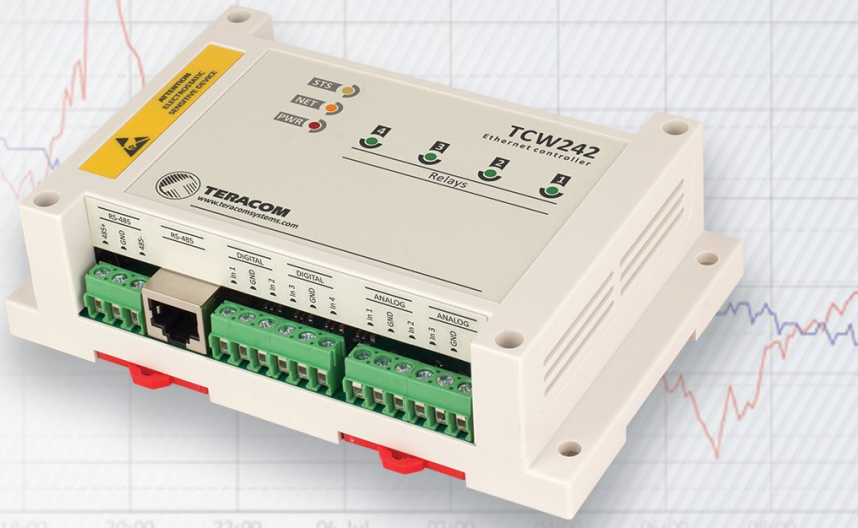
control solutions
TERACOM



TCW242 Industrial IoT module

Version 1.8 / January 2024

www.teracomsystems.com

1. Introduction

The TCW242 is an industrial-grade IoT (Internet of Things) module that offers advanced monitoring and control capabilities for industrial applications. Equipped with Ethernet connectivity and data logging functionality, this device allows users to easily monitor and analyze a wide range of parameters. The device features 4 digital inputs, 4 analog inputs, an RS-485 interface for connecting MODBUS RTU sensors, and 4 relays. The analog inputs can be configured to work in either current loop (0-20mA) or voltage (0-10V) modes, providing flexibility in monitoring different types of parameters.

The TCW242 supports up to 24 Teracom and third-party MODBUS RTU sensors, allowing users to configure up to 24 independent monitoring channels. Each channel can be set up using up to 2 input parameters and/or constants and can be configured as either discrete or general channels. This provides users with a wide range of monitoring options.

In addition to monitoring capabilities, the device also offers robust alarm functionality. Users can configure up to 24 independent alarms with 5 different user-selectable states. Each alarm can be set up using up to 2 limits and hysteresis and can be assigned to a specific channel. In this case, in an alarm state, the assigned channel is colored on the monitoring page and graphs, providing users with a visual indication of the alarm state.

The device also supports a variety of communication protocols for M2M (machine-to-machine) communication, including SNMP, HTTP API, MODBUS TCP/IP, and MQTT. These protocols allow for easy integration with other industrial systems and devices, providing users with a wide range of options for data analysis and control. Overall, the TCW242 industrial IoT module is a powerful and versatile device that offers advanced monitoring and control capabilities for industrial applications.

2. Features

- Ethernet connectivity at 10/100 Mbps with automatic cable detection;
- HTTP/HTTPS server support
- Web-based configuration and control with password protection;
- 4 digital inputs for dry contacts;
- 4 analog inputs with 0-10V and 0-20mA options;
- 4 relays with normally open/normally closed contacts;
- MODBUS RTU for up to 24 sensors (registers);
- Monitor up to 24 channels;
- Set up to 24 independent alarms;
- SNMPv3 protocol support;
- Alarm notifications via SNMP traps;
- HTTP and SNMP port changing;
- HTTP API commands;
- Periodical HTTP/HTTPS Post of XML/JSON status files for client-server systems;
- MODBUS TCP/IP protocol support;
- MQTT 3.1.1 protocol support;
- Dynamic DNS with DynDNS, No-IP, and DNS-O-Matic support;
- NTP protocol support;
- Data logging capacity for up to 70,000 records;
- Schedule single or repeating tasks;
- Custom functions available;
- ICMP Ping Watchdogs;
- Mounts on DIN rail;

- Option to backup and restore settings;
- Remote firmware update capability.

3. Applications

The industrial IoT module is a versatile solution for a variety of applications in the light industry. Its robust design and advanced features make it ideal for remote monitoring and control in communication facilities, food and beverage storage, greenhouses, water stations, and other industrial settings. With its advanced communication options, the module can easily be integrated into existing systems, and its flexible design allows it to be configured to meet the specific needs of each application. Whether you need to monitor temperature and humidity in a greenhouse or manage water levels at a water station, this industrial IoT module is a reliable and cost-effective solution.

4. Specifications

- Physical characteristics
  Dimensions: 145 x 90 x 40 mm
  Weight: 200 g

- Environmental limits
  Operating temperature range: -20 to 55°C
  Storage temperature range: -25 to 60°C
  Operating relative humidity range: 10 to 80% (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: 0.35A @ 12VDC (without MODBUS RTU sensors powering)

- RS-485 interface for MODNUS RTU sensors
  Isolation: Non-isolated
  Output voltage (pin 7 of RJ-45): 5.0 ± 0.3 VDC
  Maximum output current (pin 7 of RJ-45): 0.2 A

- Digital inputs
  Isolation: Non-isolated
  Mode: OPEN/CLOSED (“Dry contact”) or COUNTER
  Maximum input voltage: +5.5VDC
  Sampling rate: 1mS
  Digital filtering time interval: 5 to 60000mS

- Analog inputs
  Isolation: Non-isolated
  Type: Single-ended
  Resolution: 12 bits
  Mode: Voltage or current loop
  Input Range: 0/10V or 0/20mA
  Accuracy: ±1%
  Input Impedance: 1 mega-ohm (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: 0.1 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.

- Internal FLASH memory
  Settings segment endurance: 100 000 cycles (Every setting change is a memory cycle).
  Data logger segment endurance: 100 000 full scrolls (70000 records) of the logger.
  Update segment endurance: 100 000 cycles (updates).

- Lithium battery
  Type: CR1220

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

5. LED indicators

The device has 3 LED indicators that show its status:

- **PWR** (red) – lights up during normal operation, and blinks together with STS in case of a hardware error;
- **STS** (yellow) – blinks while the controller's main program is running;
- **NET** (orange) – indicates network status. It lights up when a connection is established and blinks with network activity.

6. Installation and configuration

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

The TCW242 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 D for mechanical details.

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

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

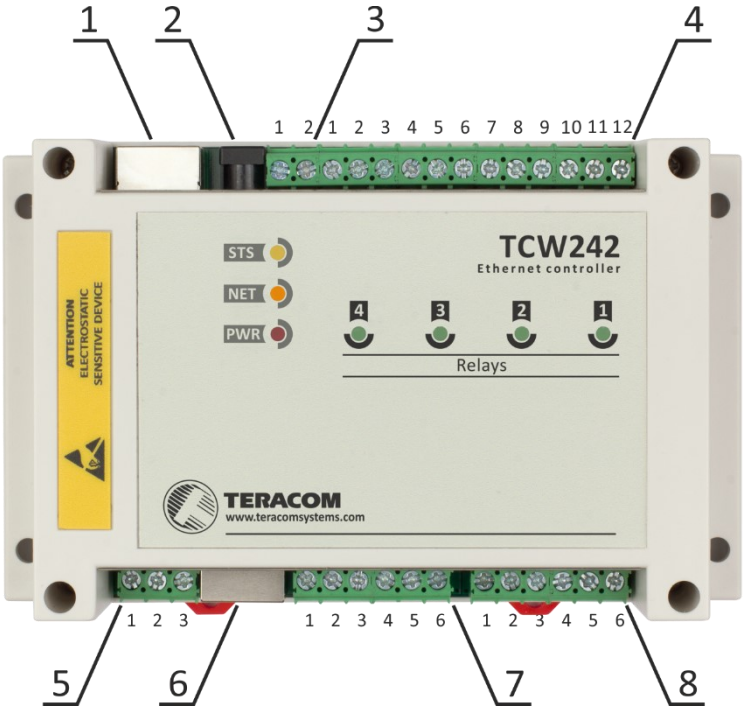
Warning! Power off before wiring.

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 TCW242 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 Relay4 **Pin2** – COM Relay4 **Pin3** – NO Relay4 **Pin4** – NC Relay3 **Pin5** – COM Relay3 **Pin6** – NO Relay3 **Pin7** – NC Relay2 **Pin8** – COM Relay2 **Pin9** – NO Relay2 **Pin10** – NC Relay1 **Pin11** – COM Relay1 **Pin12** – NO Relay1
Connector 5	**Pin1** – RS485+ **Pin2** – GND **Pin3** – RS485-
Connector 6	**Pin1** – not connected (most left) **Pin2** – not connected **Pin3** – not connected **Pin4** – RS485- **Pin5** – RS485+ **Pin6** – not connected **Pin7** – +VDD **Pin8** – GND
Connector 7	**Pin1** – Digital In 1 **Pin2** – GND **Pin3** – Digital In 2 **Pin4** – Digital In 3 **Pin5** – GND **Pin6** – Digital In 4
Connector 8	**Pin1**– Analog In 1 **Pin2** – GND **Pin3** – Analog In 2 **Pin4** – Analog In 3 **Pin5** – GND **Pin6** – Analog In 4

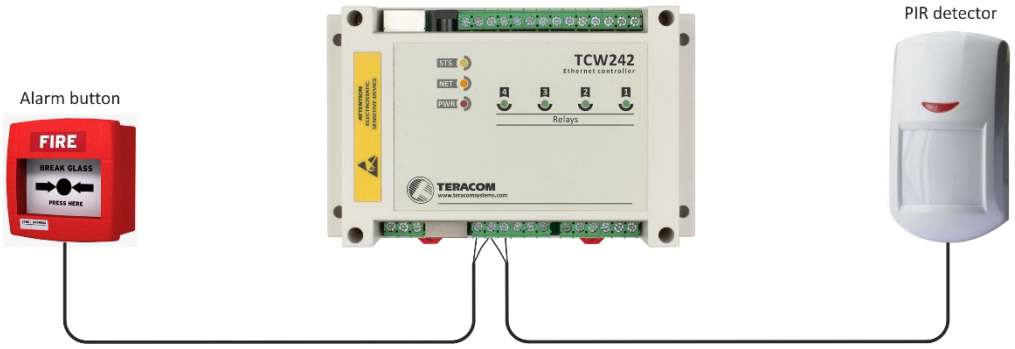
6.2.1. Power supply connection

TCW242 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 connection

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

The digital inputs of TCW242 in OPEN/CLOSED mode can monitor devices with "dry contact" outputs such as door contacts, push buttons, PIR detectors, etc. To connect an alarm button and PIR detector to TCW242, connect one side to the "Digital In" terminal and the other side to the "GND" terminal, as illustrated in a picture:

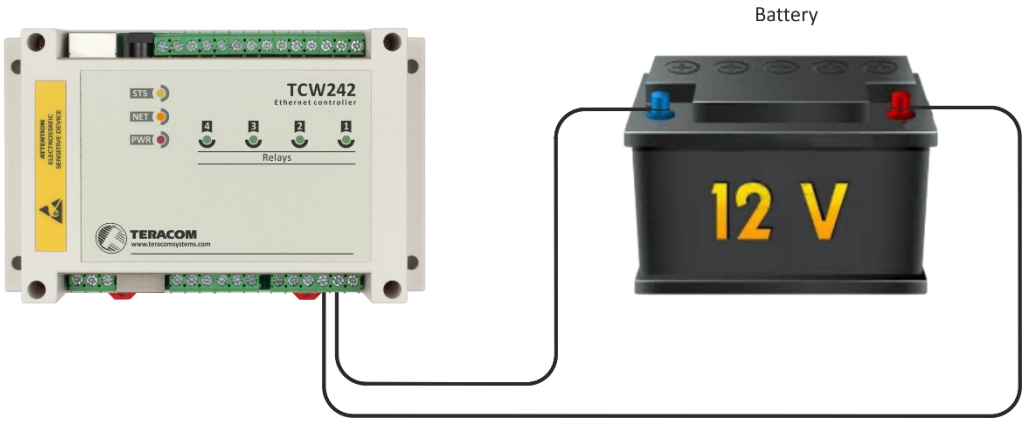


For COUNTER mode, the inputs can monitor devices with a "dry contact" pulse output interface. The maximum cable length should not exceed 30 meters.

