInt	read-write	S12 hysteresis value x1000 in Integer format	Integer32

setup -> io -> sensorsSetup -> sensor2setup

OID	Name	Access	Description	Syntax
x.2.2.2.1.2.1.0	s2description	read-write	Sensor2 description	DisplayString
x.2.2.2.1.2.2.1.0	s21MAXInt	read-write	s21 maximum value x1000 in Integer format	Integer32
x.2.2.2.1.2.2.2.0	S21MINInt	read-write	S21 minimum value x1000 in Integer format	Integer32
x.2.2.2.1.2.2.3.0	S21HYSTInt	read-write	S21 hysteresis value x1000 in Integer format	Integer32
x.2.2.2.1.2.3.1.0	S22MAXInt	read-write	S22 maximum value x1000 in Integer format	Integer32
x.2.2.2.1.2.3.2.0	S22MINInt	read-write	S22 minimum value x1000 in Integer format	Integer32
x.2.2.2.1.2.3.3.0	S22HYSTInt	read-write	S22 hysteresis value x1000 in Integer format	Integer32

setup -> io -> sensorsSetup -> sensor3setup

OID	Name	Access	Description	Syntax
x.2.2.2.1.3.1.0	S3description	read-write	Sensor 3 description	DisplayString
x.2.2.2.1.3.2.1.0	S31MAXInt	read-write	S31 maximum value x1000 in Integer format	Integer32
x.2.2.2.1.3.2.2.0	S31MINInt	read-write	S31 minimum value x1000 in Integer format	Integer32
x.2.2.2.1.3.2.3.0	S31HYSTInt	read-write	S31 hysteresis value x1000 in Integer format	Integer32
x.2.2.2.1.3.3.1.0	S32MAXInt	read-write	S32 maximum value x1000 in Integer format	Integer32
x.2.2.2.1.3.3.2.0	S32MINInt	read-write	S32 minimum value x1000 in Integer format	Integer32
x.2.2.2.1.3.3.3.0	S32HYSTInt	read-write	S32 hysteresis value x1000 in Integer format	Integer32

setup -> io -> sensorsSetup -> sensor4setup

OID	Name	Access	Description	Syntax
x.2.2.2.1.4.1.0	S4description	read-write	Sensor 4 description	DisplayString
x.2.2.2.1.4.2.1.0	S41MAXInt	read-write	S41 maximum value x1000 in Integer format	Integer32
x.2.2.2.1.4.2.2.0	S41MINInt	read-write	S41 minimum value x1000 in Integer format	Integer32
x.2.2.2.1.4.2.3.0	S41HYSTInt	read-write	S41 hysteresis value x1000 in Integer format	Integer32
x.2.2.2.1.4.3.1.0	S42MAXInt	read-write	S42 maximum value x1000 in Integer format	Integer32
x.2.2.2.1.4.3.2.0	S42MINInt	read-write	S42 minimum value x1000 in Integer format	Integer32
x.2.2.2.1.4.3.3.0	S42HYSTInt	read-write	S42 hysteresis value x1000 in Integer format	Integer32

setup -> io -> sensorsSetup -> sensor5setup

OID	Name	Access	Description	Syntax
x.2.2.2.1.5.1.0	S5description	read-write	Sensor 5 description	DisplayString
x.2.2.2.1.5.2.1.0	S51MAXInt	read-write	S51 maximum value x1000 in Integer format	Integer32
x.2.2.2.1.5.2.2.0	S51MINInt	read-write	S51 minimum value x1000 in Integer format	Integer32

x.2.2.2.1.5.2.3.0	S51HYSTInt	read-write	S51 hysteresis value x1000 in Integer format	Integer32
x.2.2.2.1.5.3.1.0	S52MAXInt	read-write	S52 maximum value x1000 in Integer format	Integer32
x.2.2.2.1.5.3.2.0	S52MINInt	read-write	S52 minimum value x1000 in Integer format	Integer32
x.2.2.2.1.5.3.3.0	S52HYSTInt	read-write	S52 hysteresis value x1000 in Integer format	Integer32

setup -> io -> sensorsSetup -> sensor6setup

OID	Name	Access	Description	Syntax
x.2.2.2.1.6.1.0	S6description	read-write	Sensor 6 description	DisplayString
x.2.2.2.1.6.2.1.0	S61MAXInt	read-write	S61 maximum value x1000 in Integer format	Integer32
x.2.2.2.1.6.2.2.0	S61MINInt	read-write	S61 minimum value x1000 in Integer format	Integer32
x.2.2.2.1.6.2.3.0	S61HYSTInt	read-write	S61 hysteresis value x1000 in Integer format	Integer32
x.2.2.2.1.6.3.1.0	S62MAXInt	read-write	S62 maximum value x1000 in Integer format	Integer32
x.2.2.2.1.6.3.2.0	S62MINInt	read-write	S62 minimum value x1000 in Integer format	Integer32
x.2.2.2.1.6.3.3.0	S62HYSTInt	read-write	S62 hysteresis value x1000 in Integer format	Integer32

setup -> io -> sensorsSetup -> sensor7setup

OID	Name	Access	Description	Syntax
x.2.2.2.1.7.1.0	S7description	read-write	Sensor 7 description	DisplayString
x.2.2.2.1.7.2.1.0	S71MAXInt	read-write	S71 maximum value x1000 in Integer format	Integer32
x.2.2.2.1.7.2.2.0	S71MINInt	read-write	S71 minimum value x1000 in Integer format	Integer32
x.2.2.2.1.7.2.3.0	S71HYSTInt	read-write	S71 hysteresis value x1000 in Integer format	Integer32
x.2.2.2.1.7.3.1.0	S72MAXInt	read-write	S72 maximum value x1000 in Integer format	Integer32
x.2.2.2.1.7.3.2.0	S72MINInt	read-write	S72 minimum value x1000 in Integer format	Integer32
x.2.2.2.1.7.3.3.0	S72HYSTInt	read-write	S72 hysteresis value x1000 in Integer format	Integer32

setup -> io -> sensorsSetup -> sensor8setup

OID	Name	Access	Description	Syntax
x.2.2.2.1.8.1.0	S8description	read-write	Sensor 8 description	DisplayString
x.2.2.2.1.8.2.1.0	S81MAXx10Int	read-write	S81 maximum value x1000 in Integer format	Integer32
x.2.2.2.1.8.2.2.0	S81MINx10Int	read-write	S81 minimum value x1000 in Integer format	Integer32
x.2.2.2.1.8.2.3.0	S81HYSTx10Int	read-write	S81 hysteresis value x1000 in Integer format	Integer32
x.2.2.2.1.8.3.1.0	S82MAXx10Int	read-write	S82 maximum value x1000 in Integer format	Integer32
x.2.2.2.1.8.3.2.0	S82MINx10Int	read-write	S82 minimum value x1000 in Integer format	Integer32
x.2.2.2.1.8.3.3.0	S82HYSTx10Int	read-write	S82 hysteresis value x1000 in Integer format	Integer32

setup -> io -> analogSetup -> analog1setup

OID	Name	Access	Description	Syntax
x.2.2.2.2.1.1.0	voltage1description	read-write	Voltage 1 description	DisplayString
x.2.2.2.2.1.2.0	voltage1max	read-write	Voltage 1 maximum	Integer32
x.2.2.2.2.1.3.0	voltage1min	read-write	Voltage 1 minimum	Integer32
x.2.2.2.2.1.4.0	voltage1hyst	read-write	Voltage 1 hysteresis	Integer32

setup -> io -> analogSetup -> analog2setup

OID	Name	Access	Description	Syntax
x.2.2.2.2.2.1.0	voltage2description	read-write	Voltage 2 description	DisplayString
x.2.2.2.2.2.2.0	voltage2max	read-write	Voltage 2 maximum	Integer32
x.2.2.2.2.2.3.0	voltage2min	read-write	Voltage 2 minimum	Integer32
x.2.2.2.2.2.4.0	voltage2hyst	read-write	Voltage 2 hysteresis	Integer32

setup -> io -> digitalSetup

OID	Name	Access	Description	Syntax
x.2.2.2.3.1.0	digitalInput1description	read-write	Digital Input 1 description"	DisplayString
x.1.2.2.3.2.0	digitalInput2description	read-write	Digital Input 2 description	DisplayString

setup -> io -> relaysSetup -> relay1setup

OID	Name	Access	Description	Syntax
x.2.2.2.4.1.1.0	relay1description	read-write	Relay 1 description	DisplayString
x.2.2.2.4.1.2.0	relay1pulseWidth	read-write	Relay1 Pulse x100ms	Integer32
x.2.2.2.4.1.3.0	relay1controlledBy	read-write	Relay1 control logic	Integer32 { manual(0),sensor11(1),sensor12(2),sensor21(3),sensor22(4), sensor31(5),sensor32(6),sensor41(7),sensor42(8),sensor51(9), sensor52(10),sensor61(11),sensor62(12),sensor71(13),sensor72(14), sensor81(15),sensor82(16),analog1(17),analog2(18), digital1(19),digital2(20),anyAlarm(21), virtual1(22),virtual2(23),virtual3(24),virtual4(25)}

setup -> io-> relaysSetup -> relay2setup

OID	Name	Access	Description	Syntax
x.2.2.2.4.2.1.0	relay2description	read-write	Relay 2 description	DisplayString
x.2.2.2.4.2.2.0	relay2pulseWidth	read-write	Relay 2 Pulse x100ms	Integer32
x.2.2.2.4.2.3.0	relay2controlledBy	read-write	Relay 2 control logic	Integer32 { manual(0),sensor11(1),sensor12(2),sensor21(3),sensor22(4), sensor31(5),sensor32(6),sensor41(7),sensor42(8),sensor51(9), sensor52(10),sensor61(11),sensor62(12),sensor71(13),sensor72(14), sensor81(15),sensor82(16),analog1(17),analog2(18), digital1(19),digital2(20),anyAlarm(21), virtual1(22),virtual2(23),virtual3(24),virtual4(25)}

setup -> io -> virtualSetup -> virtual1setup

OID	Name	Access	Description	Syntax
x.2.2.2.5.1.1.0	virtualInput1description	read-write	Virtual input 1 description	DisplayString
x.2.2.2.5.1.2.0	virtualInput1max	read-write	Virtual input 1 maximum	Integer32
x.2.2.2.5.1.3.0	virtualInput1min	read-write	Virtual input 1 minimum	Integer32
x.2.2.2.5.1.4.0	virtualInput1hyst	read-write	Virtual input 1 hysteresis	Integer32
x.2.2.2.5.1.5.0	virtualInput1Parent	read-write	Virtual input 1 parent	INTEGER{none(0),sensor11(1),sensor12(2),sensor21(3),sensor22(4),sensor31(5),sensor32(6),sensor41(7),sensor42(8),sensor51(9),sensor52(10),sensor61(11),sensor62(12),sensor71(13),sensor72(14), sensor81(15),sensor82(16),analog1(17),analog2(18)}

setup -> io -> virtualSetup -> virtual2setup

OID	Name	Access	Description	Syntax
x.2.2.2.5.2.1.0	virtualInput2description	read-write	Virtual input 2 description	DisplayString
x.2.2.2.5.2.2.0	virtualInput2max	read-write	Virtual input 2 maximum	Integer32
x.2.2.2.5.2.3.0	virtualInput2min	read-write	Virtual input 2 minimum	Integer32
x.2.2.2.5.2.4.0	virtualInput2hyst	read-write	Virtual input 2 hysteresis	Integer32
x.2.2.2.5.2.5.0	virtualInput2Parent	read-write	Virtual input 2 parent	INTEGER{none(0),sensor11(1),sensor12(2),sensor21(3),sensor22(4),sensor31(5),sensor32(6),sensor41(7),sensor42(8),sensor51(9),sensor52(10),sensor61(11),sensor62(12),sensor71(13),sensor72(14), sensor81(15),sensor82(16),analog1(17),analog2(18)}

setup -> io -> virtualSetup -> virtual3setup

OID	Name	Access	Description	Syntax
x.2.2.2.5.3.1.0	virtualInput3description	read-write	Virtual input 3 description	DisplayString
x.2.2.2.5.3.2.0	virtualInput3max	read-write	Virtual input 3 maximum	Integer32
x.2.2.2.5.3.3.0	virtualInput3min	read-write	Virtual input 3 minimum	Integer32
x.2.2.2.5.3.4.0	virtualInput3hyst	read-write	Virtual input 3 hysteresis	Integer32
x.2.2.2.5.3.5.0	virtualInput3Parent	read-write	Virtual input 3 parent	INTEGER{none(0),sensor11(1),sensor12(2),sensor21(3),sensor22(4),sensor31(5),sensor32(6),sensor41(7),sensor42(8),sensor51(9),sensor52(10),sensor61(11),sensor62(12),sensor71(13),sensor72(14), sensor81(15),sensor82(16),analog1(17),analog2(18)}

setup -> io -> virtualSetup -> virtual4setup

OID	Name	Access	Description	Syntax
x.2.2.2.5.4.1.0	virtualInput4description	read-write	Virtual input 4 description	DisplayString
x.2.2.2.5.4.2.0	virtualInput4max	read-write	Virtual input 4 maximum	Integer32
x.2.2.2.5.4.3.0	virtualInput4min	read-write	Virtual input 4 minimum	Integer32
x.2.2.2.5.4.4.0	virtualInput4hyst	read-write	Virtual input 4 hysteresis	Integer32
x.2.2.2.5.4.5.0	virtualInput4Parent	read-write	Virtual input 4 parent	INTEGER{none(0),sensor11(1),sensor12(2),sensor21(3),sensor22(4),sensor31(5),sensor32(6),sensor41(7),sensor42(8),sensor51(9),sensor52(10),sensor61(11),sensor62(12),sensor71(13),sensor72(14), sensor81(15),sensor82(16),analog1(17),analog2(18)}

monitorNcontrol -> sensors -> sensor1

OID	Name	Access	Description	Syntax
x.2.3.1.1.1.0	s11Int	read-only	S11 value x1000 in Integer format	Integer32
x.2.3.1.1.2.0	s12Int	read-only	S12 value x1000 in Integer format	Integer32
x.2.3.1.1.3.0	s1ID	read-only	S1 ID value	Mac Address

monitorNcontrol -> sensors -> sensor2

OID	Name	Access	Description	Syntax
x.2.3.1.2.1.0	s21Int	read-only	S21 value x1000 in Integer format	Integer32
x.2.3.1.2.2.0	s22Int	read-only	S22 value x1000 in Integer format	Integer32
x.2.3.1.2.3.0	s2ID	read-only	S2 ID value	OCTET STRING (SIZE (16))

monitorNcontrol -> sensors -> sensor3

OID	Name	Access	Description	Syntax
x.2.3.1.3.1.0	s31Int	read-only	S31 value x1000 in Integer format	Integer32
x.2.3.1.3.2.0	s32Int	read-only	S32 value x1000 in Integer format	Integer32
x.2.3.1.3.3.0	s3ID	read-only	S3 ID value	OCTET STRING (SIZE (16))

monitorNcontrol -> sensors -> sensor4

OID	Name	Access	Description	Syntax
x.2.3.1.4.1.0	s41Int	read-only	S41 value x1000 in Integer format	Integer32
x.2.3.1.4.2.0	s42Int	read-only	S42 value x1000 in Integer format	Integer32
x.2.3.1.4.3.0	s4ID	read-only	S4 ID value	OCTET STRING (SIZE (16))

monitorNcontrol -> sensors -> sensor5

OID	Name	Access	Description	Syntax
x.2.3.1.5.1.0	s51Int	read-only	S51 value x1000 in Integer format	Integer32
x.2.3.1.5.2.0	s52Int	read-only	S52 value x1000 in Integer format	Integer32
x.2.3.1.5.3.0	s5ID	read-only	S5 ID value	OCTET STRING (SIZE (16))

monitorNcontrol -> sensors -> sensor6

OID	Name	Access	Description	Syntax
x.2.3.1.6.1.0	s61Int	read-only	S61 value x1000 in Integer format	Integer32
x.2.3.1.6.2.0	s62Int	read-only	S62 value x1000 in Integer format	Integer32
x.2.3.1.6.3.0	s6ID	read-only	S6 ID value	OCTET STRING (SIZE (16))

monitorNcontrol -> sensors -> sensor7

OID	Name	Access	Description	Syntax
x.2.3.1.7.1.0	s71Int	read-only	S71 value x1000 in Integer format	Integer32
x.2.3.1.7.2.0	s72Int	read-only	S72 value x1000 in Integer format	Integer32
x.2.3.1.7.3.0	s7ID	read-only	S7 ID value	OCTET STRING (SIZE (16))

monitorNcontrol -> sensors -> sensor8

OID	Name	Access	Description	Syntax
x.2.3.1.8.1.0	s81Int	read-only	S81 value x1000 in Integer format	Integer32
x.2.3.1.8.2.0	s82Int	read-only	S82 value x1000 in Integer format	Integer32
x.2.3.1.8.3.0	s8ID	read-only	S8 ID value	OCTET STRING (SIZE (16))

monitorNcontrol -> analog

OID	Name	Access	Description	Syntax
x.2.3.2.1.0	voltage1Int	read-only	Voltage1 x1000 in Integer format	Integer32
x.2.3.2.2.0	voltage2Int	read-only	Voltage2 x1000 in Integer format	Integer32

monitorNcontrol -> digital

OID	Name	Access	Description	Syntax
x.2.3.3.1.0	digitalInput1State	read-only	Digital1 Input State	INTEGER {closed(0), open(1)}
x.2.3.3.2.0	digitalInput2State	read-only	Digital2 Input State	INTEGER {closed(0), open(1)}

monitorNcontrol -> relays -> relay1

OID	Name	Access	Description	Syntax
x.2.3.4.1.1.0	relay1State	read-write	Relay1 State	INTEGER {off(0), on(1)}
x.2.3.4.1.2.0	relay1Pulse	read-write	Relay1 Pulse	INTEGER {off(0), on(1)}