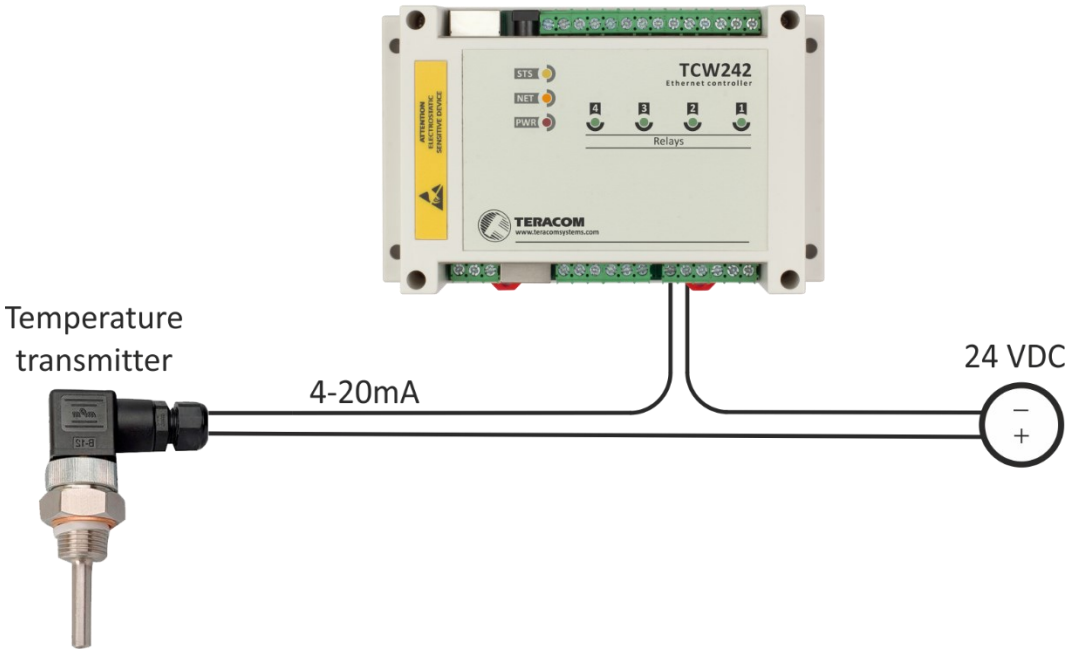
6.2.3. Analog inputs connection

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

The analog inputs of TCW242 can be used to monitor devices with voltage and current loop outputs and connect directly to sensors, such as temperature and humidity sensors, voltage and current transducers, etc. The illustration shows how to connect a battery to the analog input of TCW242 in voltage mode, with the positive terminal connected to "Analog In" and the negative terminal to "GND."



The illustration shows how to connect a temperature sensor with a current loop output to the analog input. Connect the active terminal to "Analog In" and the shield terminal to "GND."



The maximum cable length should not exceed 30 meters.

6.2.4. RS-485 connection

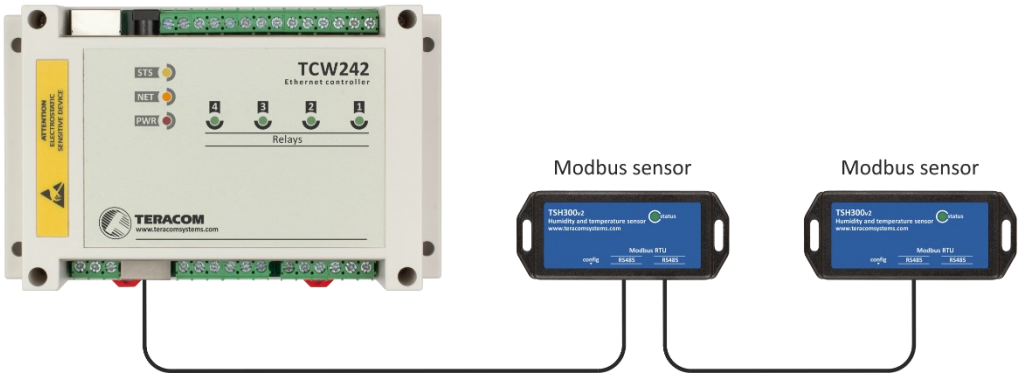
The RS-485 interface is not isolated from the power source.

Up to 24 MODBUS RTU sensors can be connected to TCW242, including both Teracom and third-party sensors, using a standard RJ-45 connector with the recommended pinout found in the "MODBUS over Serial Line Specification and Implementation Guide" on www.modbus.org.

A 120-ohm line terminator must be used at both ends of the bus, with TCW242 incorporating one terminator and the client responsible for the other.

It is recommended to use a "daisy-chained" (linear topology) for multiple sensors.

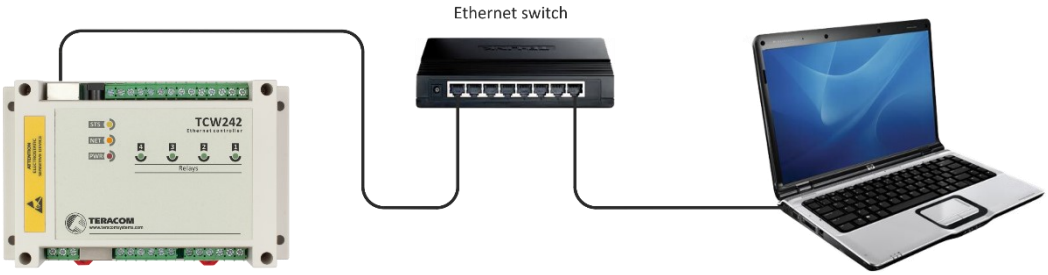
RS-485 interface is non-isolated from the power supply.



It is advised to utilize only UTP/FTP cables and limit the overall cable length to 30 m, although longer distances have been proven to work as well.

6.2.5. Ethernet connectivity

TCW242's Ethernet port should be connected to a 10/100 Base-T Ethernet hub, switch, or router.



Direct connection to a computer's Ethernet port is possible for configuration purposes, and TCW242 supports Auto-MDIX, allowing you to use either a standard straight-through or a crossover cable.

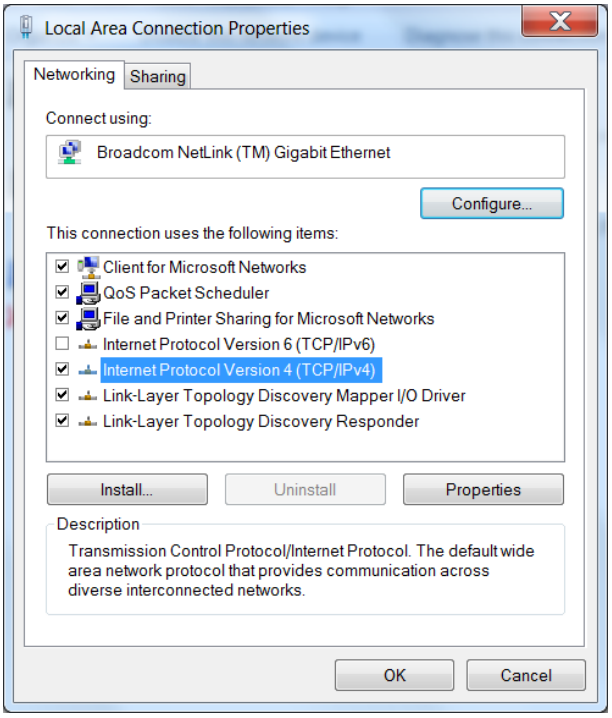
TCW242 can be integrated into a wireless network by connecting to a wireless router.

6.3. Communication setup

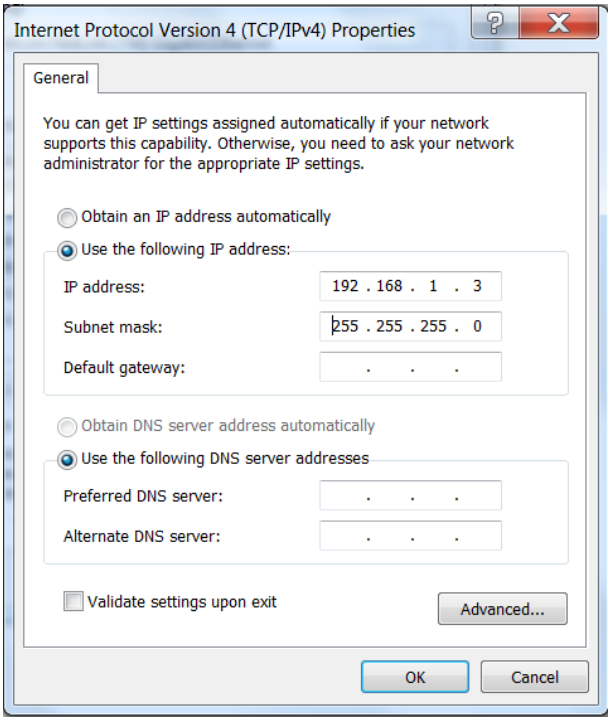
By default, TCW242 is delivered with these network settings:

IP address: ***192.168.1.2****, Subnet Mask:* ***255.255.255.0****, Default Gateway:* ***192.168.1.1***

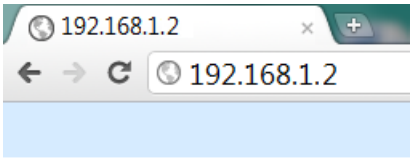
To communicate with TCW242, assign a temporary IP address to your computer. For Windows, it can be done in "Local Area Connection Properties":



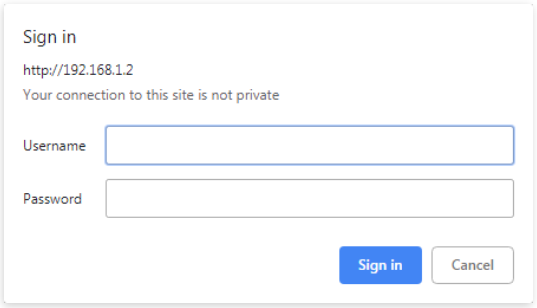
The assigned address should be on the same network, e.g. 192.168.1.3:



To access the web interface, type http://192.168.1.2 into your browser.



If the network settings are correct, the login pop-up window will appear:



Authorization is required, the default login is admin/admin.

Change username/password for security.

You can easily locate all TCW controllers connected to the LAN using the free tool "TCW discoverer." It is compatible with Windows and Mac operating systems and can be downloaded from www.teracomsystems.com.

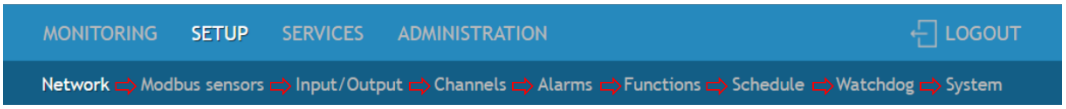
7. Concept of setup

The primary setup method of the device is via its web interface, although additional settings can be made through SNMP and HTTP API commands.

First, network settings are configured, then primary parameters such as MODBUS RTU sensors, analog inputs, and digital inputs are set up. Next, channels are created from the primary parameters, allowing up to two parameters and constants to be combined with mathematical operations, or consisting of just one primary parameter.

The alarm setup comes after tuning the channels. It's crucial to note that alarms operate with channels, not primary parameters. An alarm can have a maximum of two conditions and can involve multiple channels. Alarms are standalone but can be assigned to a channel.

Hence, the proper setup order is as follows:



Finally, desired services such as data logging, SNMP, HTTP API, etc. can be enabled.

8. Web interface

The web interface facilitates configuration and monitoring tasks, with all pages encoded in UTF-8. The controller accommodates multiple active sessions, and for the web interface, the device is compatible with both HTTP and HTTPS.

8.1. Monitoring

In the monitoring section, the status of all channels, relays, and alarms are displayed both textually and graphically. The "Data" and "Alarms" pages refresh automatically at an interval of 0 to 253 seconds, with zero meaning no refresh. This interval can be set in the "Setup-System-Refresh of channels and alarms pages" section, with a default of 1 second.

8.1.1. Channels and relays

This page shows the current values and alarm status of all monitored channels and relays. The information is updated at the refresh interval.

Channels

Channel	Description	Value	Unit	Status
1	V01-Analog	3.054	V	Normal
2	V02-Temperature	30.424	°C	Warning
3	V03-Humidity	52.147	%RH	Minor
4	V04-D1	20011		Critical

Relays

Relay	Description	Status	Control
1	Relay 1	ON	ON OFF Pulse
2	Relay 2	OFF	ON OFF Pulse
3	Relay 3	OFF	ON OFF Pulse
4	Relay 4	OFF	ON OFF Pulse
			All On All Off Pulse All

8.1.2. Alarms

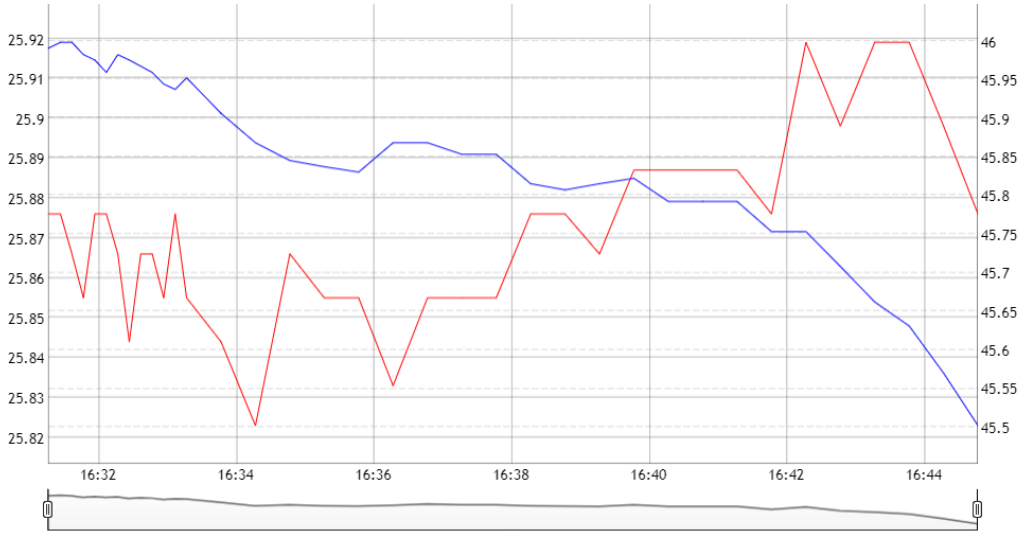
The page shows the status of all alarms. The information is updated at the refresh interval.

Alarms	Description	Status
1	AL01-AN	Normal
2	AL02-T	Warning
3	AL03-H	Minor
4	AL04-D1	Critical

8.1.3. Graphs

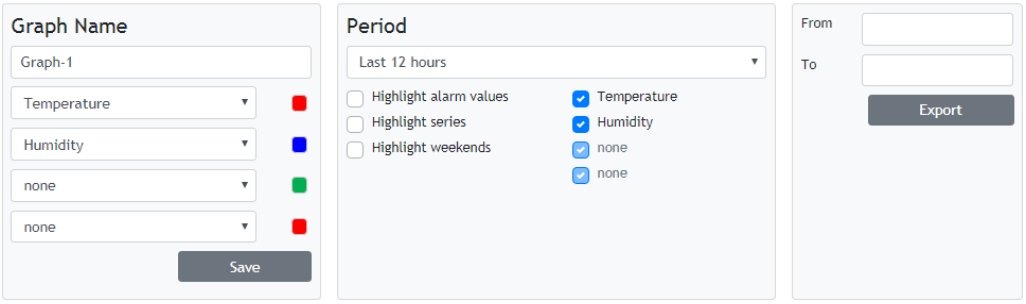
Channels and their alarm statuses can be visually monitored through tabs Graph-1 to Graph-6.

Each graph supports up to 4 channels with 2 different dimensions, and the curve color for each channel can be selected. Alarm statuses are shown with fixed colors. Modify the display with checkboxes.



To view past information, the data logger must be active.

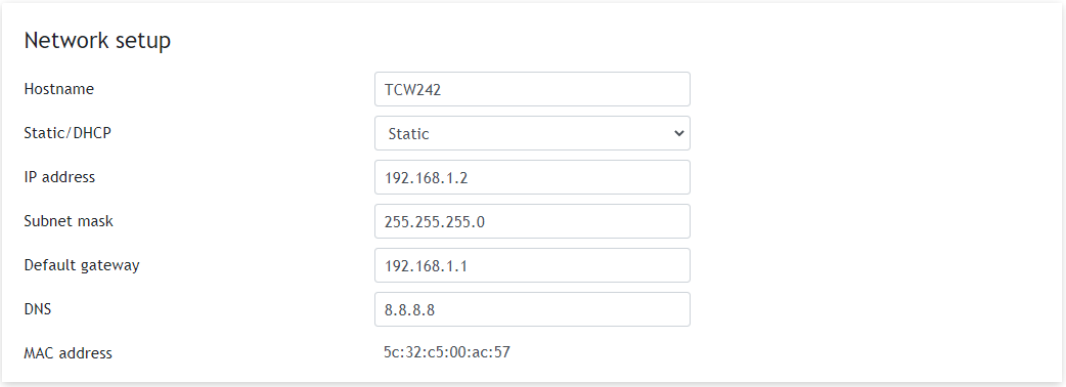
Note that graph information is static and not updated in real time. To view the latest information, reload the page. Exporting information as a CSV file is possible.



8.2. Setup

8.2.1. Network

The network parameters can be configured on this page, including static and dynamic IP address assignments.



It's recommended to change the default IP of the controller after the initial power-up to prevent collisions on the same network. Clearing the ARP cache by typing "arp -d" in the computer's command prompt may be necessary after connecting a new device.

The hostname, with a maximum length of 15 characters, is displayed in TCW discoverer search results. Using a public DNS server such as 8.8.8.8 or 8.8.4.4 is recommended over the default gateway.

8.2.2. Modbus sensors

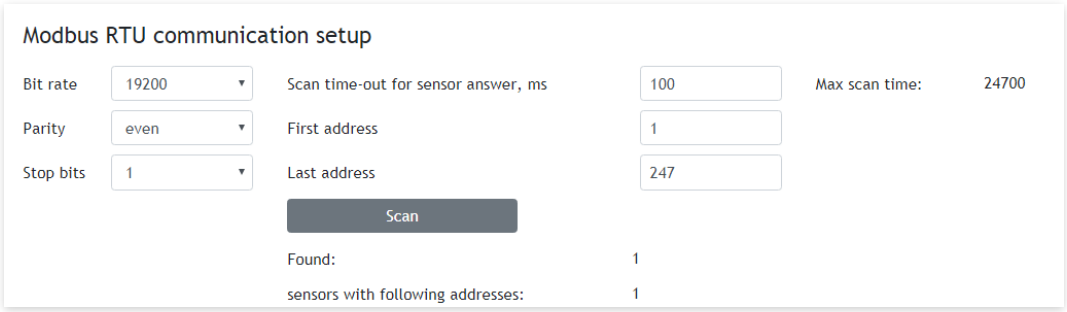
8.2.2.1. Modbus RTU communication configuration

This page enables you to configure the RS-485 interface communication parameters such as bit rate, parity, and stop bits. The default settings are 19200, even parity, and 1 stop bit.

All sensors on the bus must have the same bit rate, parity, and stop-bit settings. Before adding any sensors, ensure their parameters are properly configured.

A bus scanning tool is available to report the number of detected sensors and their addresses. This tool is useful when adding new sensors and it is recommended to use a small address range to speed up the scanning process.

You can adjust the "Scan Timeout for Sensor Answer" to find the optimal setting for an unknown sensor. The test begins with a large timeout (e.g. 500ms) and gradually reduces the time until the sensor stops responding. To ensure stability, increase the found timeout by, for example, 20%.

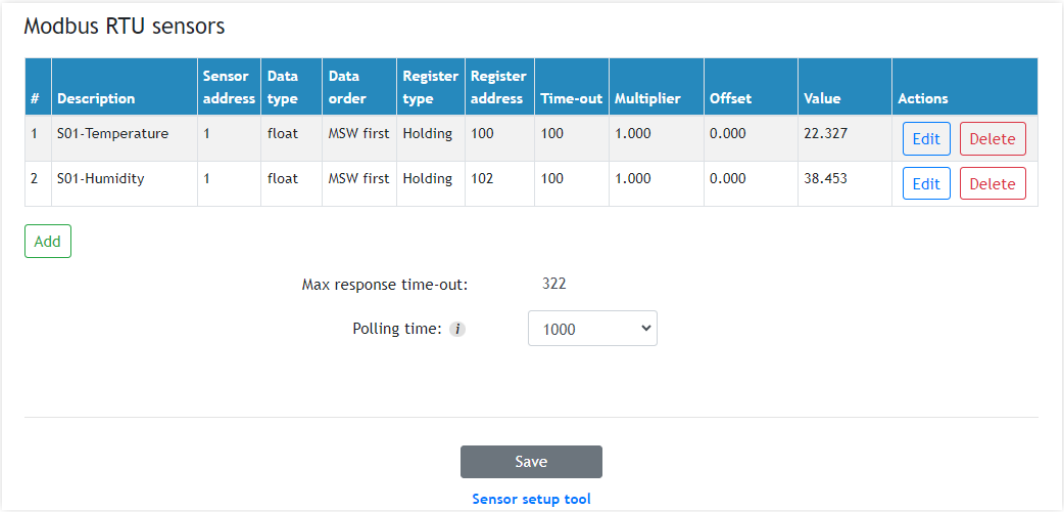


8.2.2.2. Modbus RTU sensors

This section allows for the management of MODBUS RTU sensors/registers, including adding, deleting, or editing them. These serve as the primary parameters that can be utilized in creating channels.

It is advisable to add sensors/registers individually using the scan tool outlined in 8.2.2.1.

A maximum of 24 sensors/registers can be added and displayed in a table.



According to the MODBUS convention, the address range for slaves is 1 to 247.

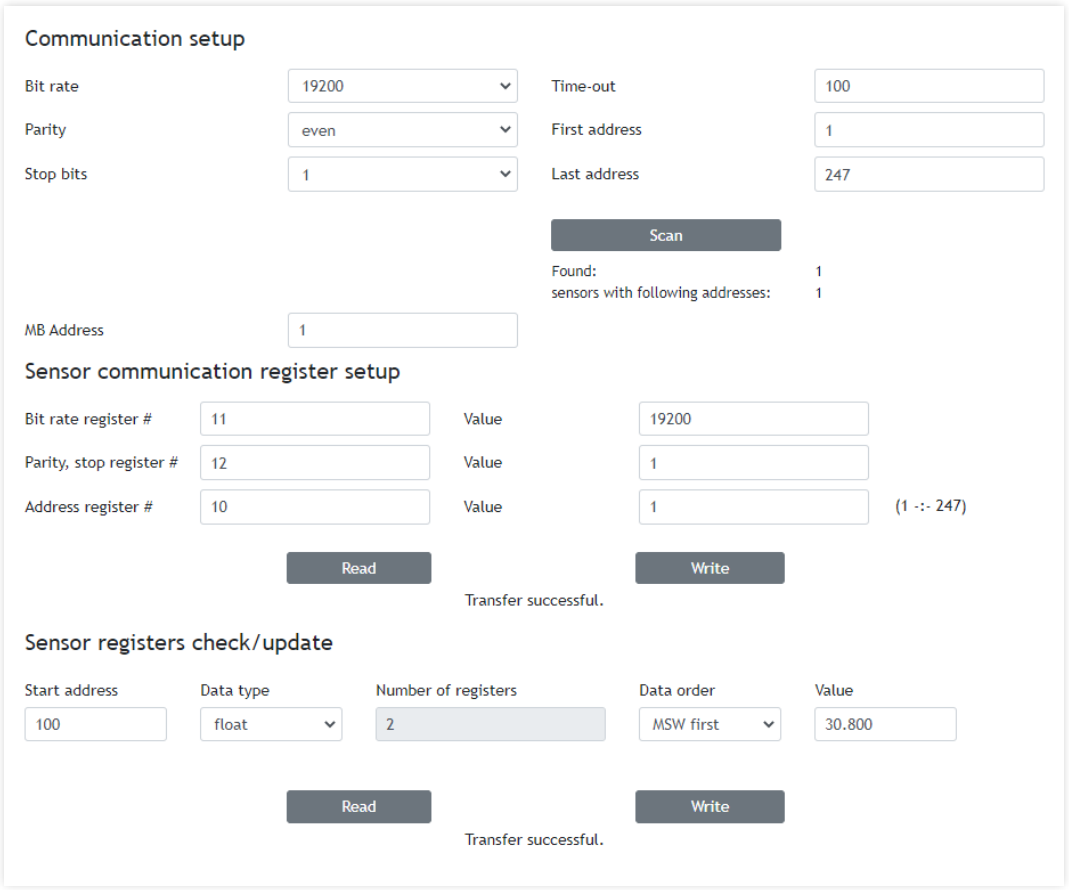
The value of the sensor can be calculated using the formula: Value = (Raw_Value * Multiplier) + Offset. To view the raw value, set Multiplier to 1 and Offset to 0.