monitorNcontrol -> relays -> relay2

OID	Name	Access	Description	Syntax
x.2.3.4.2.1.0	relay2State	read-write	Relay2 State	INTEGER {off(0), on(1)}
x.2.3.4.2.2.0	relay2pulse	read-write	Relay2 pulse length	INTEGER {off(0), on(1)}

monitorNcontrol

OID	Name	Access	Description	Syntax
x.2.3.5.0	configurationSaved	read-write	Configuration save status SAVED/UNSAVED	INTEGER { unsaved(0), saved(1)}
x.2.3.6.0	restartDevice	read-write	Restart Device	INTEGER { cancel(0), restart(1)}
x.2.3.7.0	temperatureUnit	read-only	Unit of the all temperature values	INTEGER {celcius(0), fahrenheit(1)}
x.2.3.8.0	hardwareErr	read-only	Hardware Error	INTEGER {noErr(0), owErr(1), hwErr(2)}
x.2.3.9.0	pressureUnit	read-only	Unit of the pressure value	INTEGER {hPa(0), mbar(1), mmhg(2)}

monitorNcontrol -> virtual

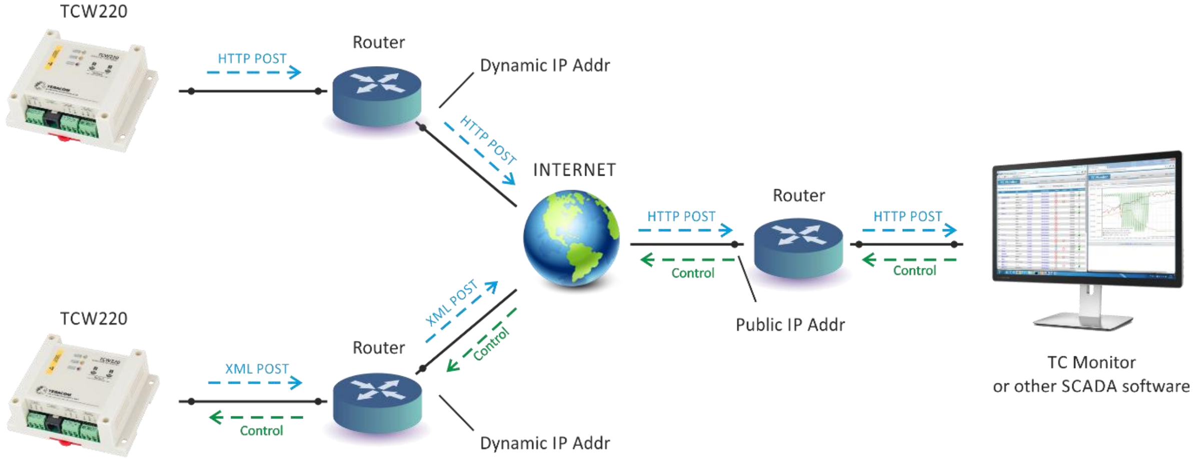
OID	Name	Access	Description	Syntax
x.2.3.11.1.0	virtualInput1Int	read-write	Virtual input 1 x1000 in Integer format	Integer32
x.2.3.11.2.0	virtualInput2Int	read-write	Virtual input 2 x1000 in Integer format	Integer32
x.2.3.11.3.0	virtualInput3Int	read-write	Virtual input 3 x1000 in Integer format	Integer32
x.2.3.11.4.0	virtualInput4Int	read-write	Virtual input 4 x1000 in Integer format	Integer32

8.2. HTTP API

8.2.1. HTTP POST

TCW220 can execute HTTP/HTTPS POST requests to upload XML or JSON files to a dedicated server. This functionality is particularly useful when the controller is situated behind a router without a public IP address or when the user lacks access to the router's configuration. It is important to note that the server should possess a public IP address.

The typical monitoring application is shown in the picture below:



HTTP/HTTPS POSTs can be sent either periodically or periodically in conjunction with alarm conditions. In response, the server can send an HTTP GET request with the appropriate command—refer to section 8.2.3 for details on HTTP API commands.

To test HTTP/HTTPS POST functionality, follow these steps:

- Save following code as post.php:

```
<?php
    define("FILENAME", 'status.xml');
    define("FOLDER", '');
    define("SEPARATOR", '');
    define("STR_SUCCESS", 'set FIN');
    define("STR_ERROR", 'error');

    if($_SERVER['REQUEST_METHOD'] == 'POST'){
        $datePrefix = date('YmdHis', strtotime('now'));
        $pathname = FOLDER.SEPARATOR.$datePrefix.'_'.FILENAME;
        $postdata = file_get_contents("php://input");
        $handle = fopen($pathname, 'w+');
        $content = var_export($postdata, true);
        fwrite($handle, substr($content, 1, strlen($content)-2));
        fclose($handle);
        echo (($handle === false) ? STR_ERROR : STR_SUCCESS)."\r\n";
    }
        else {
            echo "The PHP script is working!";
        }
?>
```

- Upload the post.php file to a public web server that supports PHP. To ensure the script is functioning correctly, enter the URL (for example, www.yourserverURL.com/post.php) in your web browser. If successful, a webpage displaying “The PHP script is working!” will appear.
- Configure the TCW220 controller to send HTTP/HTTPS POST requests to your web server. Input the URL (yourserverURL.com/post.php) in the URL field, and click the “Test HTTP POST” button.
- If the HTTP/HTTPS POST request is received and processed successfully, “OK” will be displayed next to the button. Additionally, an XML file will be created in the same directory where post.php is located. The filename will include a timestamp, formatted as 20151120103318_status.xml.

8.2.2. HTTP GET

HTTP GET can be utilized to monitor the TCW220 by retrieving XML or JSON files. The format for accessing the status is as follows:

http://device.ip.address/status.xml

http://device.ip.address/status.json

For further details regarding the structure of these files, refer to sections 8.2.4 XML file structure and 8.2.5 JSON file structure.

HTTP GET requests can be sent at any time to the TCW220 if it is on the same network or has appropriate routing. If there is no direct access to the device, an HTTP GET can be sent immediately after receiving an HTTP POST from the same device.

8.2.2.1. Commands

All commands used with HTTP POST are also applicable to HTTP GET. The correct format for sending commands is:

http://device.ip.address/status.xml?yyy=xxx

Where:

yyy is the command;

xxx is the parameter.

Example:

To turn Relay 1 ON, the following command can be used
http://device.ip.address/status.xml?ron=1.

8.2.2.2. HTTP GET authentication

If HTTP API authentication is enabled, basic access authentication is required to access the status.xml file. The format of the command is detailed in the table below:

XML/HTTP API authentication	Format
enabled	http://device.ip.address/status.xml?a=uuuu:pppp
disabled	http://device.ip.address/status.xml

Example:

To set the POST period to 120 seconds, the following command can be used, assuming the username is admin and the password is admin
http://device.ip.address/status.xml?a=admin:admin&pper=120

8.2.3. List of HTTP API commands

Command	Description

ron=**n**	Turn relay **n** ON
	(**n** is 1 or 2 for the respective relay)
	ron=1 - will turn ON relay 1
	ron=2 - will turn ON relay 2

ron=1&ron=2	Turn both relays ON

rof=**n**	Turn relay **n** OFF
	(**n** is 1 or 2 for the respective relay)
	rof=1 - will turn OFF relay 1
	rof=2 - will turn OFF relay 2

rof=1&rof=2	Turn both relays OFF

rtg=**n**	Toggle relay **n** state
	(**n** is 1 or 2 for the respective relay)
	rtg=1 - will toggle relay 1 state
	rtg=2 - will toggle relay 2 state

rpl=**n**	Pulse relay **n**
	(**n** is 1 or 2 for the respective relay)
	rpl=1 – will pulse relay 1
	rpl=2 – will pulse relay 2

vn**f**=10.0	Set Min of analog input to 10.0 (**f** is 1 or 2 for the respective input) vn1=10.0 will set Min for analog input 1
vx**f**=20.0	Set Max of analog input to 20.0 (**f** is 1 or 2 for the respective input) vx2=20.0 will set Max for analog input 2
vy**f**=1.0	Set Hys of analog input to 1.0 (**f** is 1 or 2 for the respective input) vy1=1.0 will set Hys for analog input 1
sn**pt**=30.0	Set Min of sensor to 30.0 (**p** is 1,2,3,4,5,6,7 or 8 for the respective sensor **t** is 1 or 2 for the respective parameter of sensor) sn12=30.0 will set Min for sensor 1, parameter 2
sx**pt**=40.0	Set Max of sensor to 40.0 (**p** is 1,2,3,4,5,6,7 or 8 for the respective sensor **t** is 1 or 2 for the respective parameter of sensor) sx42=40.0 will set Min for sensor 4, parameter 2

sy**pt**=2.0	Set Hys of sensor to 2.0 (**p** is 1,2,3,4,5,6,7 or 8 for the respective sensor **t** is 1 or 2 for the respective parameter of sensor) sy81=2.0 will set Hys for sensor 8, parameter 1
delsen=xxxx	Notification delay for sensors (xxxx is between 0 and 3600)
delanl=xxxx	Notification delay for analog inputs (xxxx is between 0 and 3600)
deldig=xxxx	Notification delay for digital inputs (xxxx is between 0 and 3600)
dda1=xxxx	Low to high delay for digital input 1 (xxxx is between 0 and 3600)
ddd1=xxxx	High to low delay for digital input 1 (xxxx is between 0 and 3600)
dda2=xxxx	Low to high delay for digital input 2 (xxxx is between 0 and 3600)
ddd2=xxxx	High to low delay for digital input 2 (xxxx is between 0 and 3600)
dataf=x	Data format XML/JSON for HHTP POST – 0 XML, 1 JSON
pushtls=x	HTTP/HTTPS protocol, where x is 0 for HTTP and 1 for HTTPS
purl=yyy	URL for HTTP POST to Server 1, where yyy is a full path to php file. Example: purl=212.25.45.120:30181/xampp/test/posttest.php
pper=x	HTTP POST period in seconds (x is between 10 and 14400)
dk=xxx	HTTP POST key – xxx is up to 17 characters
save	Save all previous changes (except relays’ one) in the FLASH memory. **As every save reflects the FLASH cycles (endurance), this command should be used very carefully.** pper=120&save – will set POST period to 120 seconds and save it
FIN	Terminate session. (It works with HTTP/HTTPS POST, but not with HTTP GET)

8.2.4. XML file structure

```
<Monitor>
    <DeviceInfo>
        <DeviceName>TCW220</DeviceName>
        <HostName>TCW220</HostName>
        <ID>5C:32:C5:00:69:02</ID>
        <FwVer>TCW220-v1.252</FwVer>
        <MnfInfo>www.teracomsystems.com</MnfInfo>
        <SysContact>info@teracomsystems.com</SysContact>
        <SysName>SysName</SysName>
        <SysLocation>SysLocation</SysLocation>
    </DeviceInfo>
    <S>
        <S1>
            <description>S1:TST1xx</description>
            <id>2867895F07000058</id>
            <item1>
                <value>23.313</value>
                <unit>°C</unit>
                <alarm>0</alarm>
                <min>-40.000</min>
                <max>85.000</max>
                <hys>8.500</hys>
            </item1>
            <item2>
                <value>---</value>
                <unit>---</unit>
                <alarm>0</alarm>
                <min>---</min>
                <max>---</max>
                <hys>---</hys>
            </item2>
        </S1>
        <S2>
            <description>S2:TSH2xx</description>
            <id>015225B71700FF45</id>
            <item1>
                <value>24.563</value>
                <unit>°C</unit>
                <alarm>0</alarm>
                <min>-40.000</min>
                <max>85.000</max>
                <hys>8.500</hys>
            </item1>
            <item2>
                <value>34.313</value>
                <unit>%RH</unit>
                <alarm>0</alarm>
                <min>0.000</min>
                <max>100.000</max>
                <hys>10.000</hys>
            </item2>
        </S2>
        <S3>
            <description>S3</description>
            <id>0000000000000000</id>
            <item1>
                <value>---</value>
                <unit>---</unit>
                <alarm>0</alarm>
                <min>---</min>
                <max>---</max>
                <hys>---</hys>
            </item1>
            <item2>
                <value>---</value>
                <unit>---</unit>
                <alarm>0</alarm>
                <min>---</min>
                <max>---</max>
                <hys>---</hys>
            </item2>
        </S3>
        <S4>
            <description>S4</description>
            <id>0000000000000000</id>
            <item1>
                <value>---</value>
                <unit>---</unit>
                <alarm>0</alarm>
                <min>---</min>
                <max>---</max>
                <hys>---</hys>
            </item1>
            <item2>
                <value>---</value>
                <unit>---</unit>
                <alarm>0</alarm>
                <min>---</min>
                <max>---</max>
                <hys>---</hys>
            </item2>
        </S4>
        <S5>
            <description>S5</description>
            <id>0000000000000000</id>
            <item1>
                <value>---</value>
                <unit>---</unit>
                <alarm>0</alarm>
                <min>---</min>
                <max>---</max>
                <hys>---</hys>
            </item1>
            <item2>
                <value>---</value>
                <unit>---</unit>
                <alarm>0</alarm>
                <min>---</min>
                <max>---</max>
                <hys>---</hys>
            </item2>
        </S5>
        <S6>
            <description>S6</description>
            <id>0000000000000000</id>
            <item1>
                <value>---</value>
                <unit>---</unit>
                <alarm>0</alarm>
                <min>---</min>
                <max>---</max>
                <hys>---</hys>
            </item1>
            <item2>
                <value>---</value>
                <unit>---</unit>
                <alarm>0</alarm>
                <min>---</min>
                <max>---</max>
                <hys>---</hys>
            </item2>
        </S6>
        <S7>
            <description>S7</description>
            <id>0000000000000000</id>
            <item1>
                <value>---</value>
                <unit>---</unit>
                <alarm>0</alarm>
                <min>---</min>
                <max>---</max>
                <hys>---</hys>
            </item1>
            <item2>
                <value>---</value>
                <unit>---</unit>
                <alarm>0</alarm>
                <min>---</min>
                <max>---</max>
                <hys>---</hys>
            </item2>
        </S7>
```

```
        <S8>
            <description>S8</description>
            <id>0000000000000000</id>
            <item1>
                <value>---</value>
                <unit>---</unit>
                <alarm>0</alarm>
                <min>---</min>
                <max>---</max>
                <hys>---</hys>
            </item1>
            <item2>
                <value>---</value>
                <unit>---</unit>
                <alarm>0</alarm>
                <min>---</min>
                <max>---</max>
                <hys>---</hys>
            </item2>
        </S8>
    </S>
    <AI>
        <AI1>
            <description>Analog Input 1</description>
            <value>0.033</value>
            <unit>V</unit>
            <multiplier>1.000</multiplier>
            <offset>0.0000</offset>
            <alarm>0</alarm>
            <min>0.000</min>
            <max>10.000</max>
            <hys>0.100</hys>
        </AI1>
        <AI2>
            <description>Analog Input 2</description>
            <value>0.033</value>
            <unit>V</unit>
            <multiplier>1.000</multiplier>
            <offset>0.0000</offset>
            <alarm>0</alarm>
            <min>0.000</min>
            <max>10.000</max>
            <hys>0.100</hys>
        </AI2>
    </AI>
    <VI>
        <VI1>
            <description>Virtual Input 1</description>
            <value>0.033</value>
            <unit>V</unit>
            <multiplier>1.000</multiplier>
            <offset>0.0000</offset>
            <alarm>1</alarm>
            <min>4.500</min>
            <max>5.500</max>
            <hys>0.010</hys>
        </VI1>
        <VI2>
            <description>Virtual Input 2</description>
            <value>0.033</value>
            <unit>V</unit>
            <multiplier>1.000</multiplier>
            <offset>0.0000</offset>
            <alarm>1</alarm>
            <min>6.000</min>
            <max>8.000</max>
            <hys>0.100</hys>
        </VI2>
        <VI3>
            <description>Virtual Input 3</description>
            <value>24.563</value>
            <unit>°C</unit>
            <multiplier>1.000</multiplier>
            <offset>0.0000</offset>
            <alarm>1</alarm>
            <min>25.000</min>
            <max>35.000</max>
            <hys>0.010</hys>
        </VI3>
        <VI4>
            <description>Virtual Input 4</description>
            <value>34.313</value>
            <unit>%RH</unit>
            <multiplier>1.000</multiplier>
            <offset>0.0000</offset>
            <alarm>1</alarm>
            <min>45.000</min>
            <max>55.000</max>
            <hys>0.100</hys>
        </VI4>
    </VI>
    <DI>
        <DI1>
            <description>Digital Input 1</description>
            <value>OPEN</value>
            <valuebin>1</valuebin>
            <alarmState>CLOSED</alarmState>
            <alarm>0</alarm>
        </DI1>
        <DI2>
            <description>Digital Input 2</description>
            <value>OPEN</value>
            <valuebin>1</valuebin>
            <alarmState>CLOSED</alarmState>
            <alarm>0</alarm>
        </DI2>
    </DI>
    <R>
        <R1>
            <description>Relay 1</description>
            <value>OFF</value>
            <valuebin>0</valuebin>
            <pulseWidth>0.1</pulseWidth>
            <control>0</control>
        </R1>
        <R2>
            <description>Relay 2</description>
            <value>OFF</value>
            <valuebin>0</valuebin>
            <pulseWidth>0.2</pulseWidth>
            <control>0</control>
        </R2>
    </R>
    <HTTPPush>
        <Key/>
        <PushPeriod>60</PushPeriod>
    </HTTPPush>
    <hwerr/>
    <Alarmed>1</Alarmed>
    <Scannig/>
    <Time>
        <Date>11.10.2023</Date>
        <Time>09:54:36</Time>
    </Time>
</Monitor>
```

Where:

<value>--- </value> and *<unit>--- </unit>* means no 1-Wire sensor on this position;

<alarm>1</alarm> means there is trigger condition.