The controller continuously polls all sensors and expects a response within a specified "response time-out". If the same sensor fails to respond three times, it is considered not present. The "maximum response time-out" for the system is determined by the sum of response time-outs for all sensors, and the system's response is based on this value. It is strongly recommended that "Polling time" be greater than or equal to "Maximum response time-out".

The "Polling time" can be set to 1, 2, 3, 4, 5, 10, 15, 30, 60, 120, or 180 seconds and is 1 second by default.

8.2.2.3. Sensor setup tool

The sensor setup tool allows you to configure and control MODBUS RTU sensors from different manufacturers by changing their addresses and communication parameters. It is a simple yet useful tool.



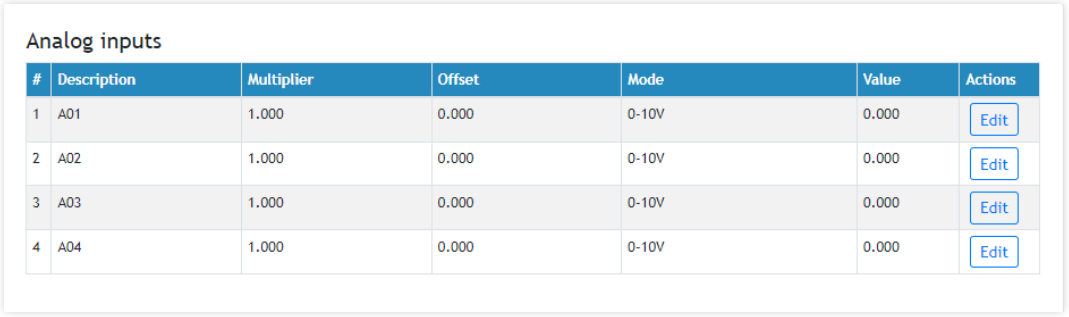
8.2.3. Input/Output

This page allows for the configuration of relays, analog, and digital inputs.

8.2.3.1. Analog inputs

The TCW242 has 4 non-isolated analog inputs. Each input can operate as either voltage (0-10V) or current loop (0-20mA).

The default settings after a "Factory default" procedure are Multiplier=1.00, Offset=0.00, and Mode=0-10V.



For every analog input, fields “Multiplier” and “Offset” are available to convert the raw voltage/current into meaningful engineering units.

The scaled value is calculated by:

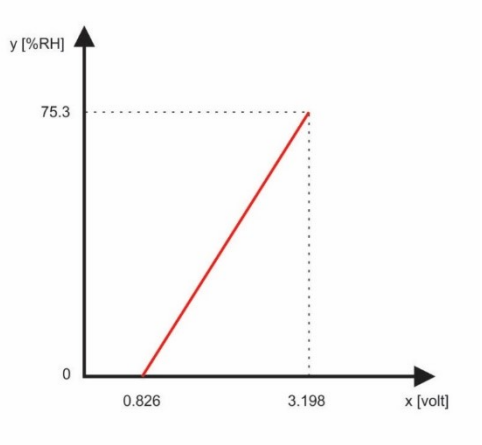
SV = RV * MU + OF
Where:
SV – scaled (displayed) value;
RV – raw voltage from the source;
MU – multiplier;
OF – offset.

Example:

For humidity sensor HIH-4000-003 following data (from the datasheet) is available:

VOUT = 0.826 at 0% RH

VOUT = 3.198 at 75.3% RH



The sensor provides raw voltage values as output, but what we actually need is the corresponding relative humidity values. To achieve this, we use a multiplier and an offset. These two parameters allow us to calculate the relative humidity for any given voltage within the sensor's working range.

The multiplier (MU) is determined by the ratio of the change in relative humidity (ΔRH%) to the change in voltage (ΔV). In geometric terms, this is akin to finding the slope of a line. For this particular sensor, the line is described 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

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

OF = SV - (RV * MU)

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

OF = 0 - (0.826 * 31.745) = 0 - 26.22 = -26.22

Similarly, we can calculate the offset using the other point, where SV = 75.3 and RV = 3.198:

OF = 75.3 - (3.198 * 31.745) = 75.3 - 101.52 = -26.22

Therefore, the formula for this sensor becomes:

SV = RV * 31.745 - 26.22

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

SV = 0.826 * 31.745 - 26.22 = 26.22 - 26.22 = 0 %RH

This confirms that the formula correctly converts the voltage to the corresponding relative humidity value.

For example, if the sensor's output voltage is 3.198V, the analog input value will be 75.28% RH:

75.28 = (3.198 - 0.826) * 31.74

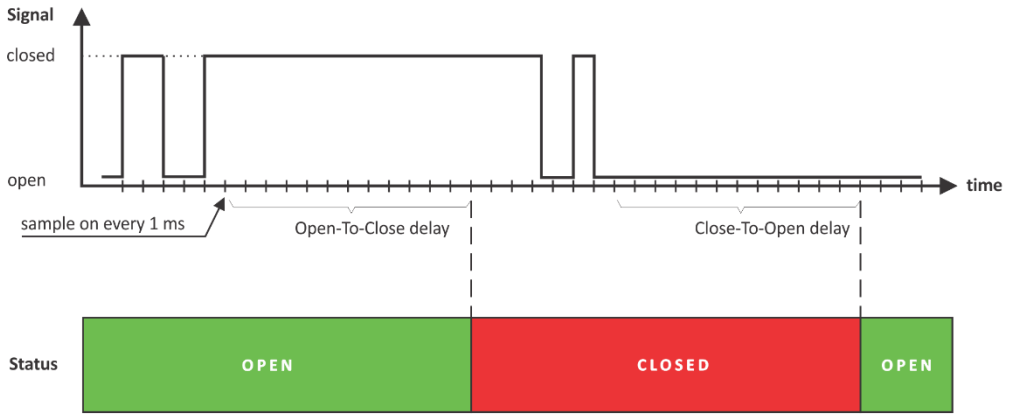
8.2.3.2. Digital inputs

The TCW242 features 4 non-isolated digital inputs. Each input can operate as either OPEN/CLOSE or COUNTER mode. In the COUNTER mode, counting can occur on rising, falling, or both edges, with the ability to set an initial counter value.

Digital inputs

#	Description	Closed state	Open state	C/O delay	O/C delay	Mode	Value	Actions
1	D1	CLOSED	OPEN	5	5	Discrete(Open/Closed)	OPEN	Edit
2	D2	CLOSED	OPEN	5	5	Counter(Rising edge)	10158	Edit
3	D3	CLOSED	OPEN	5	5	Counter(Falling edge)	11	Edit
4	D4	CLOSED	OPEN	5	5	Counter(Both edges)	16	Edit

There are two delays, Open-to-Close and Close-to-Open, that can be set between 5ms and 60000ms for additional digital filtering and are applicable in both modes.



In the illustration, the Open-to-Close and Close-to-Open delays are set to 13ms.

8.2.3.3. Relay outputs

The TCW242 has 4 relay outputs which can be controlled through the WEB interface, HTTP API, SNMP, Function, Schedulle, Watchdog or locally using alarm conditions. The WEB control for each relay offers "On", "Off", and "Pulse" buttons, as well as "All On", "All Off", and "Pulse All" for collective control. The pulse duration can be set individually for each relay in the "Setup-Input/Output-Relay Outputs" section.

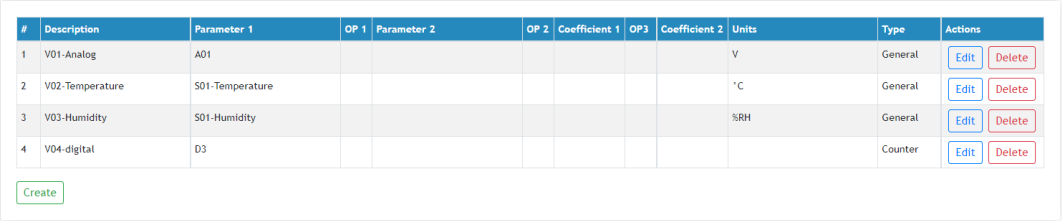
Relay outputs

#	Description	Pulse	Activated from	Action on alarm condition	Status	Actions
1	Relay 1	0.3	Any Alarm	Single pulse	OFF	Edit
2	Relay 2	1.0	Watchdog1	Turn on	ON	Edit
3	Relay 3	1.0	Watchdog3	Single pulse	OFF	Edit
4	Relay 4	1.0	Watchdog4	Single pulse	OFF	Edit

For local activation, a description of the controlling parameter is displayed instead of buttons, and the parameters for local activation can be set in the "Setup-Input/Output-Relay Outputs" section.

8.2.4. Channels

The section allows for adding, editing, and deleting channels, with a max of 24, for monitoring and periodic data logging.



#	Description	Parameter 1	OP 1	Parameter 2	OP 2	Coefficient 1	OP3	Coefficient 2	Units	Type	Actions
1	V01-Analog	A01							V	General	Edit Delete
2	V02-Temperature	S01-Temperature							°C	General	Edit Delete
3	V03-Humidity	S01-Humidity							%RH	General	Edit Delete
4	V04-digital	D3								Counter	Edit Delete

Create

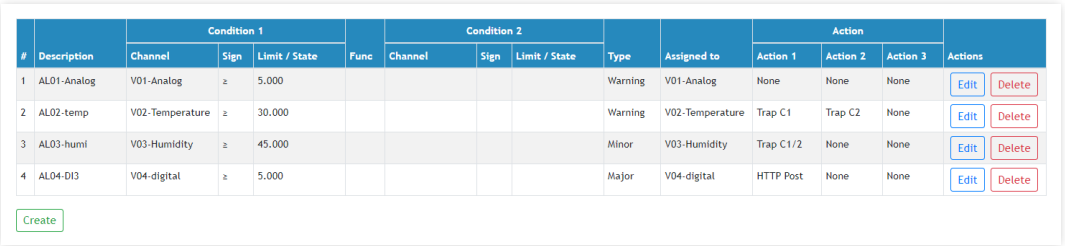
Discrete channels consist of a single digital input in OPEN/CLOSE mode, while general channels are made up of up to two primary parameters and constants.

Note that for general channels, operations are performed in the sequence of OP1, OP2, and OP3. A digital input in OPEN/CLOSE mode can also be used to form a general channel with values of 0 for CLOSE and 1 for OPEN.

8.2.5. Alarms

The Alarms section allows you to manage up to 24 alarms, including adding, deleting, or editing. Only channels can be used to create alarms, with 4 types available: Warning, Minor, Major, and Critical.

The alarms can be assigned to any channel and can have up to 2 conditions, joined by logical operators AND and OR, with no restrictions on which channel to use in each condition. It's possible to combine conditions from different channels in one alarm