8.2.5. JSON file structure

```
{
  "Monitor": {
    "DeviceInfo": {
      "DeviceName": "TCW220",
      "HostName": "TCW220",
      "ID": "5C:32:C5:00:69:02",
      "FwVer": "TCW220-v1.252",
      "MnfInfo": "www.teracomsystems.com",
      "SysContact": "info@teracomsystems.com",
      "SysName": "SysName",
      "SysLocation": "SysLocation"
    },
    "S": {
      "S1": {
        "description": "S1:TST1xx",
        "id": "2867895F07000058",
        "item1": {
          "value": "23.375",
          "unit": "°C",
          "alarm": "0",
          "min": "-40.000",
          "max": "85.000",
          "hys": "8.500"
        },
        "item2": {
          "value": "---",
          "unit": "---",
          "alarm": "0",
          "min": "---",
          "max": "---",
          "hys": "---"
        }
      },
      "S2": {
        "description": "S2:TSH2xx",
        "id": "015225B71700FF45",
        "item1": {
          "value": "24.625",
          "unit": "°C",
          "alarm": "0",
          "min": "-40.000",
          "max": "85.000",
          "hys": "8.500"
        },
        "item2": {
          "value": "34.438",
          "unit": "%RH",
          "alarm": "0",
          "min": "0.000",
          "max": "100.000",
          "hys": "10.000"
        }
      },
      "S3": {
        "description": "S3",
        "id": "0000000000000000",
        "item1": {
          "value": "---",
          "unit": "---",
          "alarm": "0",
          "min": "---",
          "max": "---",
          "hys": "---"
        },
        "item2": {
          "value": "---",
          "unit": "---",
          "alarm": "0",
          "min": "---",
          "max": "---",
          "hys": "---"
        }
      },
      "S4": {
        "description": "S4",
        "id": "0000000000000000",
        "item1": {
          "value": "---",
          "unit": "---",
          "alarm": "0",
          "min": "---",
          "max": "---",
          "hys": "---"
        },
        "item2": {
          "value": "---",
          "unit": "---",
          "alarm": "0",
          "min": "---",
          "max": "---",
          "hys": "---"
        }
      },
      "S5": {
        "description": "S5",
        "id": "0000000000000000",
        "item1": {
          "value": "---",
          "unit": "---",
          "alarm": "0",
          "min": "---",
          "max": "---",
          "hys": "---"
        },
        "item2": {
          "value": "---",
          "unit": "---",
          "alarm": "0",
          "min": "---",
          "max": "---",
          "hys": "---"
        }
      },
      "S6": {
        "description": "S6",
        "id": "0000000000000000",
        "item1": {
          "value": "---",
          "unit": "---",
          "alarm": "0",
          "min": "---",
          "max": "---",
          "hys": "---"
        },
        "item2": {
          "value": "---",
          "unit": "---",
          "alarm": "0",
          "min": "---",
          "max": "---",
          "hys": "---"
        }
      },
      "S7": {
        "description": "S7",
        "id": "0000000000000000",
        "item1": {
          "value": "---",
          "unit": "---",
          "alarm": "0",
          "min": "---",
          "max": "---",
          "hys": "---"
        },
        "item2": {
          "value": "---",
          "unit": "---",
          "alarm": "0",
          "min": "---",
          "max": "---",
          "hys": "---"
        }
      },
      "S8": {
        "description": "S8",
        "id": "0000000000000000",
        "item1": {
        "value": "---",
        "unit": "---",
        "alarm": "0",
        "min": "---",
        "max": "---",
        "hys": "---"
        },
        "item2": {
        "value": "---",
        "unit": "---",
        "alarm": "0",
        "min": "---",
        "max": "---",
        "hys": "---"
        }
      }
    },
    "AI": {
      "AI1": {
        "description": "Analog Input 1",
        "value": "0.033",
        "unit": "V",
        "multiplier": "1.000",
        "offset": "0.0000",
        "alarm": "0",
        "min": "0.000",
        "max": "10.000",
        "hys": "0.100"
      },
      "AI2": {
        "description": "Analog Input 2",
        "value": "0.033",
        "unit": "V",
        "multiplier": "1.000",
        "offset": "0.0000",
        "alarm": "0",
        "min": "0.000",
        "max": "10.000",
        "hys": "0.100"
      }
    },
    "VI": {
      "VI1": {
        "description": "Virtual Input 1",
        "value": "0.033",
        "unit": "V",
        "multiplier": "1.000",
        "offset": "0.0000",
        "alarm": "1",
        "min": "4.500",
        "max": "5.500",
        "hys": "0.010"
      },
      "VI2": {
        "description": "Virtual Input 2",
        "value": "0.033",
        "unit": "V",
        "multiplier": "1.000",
        "offset": "0.0000",
        "alarm": "1",
        "min": "6.000",
        "max": "8.000",
        "hys": "0.100"
      },
      "VI3": {
        "description": "Virtual Input 3",
        "value": "24.625",
        "unit": "°C",
        "multiplier": "1.000",
        "offset": "0.0000",
        "alarm": "1",
        "min": "25.000",
          "max": "35.000",
          "hys": "0.010"
       },
       "VI4": {
          "description": "Virtual Input 4",
          "value": "34.438",
          "unit": "%RH",
          "multiplier": "1.000",
          "offset": "0.0000",
          "alarm": "1",
          "min": "45.000",
          "max": "55.000",
          "hys": "0.100"
       }
        },
        "DI": {
       "DI1": {
          "description": "Digital Input 1",
          "value": "OPEN",
          "valuebin": "1",
          "alarmState": "CLOSED",
          "alarm": "0"
       },
       "DI2": {
          "description": "Digital Input 2",
          "value": "OPEN",
          "valuebin": "1",
          "alarmState": "CLOSED",
          "alarm": "0"
       }
        },
        "R": {
       "R1": {
          "description": "Relay 1",
          "value": "OFF",
          "valuebin": "0",
          "pulseWidth": "0.1",
          "control": "0"
       },
       "R2": {
          "description": "Relay 2",
          "value": "OFF",
          "valuebin": "0",
          "pulseWidth": "0.2",
          "control": "0"
       }
        },
        "HTTPPush": {
       "Key": "",
       "PushPeriod": "60"
        },
        "hwerr": "",
        "Alarmed": "1",
        "Scannig": "",
        "Time": {
       "Date": "11.10.2023",
       "Time": "10:07:15"
        }
     }
}
```

8.3. Modbus TCP/IP

Modbus TCP/IP is a serial communications protocol originally published by Modicon in 1979. It facilitates master-slave/client-server communication between intelligent devices. This protocol is commonly employed to connect a supervisory computer with a remote terminal unit (RTU) within supervisory control and data acquisition (SCADA) systems.

Modbus TCP/IP operates over Ethernet, enabling communication between devices such as sensors, actuators, and controllers. The protocol is widely adopted due to its simplicity and reliability in industrial and automation applications. It allows for the exchange of data and commands, making it an essential component for integrating and monitoring various devices within a networked environment.

8.3.1. Codes and answers

8.3.1.1. Read Coil Status (FC=01)

Request

This command is requesting the ON/OFF status of discrete coils on address 100.

01 0064 0001

01: The Function Code 1 (read Coil Status)
0064: The Data Address of the coil to read (0064 hex = 100)
0001: The total number of coils requested. (01 hex = 1)

Response

01 01 01

01: The Function Code 1 (read Coil Status)
01: The number of data bytes to follow
01: 7 space holders & Coils 1 (0000 0001)

Due to the number of coils requested, the last data field **01** contains the status of only 1 coil. The 7 most significant bits in this data field are filled in with zeroes. The activated relay is 1.

8.3.1.2. Force Single Coil (FC=05)

Request

This command is writing the contents of discrete on address 100 to ON.

05 0064 FF00

05: The Function Code 5 (Force Single Coil)
0064: The Data Address of the coil. (0064 hex = 100)
FF00: The status to write (FF00 = ON, 0000 = OFF)

Response

The normal response is an echo of the query, returned after the coil has been written.

05 0064 FF00

05: The Function Code 5 (Force Single Coil)
0064: The Data Address of the coil. (0064 hex = 100)
FF00: The status written (FF00 = ON, 0000 = OFF)

8.3.1.3. Read Input Status (FC=02)

Request

This command is requesting the ON/OFF status of discrete input 1

02 0064 0001

02: The Function Code 2 (read Input Status)
0064: The Data Address of the input to read (0064 hex = 100)
0001: The total number of coils requested.

Response

02 01 01

02: The Function Code 2 (read Input Status)
01: The number of data bytes to follow
01: 7 space holders & Discrete Input 100 (0000 0001)
The 7 most significant bits are filled in with zeroes.

8.3.1.4. Read Holding Registers (FC=03)

Request

This command is requesting the content of holding registers 19800.

03 4D58 0002

03: The Function Code 3 (read Holding Registers)
4D58: The Data Address of the first register requested (4D58 hex = 19800)
0002: The total number of registers requested. (read 2 registers each 2 byte = 4 bytes)

Response

03 4 41BD 0655

03: The Function Code 3 (read Sensor 1 Part 1 Holding Registers)
04: The number of data bytes to follow (2 registers x 2 bytes each = 4 bytes)
41BD 0655: 4 bytes value

All holding registers with float value are sent in big-endian.

In the example, the above value of 23.628 is sent.

Request

This command is requesting the content of holding registers 18100.

03 46B4 0020

03: Function Code 3 (read Sensor 1 description Holding Registers)
46B4: The Data Address of the first register requested (46B4 hex = 18100)
0020: The total number of registers requested (read 32 registers each 2 byte = 64 bytes)

Response

03 40 5365 6E73 6F72 3100 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000

03: Function Code 3 (read Analog Output Holding Registers)
40: The number of data bytes to follow (32 registers x 2 bytes each = 64 bytes)
5365 6E73 6F72 3100 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000: 64 bytes value
All holding registers with strings are sent in big-endian.
The answer is padded with 0.
In the example above string “Sensor1” is sent.

8.3.1.5. Exception codes

All exceptions are signaled by adding 0x80 to the function code of the request, and following this byte by a single reason byte for example as follows:

01 Illegal function

The function code received in the query is not an allowable action for the controller.

02 Illegal data address

The data address received in the query is not an allowable address for the slave. More specifically, the combination of the reference number and the transfer length is invalid. For a controller with 100 registers, a request with offset 96 and length 4 would succeed, a request with offset 96 and length 5 will generate exception 02.

8.3.2. Address table

Parameter	FC	PDU decimal address	Data size	Data
Relay 1	01,05,15	100	Discrete	
Relay 2	01,05,15	101	Discrete	
Digital input 1	02	100	Discrete	
Digital input 2	02	101	Discrete	
Relay 1 description	03,16	15000	64 bytes UTF-8	
Relay 2 description	03,16	15032	64 bytes UTF-8	
Relay 1 pulse width	03,16	15200	32-bit Float	
Relay 2 pulse width	03,16	15202	32-bit Float	
Relay 1 activated from	03,06,16	15300	16-bit unsign int	manual(0), sensor11(1), sensor12(2), sensor21(3), sensor22(4), sensor31(5), sensor32(6), sensor41(7), sensor42(8), sensor51(9), sensor52(10), sensor61(11), sensor62(12), sensor71(13), sensor72(14), sensor81(15), sensor82(16), analog1(17), analog2(18), digital1(19), digital2(20), anyAlarm(21), virtual1(22), virtual2(23), virtual3(24), virtual4(25)
Relay 2 activated from	03,06,16	15301	16-bit unsign int	—"—
Relay 1 action on alarm	03,06,16	15400	16-bit unsign int	on (0), pulse (2)
Relay 2 action on alarm	03,06,16	15401	16-bit unsign int	—"—
Relays state after restart	03,06	15500	16-bit unsign int	off (0), on (1), last state (2)

Digital input 1 description	03,16	16000	64 bytes UTF-8	
Digital input 2 description	03,16	16032	64 bytes UTF-8	
Digital input 1 alarm state	03	16200	16-bit unsign int	
Digital input 2 alarm state	03	16201	16-bit unsign int	
Analog input 1 description	03,16	17000	64 bytes UTF-8	
Analog input 2 description	03,16	17032	64 bytes UTF-8	
Analog input 1 max	03,16	17200	32-bit Float	
Analog input 2 max	03,16	17202	32-bit Float	
Analog input 1 min	03,16	17300	32-bit Float	
Analog input 2 min	03,16	17302	32-bit Float	
Analog input 1 hysteresis	03,16	17400	32-bit Float	
Analog input 2 hysteresis	03,16	17402	32-bit Float	
Analog input 1 value	03	17500	32-bit Float	
Analog input 2 value	03	17502	32-bit Float	
Sensor 1 description	03,16	18100	64 bytes UTF-8	
Sensor 2 description	03,16	18132	64 bytes UTF-8	
Sensor 3 description	03,16	18164	64 bytes UTF-8	
Sensor 4 description	03,16	18196	64 bytes UTF-8	
Sensor 5 description	03,16	18228	64 bytes UTF-8	
Sensor 6 description	03,16	18260	64 bytes UTF-8	
Sensor 7 description	03,16	18292	64 bytes UTF-8	
Sensor 8 description	03,16	18324	64 bytes UTF-8	
Sensor 1, S11 dimension	03	18400	64 bytes UTF-8	
Sensor 1, S12 dimension	03	18432	64 bytes UTF-8	
Sensor 2, S21 dimension	03	18464	64 bytes UTF-8	
Sensor 2, S22 dimension	03	18496	64 bytes UTF-8	
Sensor 3, S31 dimension	03	18528	64 bytes UTF-8	
Sensor 3, S32 dimension	03	18560	64 bytes UTF-8	
Sensor 4, S41 dimension	03	18592	64 bytes UTF-8	
Sensor 4, S42 dimension	03	18624	64 bytes UTF-8	
Sensor 5, S51 dimension	03	18656	64 bytes UTF-8	
Sensor 5, S52 dimension	03	18688	64 bytes UTF-8	
Sensor 6, S61 dimension	03	18720	64 bytes UTF-8	
Sensor 6, S62 dimension	03	18752	64 bytes UTF-8	
Sensor 7, S71 dimension	03	18784	64 bytes UTF-8	
Sensor 7, S72 dimension	03	18816	64 bytes UTF-8	
Sensor 8, S81 dimension	03	18848	64 bytes UTF-8	
Sensor 8, S82 dimension	03	18880	64 bytes UTF-8	

Sensor 1, S11 max	03,16	19200	32-bit Float	
Sensor 1, S12 max	03,16	19202	32-bit Float	
Sensor 2, S21 max	03,16	19204	32-bit Float	
Sensor 2, S22 max	03,16	19206	32-bit Float	
Sensor 3, S31 max	03,16	19208	32-bit Float	
Sensor 3, S32 max	03,16	19210	32-bit Float	
Sensor 4, S41 max	03,16	19212	32-bit Float	
Sensor 4, S42 max	03,16	19214	32-bit Float	
Sensor 5, S51 max	03,16	19216	32-bit Float	
Sensor 5, S52 max	03,16	19218	32-bit Float	
Sensor 6, S61 max	03,16	19220	32-bit Float	
Sensor 6, S62 max	03,16	19222	32-bit Float	
Sensor 7, S71 max	03,16	19224	32-bit Float	
Sensor 7, S72 max	03,16	19226	32-bit Float	
Sensor 8, S81 max	03,16	19228	32-bit Float	
Sensor 8, S82 max	03,16	19230	32-bit Float	
Sensor 1, S11 min	03,16	19300	32-bit Float	
Sensor 1, S12 min	03,16	19302	32-bit Float	
Sensor 2, S21 min	03,16	19304	32-bit Float	
Sensor 2, S22 min	03,16	19306	32-bit Float	
Sensor 3, S31 min	03,16	19308	32-bit Float	
Sensor 3, S32 min	03,16	19310	32-bit Float	
Sensor 4, S41 min	03,16	19312	32-bit Float	
Sensor 4, S42 min	03,16	19314	32-bit Float	
Sensor 5, S51 min	03,16	19316	32-bit Float	
Sensor 5, S52 min	03,16	19318	32-bit Float	
Sensor 6, S61 min	03,16	19320	32-bit Float	
Sensor 6, S62 min	03,16	19322	32-bit Float	
Sensor 7, S71 min	03,16	19324	32-bit Float	
Sensor 7, S72 min	03,16	19326	32-bit Float	
Sensor 8, S81 min	03,16	19328	32-bit Float	
Sensor 8, S82 min	03,16	19330	32-bit Float	
Sensor 1, S11 hysteresis	03,16	19400	32-bit Float	
Sensor 1, S12 hysteresis	03,16	19402	32-bit Float	
Sensor 2, S21 hysteresis	03,16	19404	32-bit Float	
Sensor 2, S22 hysteresis	03,16	19406	32-bit Float	
Sensor 3, S31 hysteresis	03,16	19408	32-bit Float	
Sensor 3, S32 hysteresis	03,16	19410	32-bit Float	
Sensor 4, S41 hysteresis	03,16	19412	32-bit Float	
Sensor 4, S42 hysteresis	03,16	19414	32-bit Float	
Sensor 5, S51 hysteresis	03,16	19416	32-bit Float	
Sensor 5, S52 hysteresis	03,16	19418	32-bit Float	
Sensor 6, S61 hysteresis	03,16	19420	32-bit Float	
Sensor 6, S62 hysteresis	03,16	19422	32-bit Float	
Sensor 7, S71 hysteresis	03,16	19424	32-bit Float	
Sensor 7, S72 hysteresis	03,16	19426	32-bit Float	