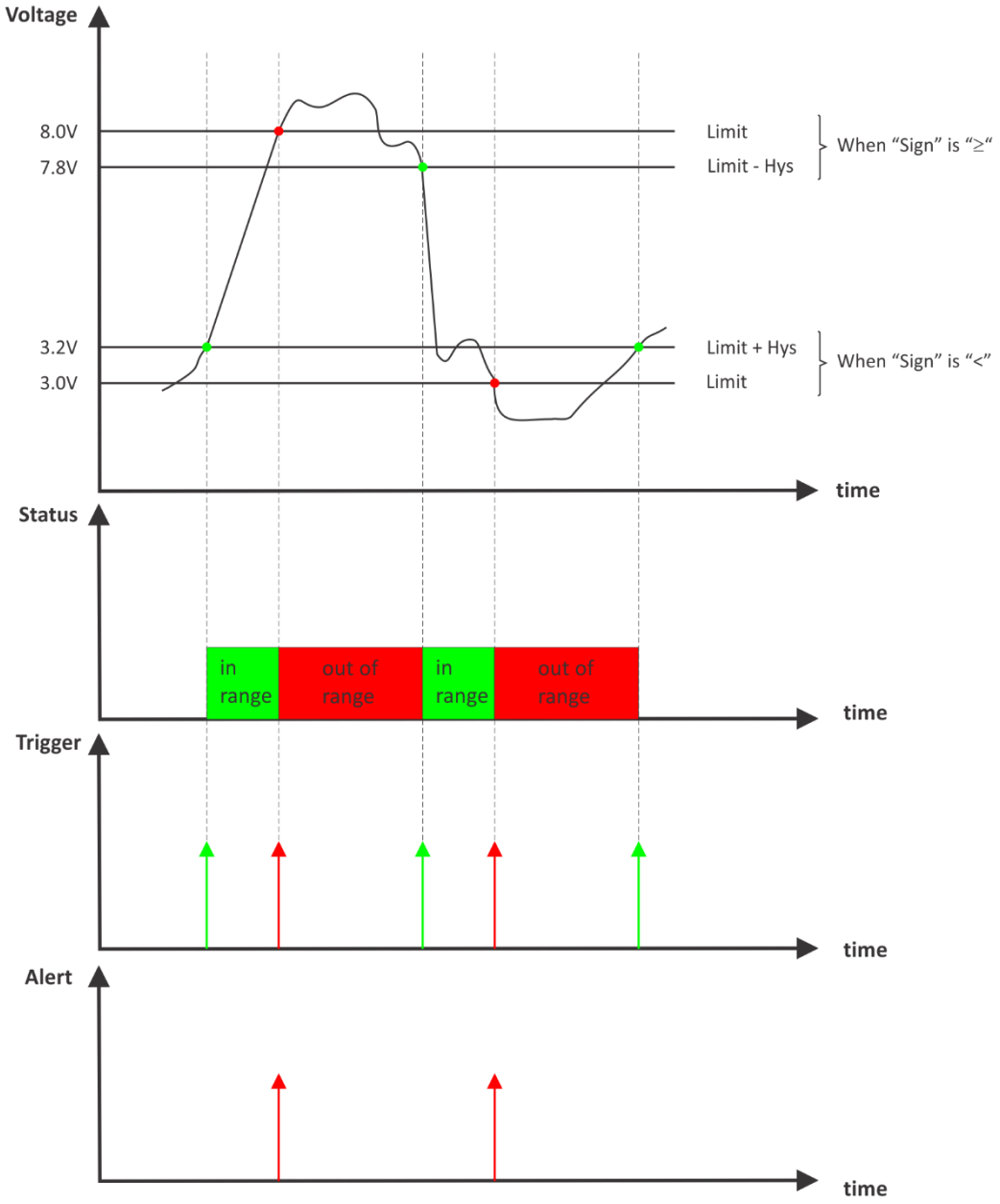
#	Description	Condition 1			Func	Condition 2			Type	Assigned to	Action			Actions
		Channel	Sign	Limit / State		Channel	Sign	Limit / State			Action 1	Action 2	Action 3	
1	AL01-Analog	V01-Analog	≥	5.000					Warning	V01-Analog	None	None	None	Edit Delete
2	AL02-temp	V02-Temperature	≥	30.000					Warning	V02-Temperature	Trap C1	Trap C2	None	Edit Delete
3	AL03-humi	V03-Humidity	≥	45.000					Minor	V03-Humidity	Trap C1/2	None	None	Edit Delete
4	AL04-DI3	V04-digital	≥	5.000					Major	V04-digital	HTTP Post	None	None	Edit Delete

Create

The "Limit" sets the boundary for the normal operating range of the monitored channel. If the channel value exceeds the limit (exceeds it by going higher with "Sign" set to "≥" or goes lower with "Sign" set to "<"), a trigger event occurs.

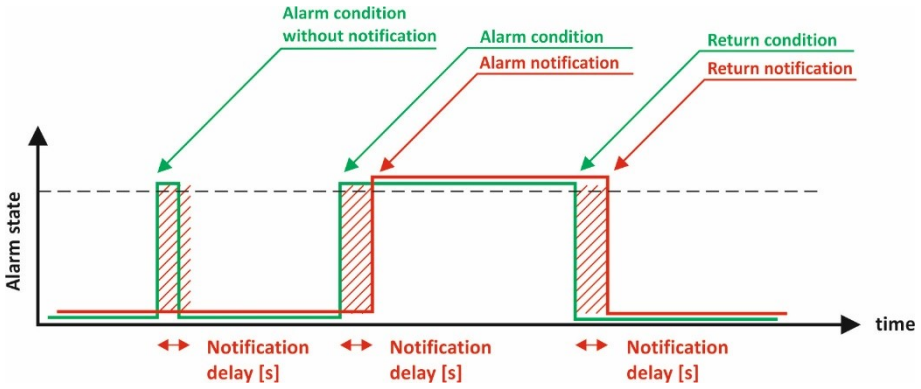
The trigger event is considered resolved when the value returns within the range, either by going higher than the Limit + Hysteresis with "Sign" set to "<" or lower than Limit - Hysteresis with "Sign" set to "≥".

Hysteresis helps prevent frequent triggering due to small fluctuations around the trigger point and it is highly recommended to not use a value of 0.0 for Hysteresis.



Every alarm has a "Return notification" option, enabling a notification when the parameter returns to the normal range.

Additionally, each alarm has a "Notification delay" parameter, serving as a filter for brief alarm occurrences.



8.2.6. Functions

This section allows you to set up four separate functions, each of which can include up to four alarm conditions connected by AND and OR operators. The functions can be selected from a drop-down menu to activate local relays. Each function has parameters for "Action delay" and "Action on return".

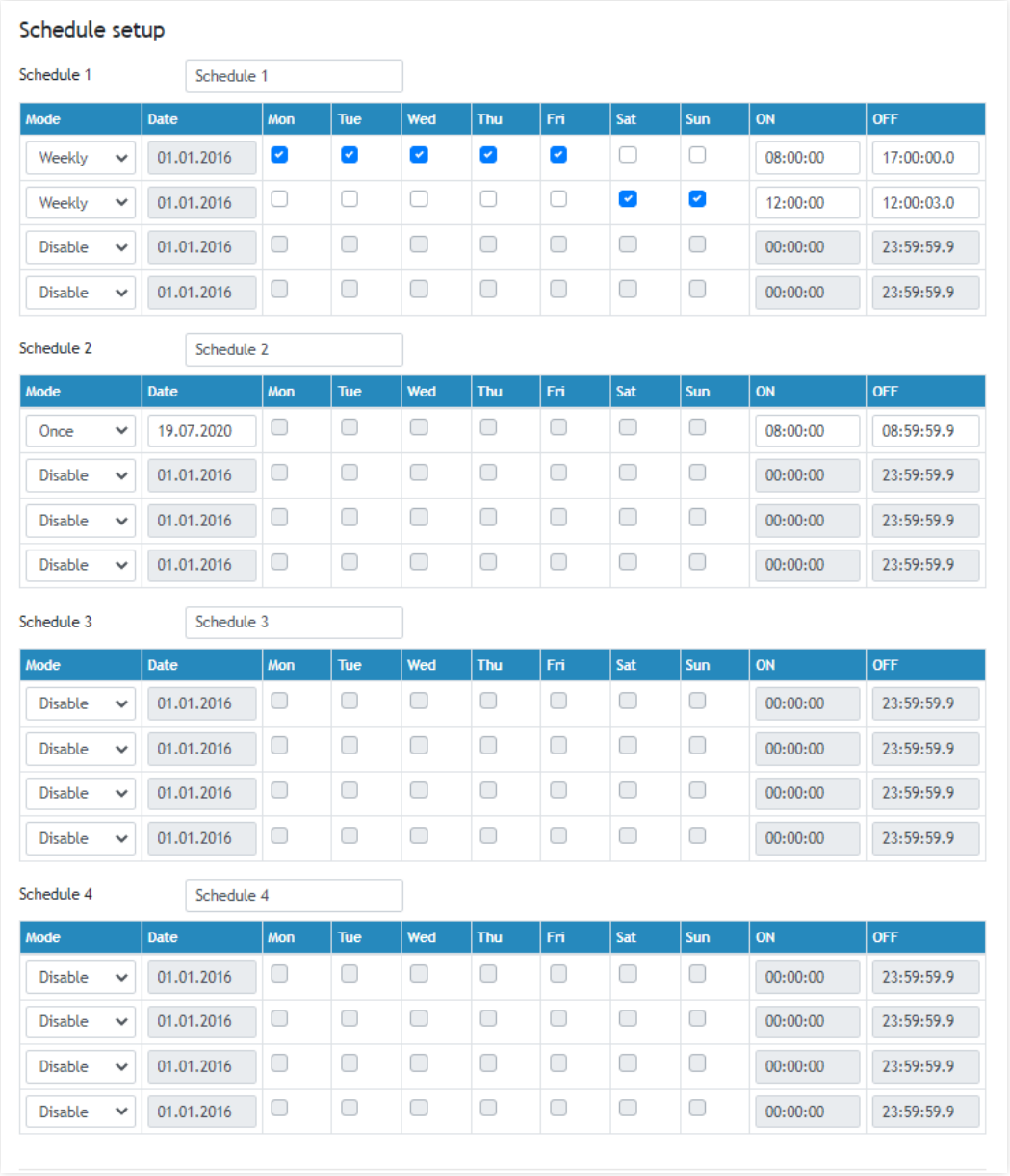
#	Condition 1	Func	Condition 2	Func	Condition 3	Func	Condition 4	Action		Actions
								Action 1	Action 2	
1	AL02-D1	AND	AL03-D2					SNMP Trap	None	Edit Delete
2	AL01-TST300-T	OR	AL02-D1					HTTP Post	None	Edit Delete

Create

8.2.7. Schedule

TCW242 has 4 schedules, each with up to 4 tasks.

The schedules allow you to create tasks based on calendar dates and can be combined with alarms for more advanced control. Two relays can control a single device, with one following the alarm and the other following the schedule.



Schedule setup

Schedule 1 — Schedule 1

Mode	Date	Mon	Tue	Wed	Thu	Fri	Sat	Sun	ON	OFF
Weekly	01.01.2016	☑	☑	☑	☑	☑	☐	☐	08:00:00	17:00:00.0
Weekly	01.01.2016	☐	☐	☐	☐	☐	☑	☑	12:00:00	12:00:03.0
Disable	01.01.2016	☐	☐	☐	☐	☐	☐	☐	00:00:00	23:59:59.9
Disable	01.01.2016	☐	☐	☐	☐	☐	☐	☐	00:00:00	23:59:59.9

Schedule 2 — Schedule 2

Mode	Date	Mon	Tue	Wed	Thu	Fri	Sat	Sun	ON	OFF
Once	19.07.2020	☐	☐	☐	☐	☐	☐	☐	08:00:00	08:59:59.9
Disable	01.01.2016	☐	☐	☐	☐	☐	☐	☐	00:00:00	23:59:59.9
Disable	01.01.2016	☐	☐	☐	☐	☐	☐	☐	00:00:00	23:59:59.9
Disable	01.01.2016	☐	☐	☐	☐	☐	☐	☐	00:00:00	23:59:59.9

Schedule 3 — Schedule 3

Mode	Date	Mon	Tue	Wed	Thu	Fri	Sat	Sun	ON	OFF
Disable	01.01.2016	☐	☐	☐	☐	☐	☐	☐	00:00:00	23:59:59.9
Disable	01.01.2016	☐	☐	☐	☐	☐	☐	☐	00:00:00	23:59:59.9
Disable	01.01.2016	☐	☐	☐	☐	☐	☐	☐	00:00:00	23:59:59.9
Disable	01.01.2016	☐	☐	☐	☐	☐	☐	☐	00:00:00	23:59:59.9

Schedule 4 — Schedule 4

Mode	Date	Mon	Tue	Wed	Thu	Fri	Sat	Sun	ON	OFF
Disable	01.01.2016	☐	☐	☐	☐	☐	☐	☐	00:00:00	23:59:59.9
Disable	01.01.2016	☐	☐	☐	☐	☐	☐	☐	00:00:00	23:59:59.9
Disable	01.01.2016	☐	☐	☐	☐	☐	☐	☐	00:00:00	23:59:59.9
Disable	01.01.2016	☐	☐	☐	☐	☐	☐	☐	00:00:00	23:59:59.9

TCW242 has two types of schedules: single and weekly.

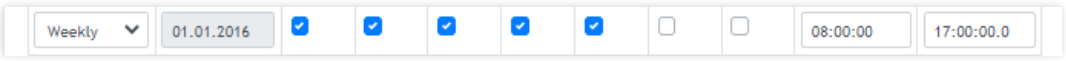
There are two types of schedules depending on repetition and duration – single and weekly tasks.

Here are some examples:

- A single task for a time period:

An example of a single task with a set time period could look like this: Event on 06.04.2020 from 08:00 to 09:00, with a resolution of 0.1 seconds for "OFF time." This allows for short pulse support.

- A weekly task for a time period:



With the above setting, an event will occur every weekday starting at 08:00 and ending at 17:00.