Sensor 8, S81 hysteresis	03,16	19428	32-bit Float	
Sensor 8, S82 hysteresis	03,16	19430	32-bit Float	
Sensor 1, S11 multiplier	03,16	19500	32-bit Float	
Sensor 1, S12 multiplier	03,16	19502	32-bit Float	
Sensor 2, S21 multiplier	03,16	19504	32-bit Float	
Sensor 2, S22 multiplier	03,16	19506	32-bit Float	
Sensor 3, S31 multiplier	03,16	19508	32-bit Float	
Sensor 3, S32 multiplier	03,16	19510	32-bit Float	
Sensor 4, S41 multiplier	03,16	19512	32-bit Float	
Sensor 4, S42 multiplier	03,16	19514	32-bit Float	
Sensor 5, S51 multiplier	03,16	19516	32-bit Float	
Sensor 5, S52 multiplier	03,16	19518	32-bit Float	
Sensor 6, S61 multiplier	03,16	19520	32-bit Float	
Sensor 6, S62 multiplier	03,16	19522	32-bit Float	
Sensor 7, S71 multiplier	03,16	19524	32-bit Float	
Sensor 7, S72 multiplier	03,16	19526	32-bit Float	
Sensor 8, S81 multiplier	03,16	19528	32-bit Float	
Sensor 8, S82 multiplier	03,16	19530	32-bit Float	
Sensor 1, S11 offset	03,16	19600	32-bit Float	
Sensor 1, S12 offset	03,16	19602	32-bit Float	
Sensor 2, S21 offset	03,16	19604	32-bit Float	
Sensor 2, S22 offset	03,16	19606	32-bit Float	
Sensor 3, S31 offset	03,16	19608	32-bit Float	
Sensor 3, S32 offset	03,16	19610	32-bit Float	
Sensor 4, S41 offset	03,16	19612	32-bit Float	
Sensor 4, S42 offset	03,16	19614	32-bit Float	
Sensor 5, S51 offset	03,16	19616	32-bit Float	
Sensor 5, S52 offset	03,16	19618	32-bit Float	
Sensor 6, S61 offset	03,16	19620	32-bit Float	
Sensor 6, S62 offset	03,16	19622	32-bit Float	
Sensor 7, S71 offset	03,16	19624	32-bit Float	
Sensor 7, S72 offset	03,16	19626	32-bit Float	
Sensor 8, S81 offset	03,16	19628	32-bit Float	
Sensor 8, S82 offset	03,16	19630	32-bit Float	
Sensor 1 ID	03	19700	16 bytes UTF-8	Example: 2860B85F07000094
Sensor 2 ID	03	19708	16 bytes UTF-8	
Sensor 3 ID	03	19716	16 bytes UTF-8	
Sensor 4 ID	03	19724	16 bytes UTF-8	
Sensor 5 ID	03	19732	16 bytes UTF-8	
Sensor 6 ID	03	19740	16 bytes UTF-8	
Sensor 7 ID	03	19748	16 bytes UTF-8	
Sensor 8 ID	03	19756	16 bytes UTF-8	

Sensor 1, S11 value	03	19800	32-bit Float	
Sensor 1, S12 value	03	19802	32-bit Float	
Sensor 2, S21 value	03	19804	32-bit Float	
Sensor 2, S22 value	03	19806	32-bit Float	
Sensor 3, S31 value	03	19808	32-bit Float	
Sensor 3, S32 value	03	19810	32-bit Float	
Sensor 4, S41 value	03	19812	32-bit Float	
Sensor 4, S42 value	03	19814	32-bit Float	
Sensor 5, S51 value	03	19816	32-bit Float	
Sensor 5, S52 value	03	19818	32-bit Float	
Sensor 6, S61 value	03	19820	32-bit Float	
Sensor 6, S62 value	03	19822	32-bit Float	
Sensor 7, S71 value	03	19824	32-bit Float	
Sensor 7, S72 value	03	19826	32-bit Float	
Sensor 8, S81 value	03	19828	32-bit Float	
Sensor 8, S82 value	03	19830	32-bit Float	
Sensor 1, S11 alarm status	03	20000	16-bit unsign int	normal (0), alarm (1)
Sensor 1, S12 alarm status	03	20001	16-bit unsign int	normal (0), alarm (1)
Sensor 2, S21 alarm status	03	20002	16-bit unsign int	normal (0), alarm (1)
Sensor 2, S22 alarm status	03	20003	16-bit unsign int	normal (0), alarm (1)
Sensor 3, S31 alarm status	03	20004	16-bit unsign int	normal (0), alarm (1)
Sensor 3, S32 alarm status	03	20005	16-bit unsign int	normal (0), alarm (1)
Sensor 4, S41 alarm status	03	20006	16-bit unsign int	normal (0), alarm (1)
Sensor 4, S42 alarm status	03	20007	16-bit unsign int	normal (0), alarm (1)
Sensor 5, S51 alarm status	03	20008	16-bit unsign int	normal (0), alarm (1)
Sensor 5, S52 alarm status	03	20009	16-bit unsign int	normal (0), alarm (1)
Sensor 6, S61 alarm status	03	20010	16-bit unsign int	normal (0), alarm (1)
Sensor 6, S62 alarm status	03	20011	16-bit unsign int	normal (0), alarm (1)
Sensor 7, S71 alarm status	03	20012	16-bit unsign int	normal (0), alarm (1)
Sensor 7, S72 alarm status	03	20013	16-bit unsign int	normal (0), alarm (1)
Sensor 8, S81 alarm status	03	20014	16-bit unsign int	normal (0), alarm (1)
Sensor 8, S82 alarm status	03	20015	16-bit unsign int	normal (0), alarm (1)
Save configuration	03,06	50000	16-bit unsign int	unsaved (0), saved (1)
Restart device	03,06	50001	16-bit unsign int	cancel (0), restart (1)
Temperature unit	03,06	50002	16-bit unsign int	Celsius (0), Fahrenheit (1)
HW error	03,06	50003	16-bit unsign int	noErr (0), hwErr (1)
Device ID	03	50100	18 bytes UTF-8	Example: 5c:32:c5:00:ac:52
Hostname	03	50200	16 bytes UTF-8	
Device IP	03	50300	16 bytes UTF-8	Example: 192.168.1.2

8.4. Logger

The logger uses a circular buffer in the device’s FLASH memory. When the buffer becomes full, new data automatically overwrites the oldest entries, ensuring that the FLASH memory always holds the most recent complete log. There is no command to clear the log, and a copy of the full log is always available for download.

The number of log entries depends on the length of descriptions and the type of characters used. In the worst-case scenario (with 15-byte descriptions using characters from the upper range of UTF-8), the logger can store approximately 52,000 records, which is sufficient for 36 days with records logged every minute.

New log data can also be periodically uploaded as a file to a dedicated HTTP server at time intervals of 1, 2, 3, 4, 6, 8, 12, or 24 hours. The data is uploaded in CSV format, using a semicolon (;) as the delimiter.

Each log file begins with a header row, and all rows (including the header) start with a record ID and a timestamp.

The structure of each log entry (record) is as follows:

ID	Time	Type of record	Inputs value	Relays	Alarm conditions

ID A unique 32-bit number assigned to each row (record).

Time The timestamp of the record in the format yyyy.mm.dd, hh:mm.

Type of record Specifies the reason for the log entry. The available record types are:

"Time" For periodical record;
"Event" For records triggered by an alarm condition;
"Type" For header records;
"Start" Created after a power-up event;
"Restart" Created after a reset event;
"Power Down" Generated after a power-down event;
"Bad" For problematic records.

Inputs value Lists the values of all monitored inputs in the following order: sensors, analog inputs, and digital inputs.

Relays Displays the status of the relays.

Alarm conditions Indicates the alarm status for each input. A value of "1" means an active alarm condition.

Example of the log file:

```
1131901;15.10.2015,01:02:23;Type;S11/°C;S12;S21/°C;S22;S31/°C;S32;S41/°C;S42;S51/°C;S52;S61/°C;S62;S71/°C;S72;S81/°C;S82;A1/V;A2/V;D1;D2;R1;R2;S1
1/°C;S12;S21/°C;S22;S31/°C;S32;S41/°C;S42;S51/°C;S52;S61/°C;S62;S71/°C;S72;S81/°C;S82;A1/V;A2/V;D1;D2;
1131902;15.10.2015,01:02:23;Time;18.250;;18.375;;18.125;;18.500;;18.188;;18.125;;18.375;;18.375;;11.352;0.065;1;0;1;0;1;;1;;1;;1;;1;;1;;1;;1;;1;0;1;0;
1131903;15.10.2015,01:02:23;Event;18.250;;18.438;;18.125;;18.500;;18.188;;18.125;;18.313;;18.375;;11.352;0.066;0;1;0;1;1;;1;;1;;1;;1;;1;;1;;1;0;0;1;
1131904;15.10.2015,01:02:24;Time;18.250;;18.438;;18.125;;18.500;;18.188;;18.125;;18.313;;18.375;;11.352;0.066;0;1;0;1;1;;1;;1;;1;;1;;1;;1;;1;0;0;1;
1131905;15.10.2015,01:02:25;Time;18.250;;18.375;;18.125;;18.500;;18.188;;18.125;;18.313;;18.375;;11.352;0.066;0;1;0;1;1;;1;;1;;1;;1;;1;;1;;1;0;0;1;
1131906;15.10.2015,01:02:26;Time;18.250;;18.375;;18.125;;18.500;;18.188;;18.125;;18.313;;18.313;;11.352;0.066;0;1;0;1;1;;1;;1;;1;;1;;1;;1;;1;0;0;1;
1131907;15.10.2015,01:02:27;Time;18.250;;18.375;;18.125;;18.438;;18.188;;18.125;;18.313;;18.313;;11.352;0.066;0;1;0;1;1;;1;;1;;1;;1;;1;;1;;1;0;0;1;
1131908;15.10.2015,01:02:27;Event;18.250;;18.375;;18.125;;18.438;;18.188;;18.125;;18.313;;18.313;;2.198;9.092;0;1;0;1;1;;1;;1;;1;;1;;1;;1;;0;0;0;1;
```

9. Factory default settings

TCW220 can be restored to its original factory default settings in three different ways.