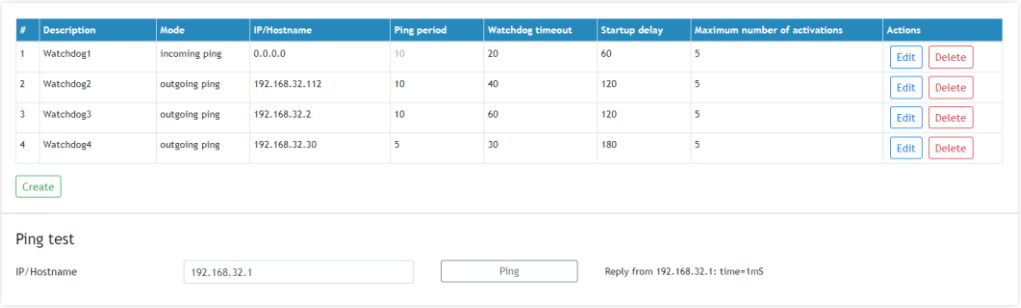
- A weekly task for a time period which includes midnight:

With the above configuration, there will be an event lasting 2 hours, occurring from Monday 23:00:00 to Tuesday 01:00:00.

8.2.8. Watchdog

The Watchdog feature enables the remote supervision of network devices using the ICMP protocol. In the absence of an ICMP echo reply (outgoing ping) or ICMP echo request (incoming ping), it activates a relay, which can be a reboot for certain devices.

TCW242 provides up to 4 independend Watchdogs.



Each Watchdog can be configured to operate in either ICMP echo reply (outgoing ping) or ICMP echo request (incoming ping) mode, or it can be disabled. Disabling the Watchdog will retain the previous settings.

The Ping reply timeout establishes the duration within which a ping reply must be received, and it is applicable solely for outgoing pings.

The Ping period sets the time gap between individual pings, exclusively applied for outgoing pings.

The Watchdog timeout dictates the duration within which an echo reply (outgoing ping) or echo request (incoming ping) must occur; beyond this period, the watchdog is triggered.

The startup delay specifies the waiting period after a reboot before the Watchdog reactivates.

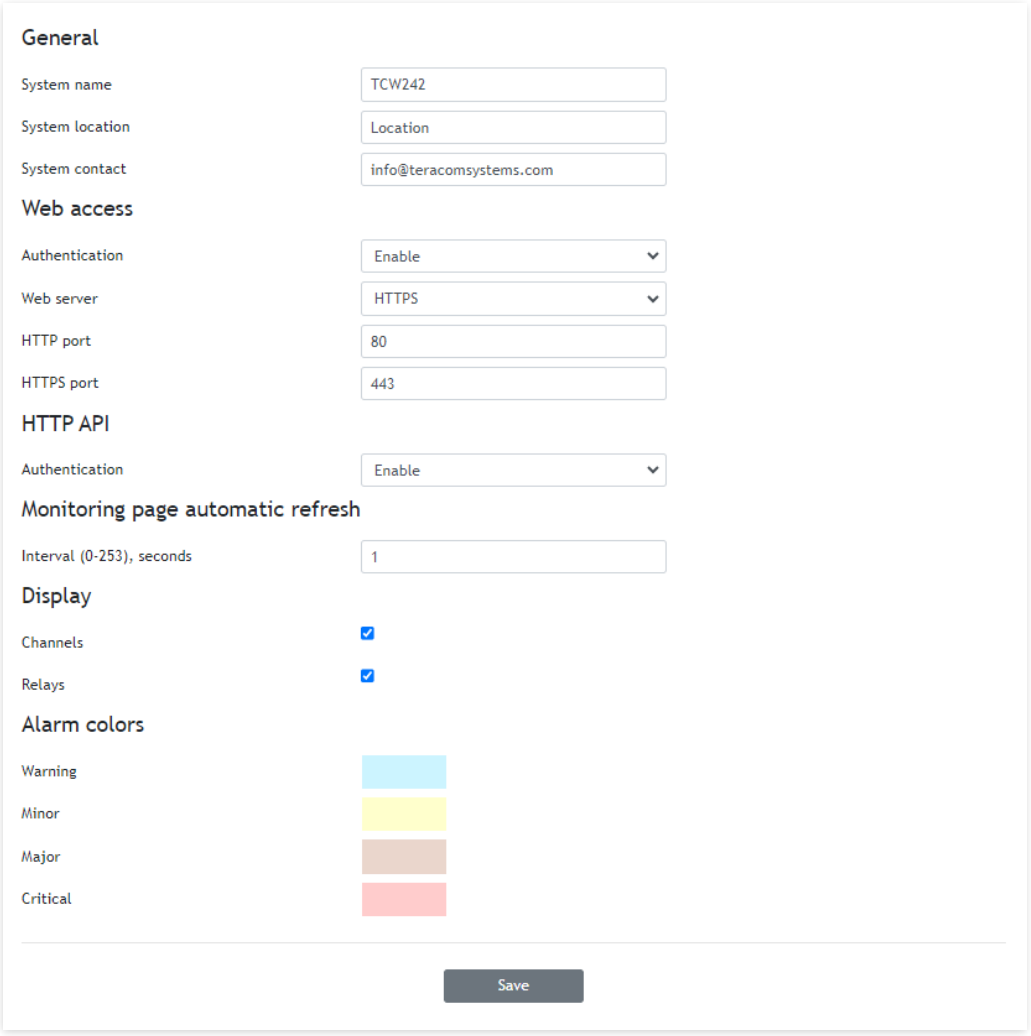
The Maximum number of activations determines the consecutive instances of relay activations during sequential Watchdog triggers.

To be able to activate a given relay, the corresponding Watchdog must be connected to this relay in the "Setup-Input/Output-Relay Outputs" section

8.2.9. System

On the "System" page, you can configure general system settings.

This includes the system name, location, and contact for device identification, which appear in SNMP and XML/JSON status files.



The default WEB access requires authentication with the username and password "admin/admin". The web server typically operates on the HTTP protocol through port 80, a customizable setting. This adaptability proves especially beneficial for routers lacking support for distinct outside/inside ports in port forwarding.

To enhance security, the web server can transition to HTTPS, employing port 443 by default, also customizable. The HTTPS implementation utilizes TLS 1.0, TLS 1.1, and TLS 1.2, with RSA serving as the key exchange/agreement and authentication mechanism.

It's worth noting that TCW242 utilizes a self-signed certificate for HTTPS, a choice that may trigger a "Your connection is not private" warning in some web browsers.

Keep in mind that the TCW242, akin to any embedded device, operates with limited resources. Running in HTTPS mode is slower than in HTTP mode due to these constraints. When in HTTPS mode, we recommend using Firefox, given its notably lower resource requirements compared to other browsers.

HTTP API access authentication by default it is active. Authentication details are same as WEB access. The controller support two types of authentication – see the explanation for HTTP API below.

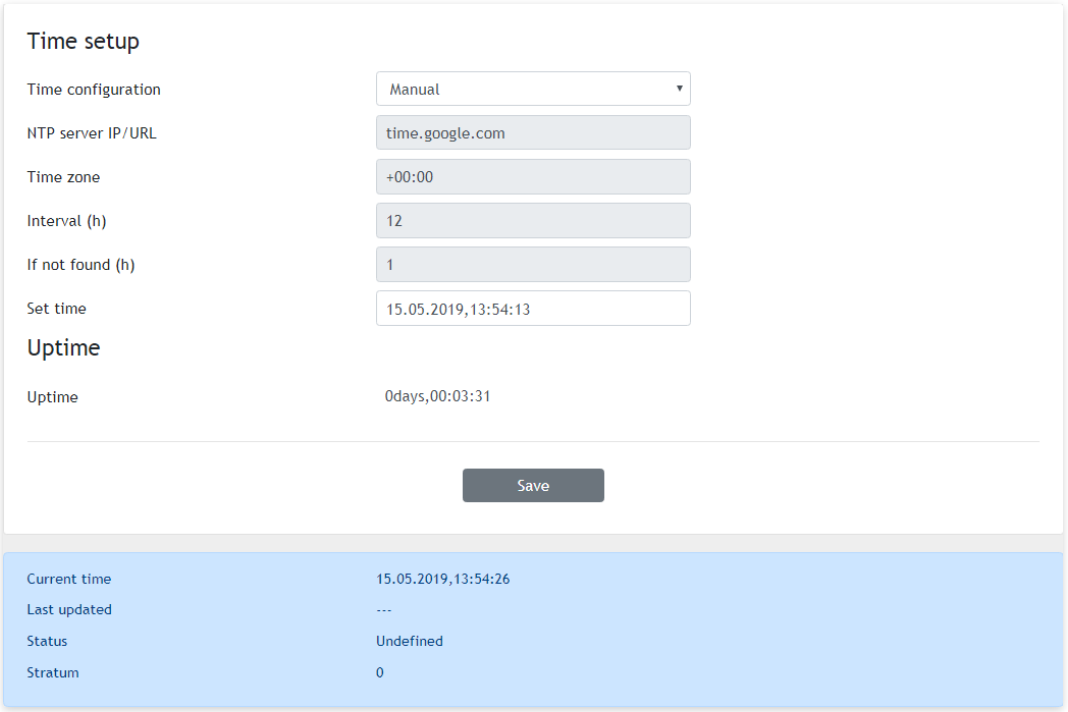
The refresh interval can be set between 0 and 253 seconds. Zero means no automatic refresh.

Alarms colors are fixed and are presented here just for information.

8.3. Services

8.3.1. NTP

The internal clock of the controller can be set either manually or using NTP (Network Time Protocol).



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

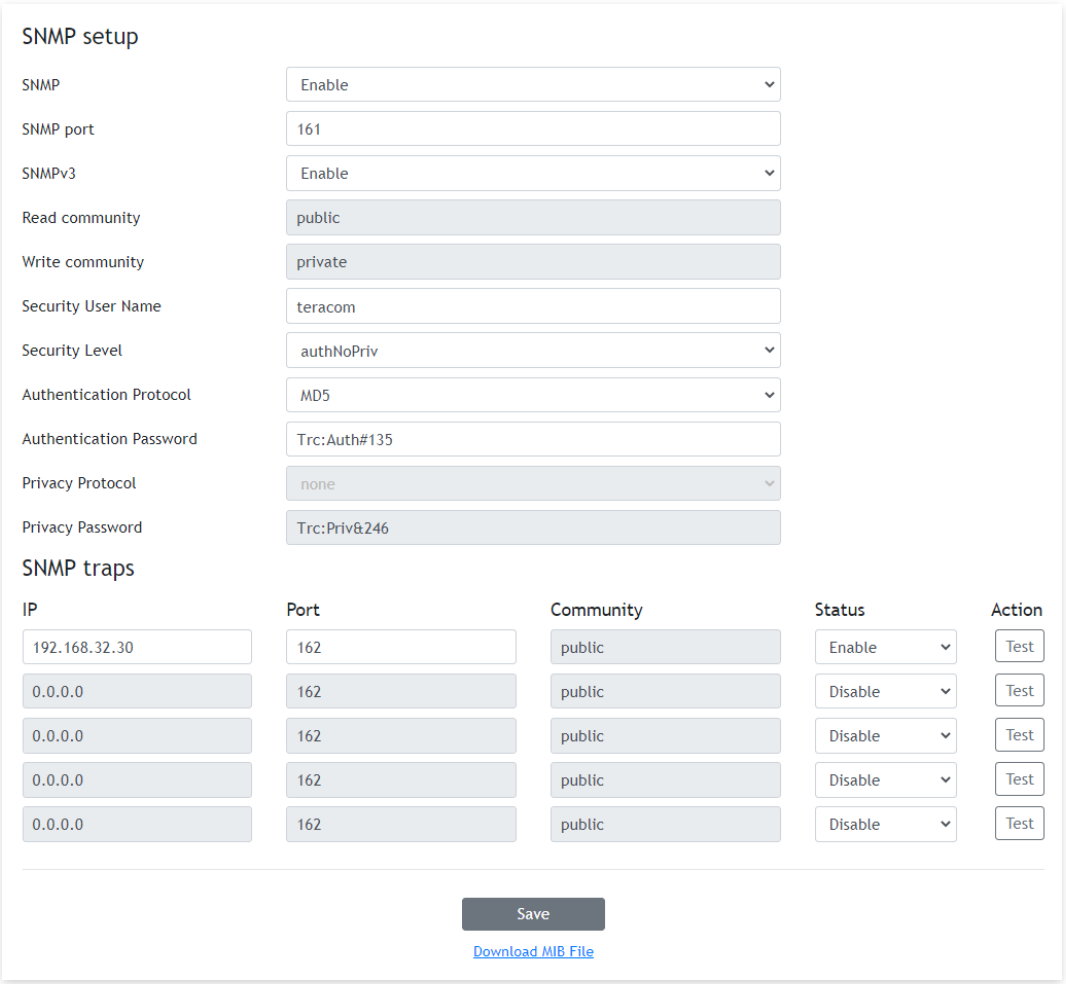
8.3.2. SNMP

TCW242 supports SNMPv2 and SNMPv3 and the default settings are:

- SNMP disabled
- Port 161
- SNMPv3 disabled
- read community public
- write community private
- Security User Name teracom
- Security Level noAuthNoPriv
- Authentication Protocol none
- Authentication Password Trc:Auth#135
- Privacy Protocol none
- Privacy Password Trc:Priv&246

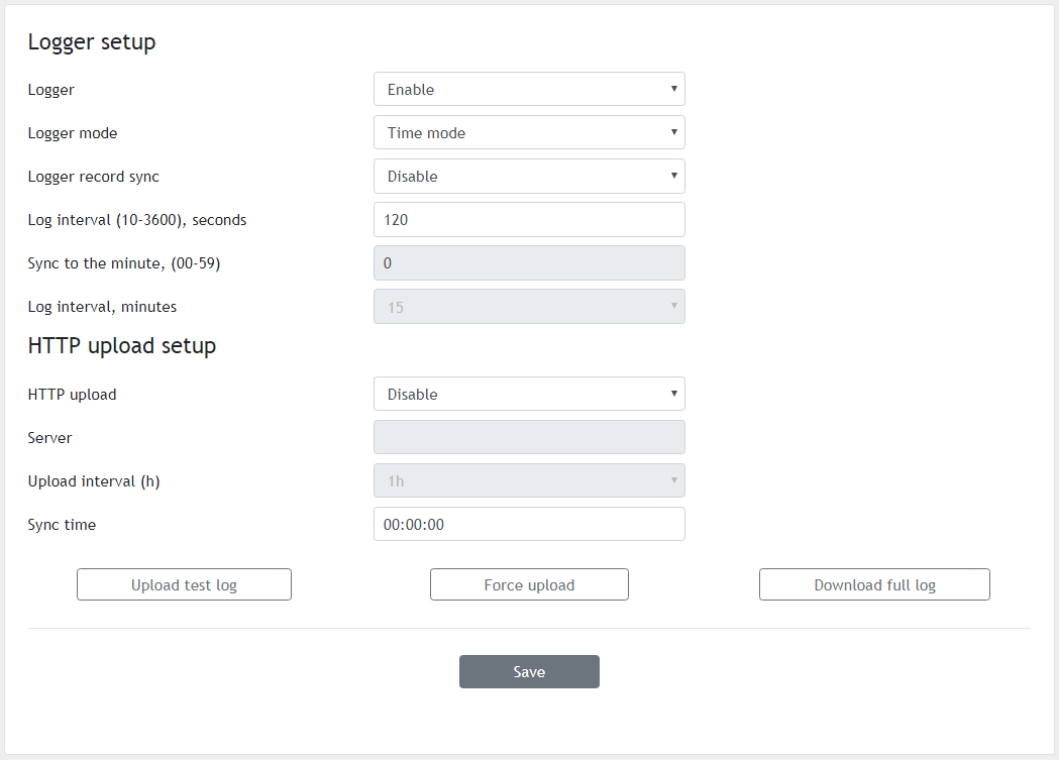
SNMPv3 provides improved security features including authentication and privacy options (noAuthNoPriv, authNoPriv, authPriv). SNMPv3 also supports user-based authentication mechanisms (MD5, SHA, none) and privacy mechanisms (DES, AES, none).

For alarm notifications, up to 5 independent recipients can receive SNMP traps, each with its own port and community. A trap is also sent after reset. The MIB file can be downloaded from the web page.



8.3.3. Logger

The logger operates in three modes: Time, Alarm, and Time & Alarm. The mode determines the trigger for creating a record in the logger's memory. In Time mode, records are made at regular intervals based on the "Log Interval." In Alarm mode, records are made whenever an alarm condition is met. In Time & Alarm mode, a combination of both triggers is used.



The time between two log entries is determined by the log interval. Lowering the log interval increases the accuracy of the records, but reduces the amount of history captured. The logger can be synced to a specific minute in an hour, useful for monitoring utilities like electricity and gas. The log interval can be set from 1 to 60 minutes in a drop-down menu. "Sync to the minute" specifies the minute for syncing and it's best to use the default value, 00.

Example:

The current settings are:

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

With the current settings, 4 records will be recorded per hour at HH:00, HH:15, HH:30, and HH:45. The device has been powered up at 09:12, so the first record will be taken immediately after power-up at 09:12. The subsequent records will be taken at 09:15, 09:30, 09:45, 10:00, 10:15, and so on.

Data can be accessed in two ways: downloading the full log file through the WEB interface or uploading the last un-sent records to a designated HTTP server. The upload frequency can be set to any value between 1 and 24 hours, but real-time clock synchronization is necessary (using the NTP service). The HTTP server can be either a domain or IP address, but proper DNS settings are important.

The "Sync Time" is the time of day when the upload period is synchronized.

For example, if the current time is 19:31, the upload period is 3 hours, and the Sync Time is 9:00, then the upload times will be: 9:00, 12:00, 15:00, 18:00, 21:00, 24:00, 3:00, and 6:00. The first upload after enabling the logger at 19:31 will occur at 21:00.

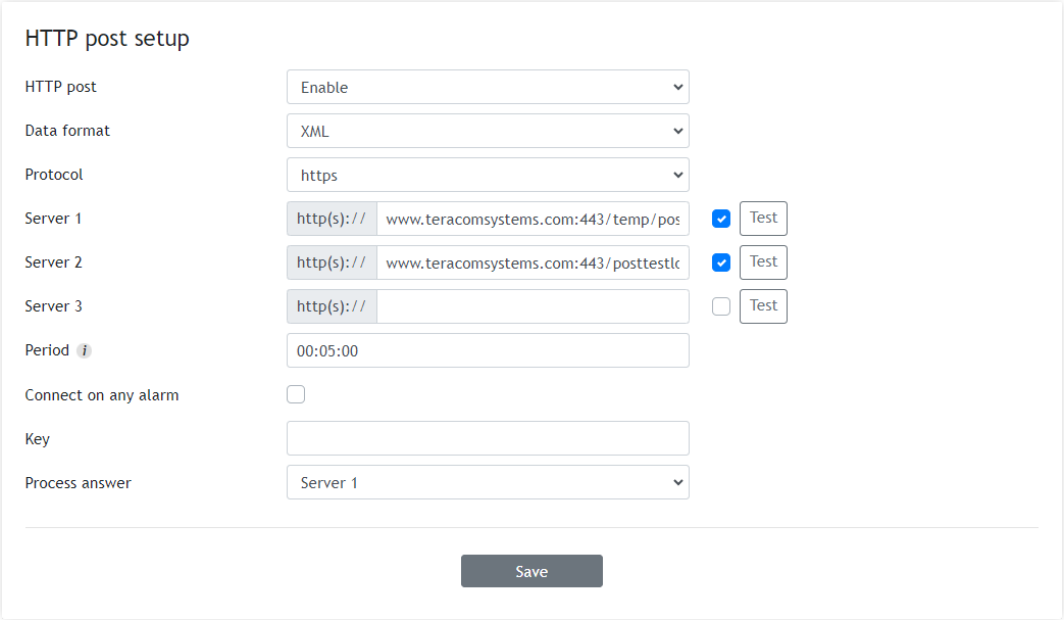
The "Force Upload" button triggers an upload of recorded data between the previous periodical upload and the present moment.

By default, the logger is disabled.

Additional information on the logger can be found in the "Datalogger" section.

8.3.4. HTTP Post

TCW242 can regularly transfer an XML/JSON file to a designated server through HTTP or HTTPS Post. HTTPS uses TLS 1.0, 1.1, or 1.2 with RSA key exchange/agreement and authentication. The XML/JSON file includes current information on all monitored parameters and additional system details. You can select the file format from a drop-down menu.



TCW242 can send an XML/JSON file to 3 servers via HTTP or HTTPS Post. HTTP servers can be addressed by domain name or IP address.

The "Period" of sending, which can range from 10 to 14400 seconds, can be changed remotely via HTTP API. Shortening the "Period" leads to closer real-time operation, but also higher data traffic.

If the "Connect on any alarm" box is checked, an HTTP/HTTPS Post request will be sent during any alarm.

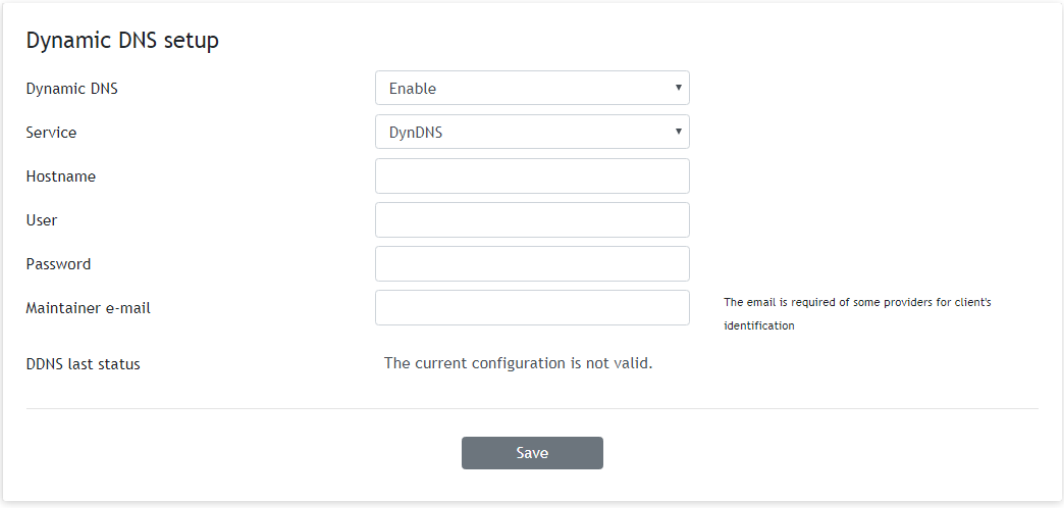
The "Key" field is user-defined and sent in the XML/JSON file for device identification.

If "Process Answer" is enabled, TCW242 will execute commands sent by the remote server as a response to HTTP/HTTPS Post.

Learn more about HTTP/HTTPS Post in the HTTP API section.

8.3.5. Dynamic DNS

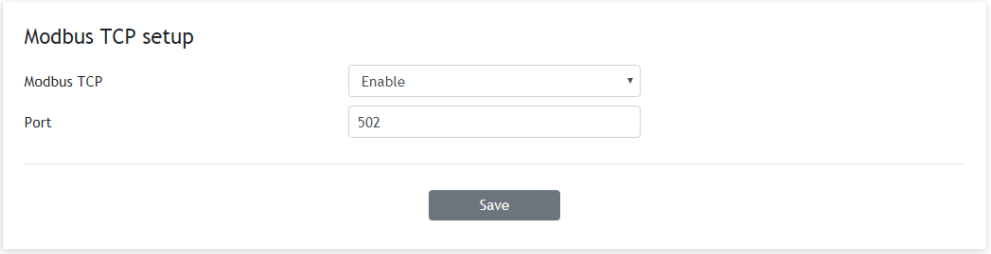
TCW242 supports DynDNS, No-IP, and DNS-O-Matic for dynamic DNS.



This enables access to TCW242 from the internet using only a dynamic public IP address.

8.3.6. MODBUS

TCW242 supports MODBUS over TCP/IP, using port 502 as the default.