9.1. Factory default from the WEB interface

Pressing the “Factory default” button in the Administration -> Backup/Restore section will reset all parameters to their factory defaults, except for network settings.

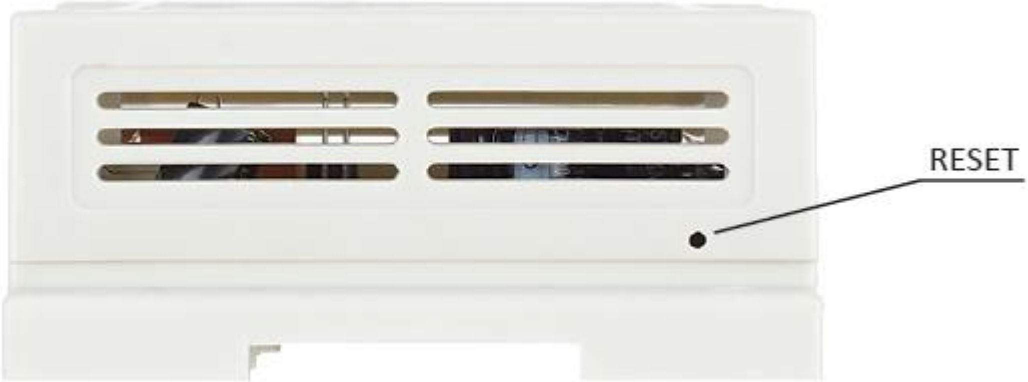
9.2. Factory default with the reset button

If the reset button is pressed for more than 5 seconds while the device is powered on, all network settings will be reverted to factory defaults.

9.3. General factory default with the reset button

To perform a complete factory reset of all parameters, follow these steps:

- Press and hold the RESET button, then turn on the power supply.
- The yellow LED will illuminate, and the red LED will blink approximately five times per second.
- After about 5 seconds, the red LED will turn off, and you can release the button.
- The yellow LED will flash once per second while the red LED remains illuminated, indicating that the device is in working mode with factory default settings.



The factory default settings are:

User Name	admin
Password	admin
IP Address	192.168.1.2
Subnet Mask	255.255.255.0
Default Gateway	192.168.1.1
SNMPConfiguration	disabled
readCommunity	public
writeCommunity	private
Analog inputs unit	voltage
Analog inputs multiplier	1.000
Analog inputs ofset	0.000

10. Environment information

This equipment is intended for use in a Pollution Degree 2 environment, at altitudes of up to 2000 meters.

When the controller is part of a larger system, all other elements of the system must comply with EMC (Electromagnetic Compatibility) requirements and be suitable for use under the same ambient conditions.

11. Safety

This device must not be used for medical or life-saving purposes or for any application where its failure could result in serious injury or loss of life.

To reduce the risk of fire, use only flexible stranded wire with a cross-section of 0.5mm² or larger for wiring digital and analog inputs and relay outputs of the device.

To avoid electric shock and fire hazards, do not expose this product to liquids, rain, or moisture. Objects filled with liquids, such as vases, should not be placed on this device.

There is a risk of overheating (and potential damage) to the controller if the recommended free spaces next to adjacent devices are not maintained. Ensure that there is sufficient space for attaching and removing cables after installation.

Teracom does not guarantee the successful operation of the product if it is used under conditions that deviate from the product specifications.

To ensure that the device works correctly follow these steps:

- Ensure that the device is installed according to the instructions in this user manual;
- Log into the device via a web browser;
- Perform the necessary configuration settings;
- Short the “Din1” and “GND” terminals;
- Install a TSH2XX or TST1XX sensor on the 1-Wire bus;
- Navigate to the “Monitoring page” of the web interface. Proper parameter values should be displayed, and the flashing “STS” LED should indicate that the device is operating correctly.

If the equipment is used in a manner not specified by the manufacturer, the protection provided by the equipment may be impaired.

In no event will Teracom Ltd. be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment.

12. Maintenance

After any service or repair of the device, or at least once a year, a safety check must be performed to ensure that the product is in proper operating condition.

Clean the device only with dry cloth. Do not use liquid cleaners or aerosol cleaners. Do not use a magnetic/static cleaning device (dust remover) or any kind of abrasive materials to clean the device.

Following these guidelines will help maintain the device in optimal condition and ensure its longevity.

Appendix A

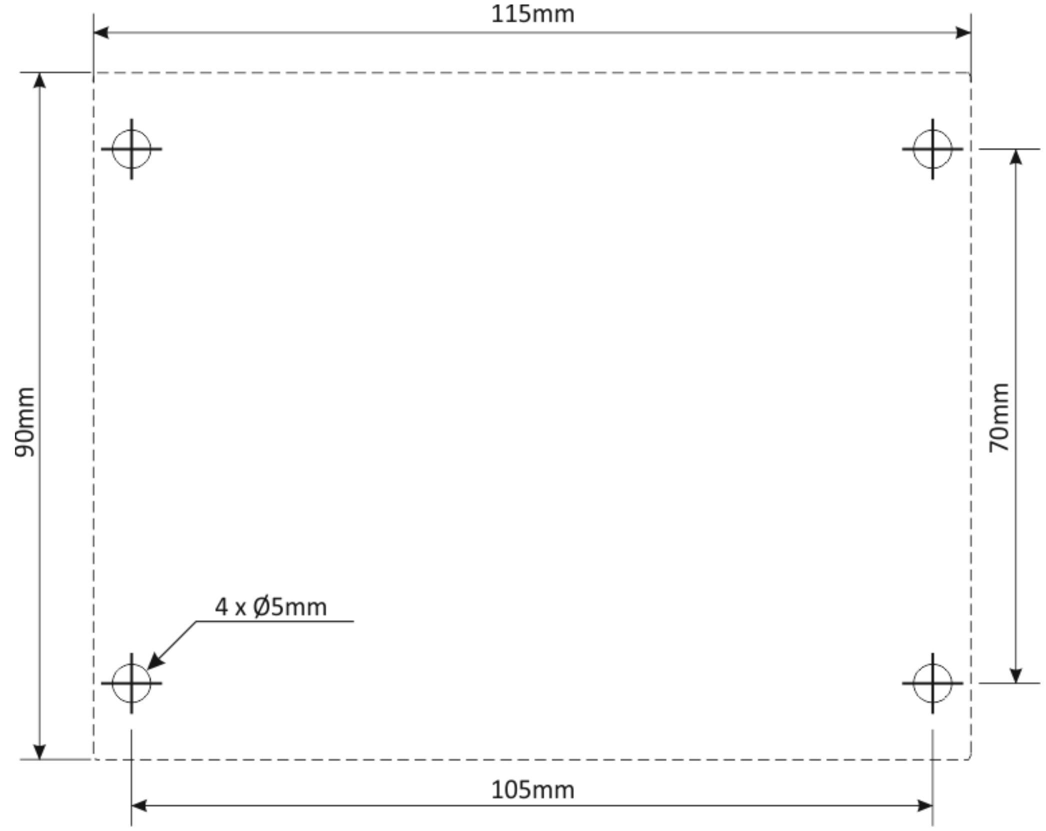


Fig.1

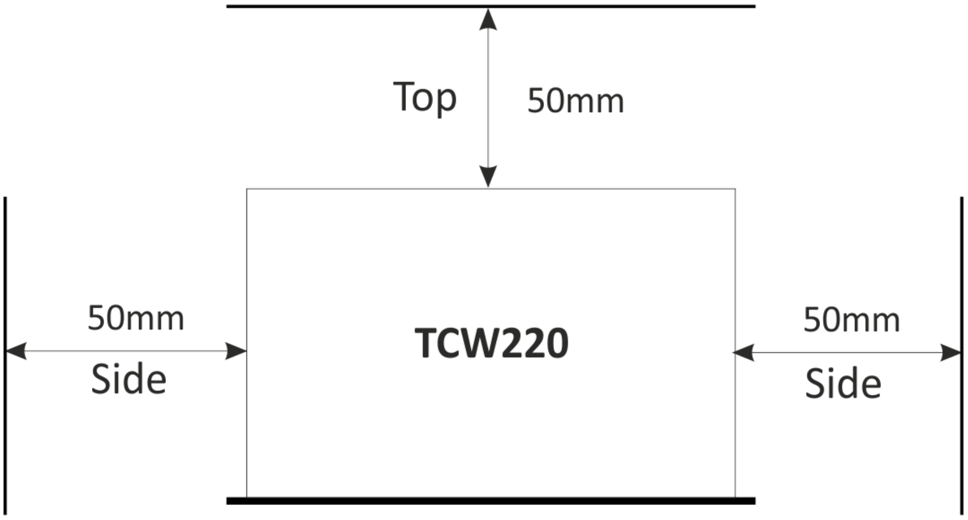


Fig.2