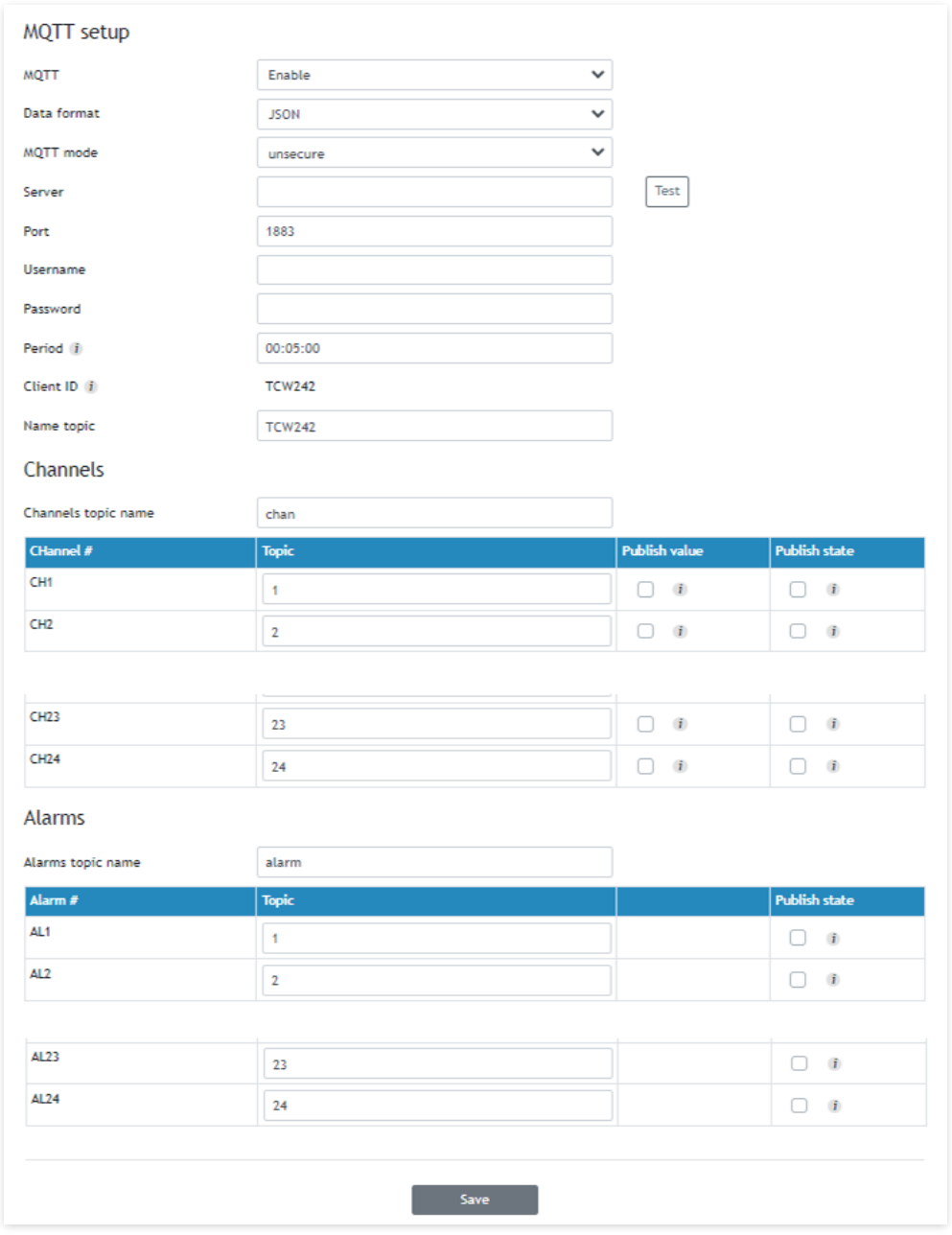


Modbus is disabled by default, but more information about this feature can be found in the MODBUS section.

8.3.7. MQTT

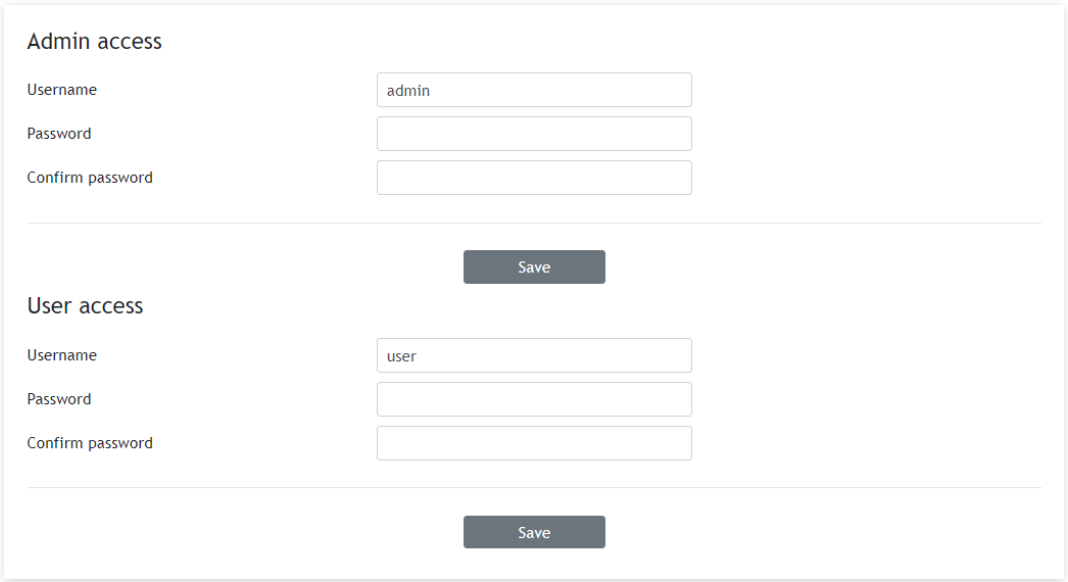
The device supports MQTT 3.1.1 in unsecured and encrypted (SSL/TLS) communication, using JSON and XML data format.

There are independent topics for every channel and alarm.



8.4. Administration

8.4.1. User/Pass

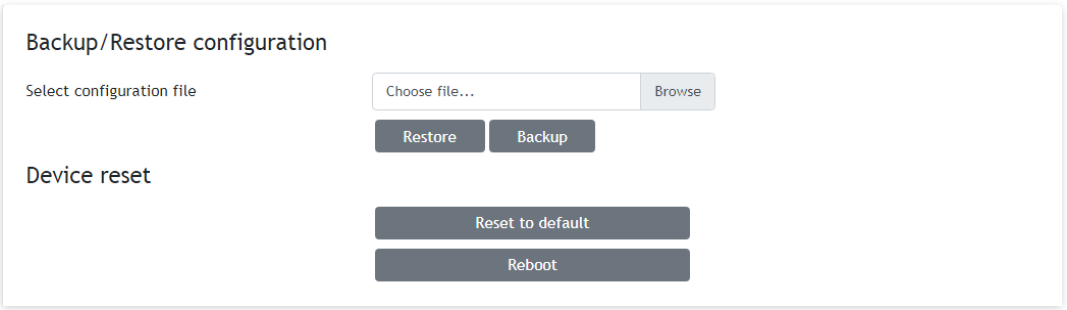


The TCW242 has two user levels: "Admin" with full control and "User" with limited access.

Both usernames and passwords can be up to 31 characters long.

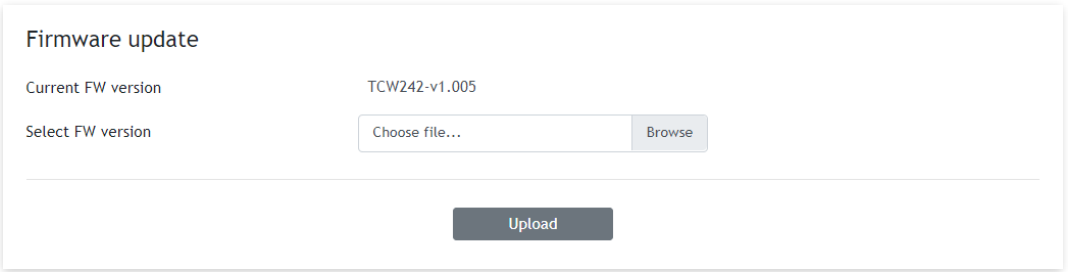
8.4.2. Backup/Restore

The TCW242 supports backing up and restoring all user settings via an XML backup file, making it useful for replicating similar settings across multiple controllers..



8.4.3. FW update

The TCW242 can be updated through its web interface.



To update the device:

- Download the latest firmware from www.teracomsystems.com
- Go to Administration->FW update and select the downloaded .cod file and press "UPLOAD"
- Wait until the update is complete.

Attention! Do not turn off the power during the update, it may damage the device.

8.5. Logout

The TCW242 supports multiple sessions, but it's recommended to log out after finishing work for best practice.

9. Protocols and API

9.1. SNMP

The TCW242 supports SNMP (Simple Network Management Protocol), a standard protocol for managing IP network devices, allowing administrative computers (managers) to monitor and control them. The device can be configured and monitored using an SNMP v.2 or v.3 compatible program, with parameters grouped by function. To obtain a valid OID number, replace "!" with "1.3.6.1.4.1.38783". To save changes, set configurationSaved (OID !.8.6.3.0) to "1".

product

OID	Name	Access	Description	Syntax
!.8.1.1.0	name	read-only	Device name	DisplayString
!.8.1.2.0	version	read-only	Firmware version	DisplayString
!.8.1.3.0	date	read-only	Release date	DisplayString

setup -> network

replace "**!**" with "1.3.6.1.4.1.38783" in the table below

OID	Name	Access	Description	Syntax
!.8.2.1.1.0	deviceID	read-only	Device ID (default MAC address)	MacAddress
!.8.2.1.2.0	hostName	read-only	Hostname	DisplayString (SIZE (0..38))
!.8.2.1.3.0	deviceIP	read-only	Device IP address	IpAddress

setup -> parameters -> mbSensors -> mbSensorsTable -> mbSensorsEntry -> msSensIndex 1 to 24

replace "**?**" with a number from 1 to 24 and "**!**" with "1.3.6.1.4.1.38783" in the table below

OID	Name	Access	Description	Syntax
!.8.2.2.1.1.1.2.**?**.0	mbSenDescription.**?**	read-write	Sensor description	DisplayString
!.8.2.2.1.1.1.3.**?**.0	mbSenMult.**?**	read-write	Sensor multiplier x1000 in Integer format	Integer32
!.8.2.2.1.1.1.4.**?**.0	mbSenOffset.**?**	read-write	Sensor offset x1000 in Integer format	Integer32
!.8.2.2.1.1.1.5.**?**.0	mbSenVal.**?**	read-only	Sensor value x1000 in Integer format	Integer32
!.8.2.2.1.1.1.6.**?**.0	mbSenCounter.**?**	read-only	Sensor as 32-bit Counter	Counter32

setup -> parameters -> analogInputs -> analogInpTable -> analogInpEntry -> analogInpIndex 1 to 4

replace "**?**" with a number from 1 to 4 and "**!**" with "1.3.6.1.4.1.38783" in the table below

OID	Name	Access	Description	Syntax
!.8.2.2.2.1.1.2.**?**.0	analogInpDescription.**?**	read-write	Analog input description	DisplayString
!.8.2.2.2.1.1.3.**?**.0	analogInpMult.**?**	read-write	Analog input multiplier x1000 in Integer format	Integer32
!.8.2.2.2.1.1.4.**?**.0	analogInpOffset.**?**	read-write	Analog input offset x1000 in Integer format	Integer32
!.8.2.2.2.1.1.5.**?**.0	analogInpMode.**?**	read-write	Analog input mode - 0-10V or 4-20mA	Integer32
!.8.2.2.2.1.1.6.**?**.0	analogInpValue.**?**	read-only	Analog input value x1000 in Integer format	Integer32

setup -> parameters -> digitalInputs -> digitalInpTable -> digitalInpEntry -> digitalInpIndex 1 to 4

replace "**?**" with a number from 1 to 4 and "**!**" with "1.3.6.1.4.1.38783" in the table below

OID	Name	Access	Description	Syntax
!.8.2.2.3.1.1.2.**?**.0	digInpDescription.**?**	read-write	Digital Input description	DisplayString
!.8.2.2.3.1.1.3.**?**.0	digInplowLevel.**?**	read-write	Digital Input closed state	DisplayString
!.8.2.2.3.1.1.4.**?**.0	digInphighLevel.**?**	read-write	Digital Input open state	DisplayString
!.8.2.2.3.1.1.5.**?**.0	digInpMode.**?**	read-write	Digital Input mode - Discrete or Counter	INTEGER { openclosed(0), risingEdge(1), fallingEdge(2), bothEdges(3) }
!.8.2.2.3.1.1.6.**?**.0	digInpCloseToOpenDelay.**?**	read-write	Digital input Close To Open delay	Integer32(0..60000)

!.8.2.2.3.1.1.7.?.0	digInpOpenToCloseDelay.?	read-write	Digital input Open To Close delay	Integer32(0..60000)
!.8.2.2.3.1.1.8.?.0	digInpCounterInitValue.?	read-only	Digital input counter initial value	Integer32
!.8.2.2.3.1.1.9.?.0	digInpValue.?	read-only	Digital input value	Unsigned32

setup -> parameters -> relaySetup -> relOutSetupTable -> relOutSetupEntry -> relOutSetupIndex 1 to 4
replace "**?**" with a number from 1 to 4 and "**!**" with "1.3.6.1.4.1.38783" in the table below

OID	Name	Access	Description	Syntax
!.8.2.2.4.1.1.2.?.0	relOutDescription.?	read-write	Relay description	DisplayString
!.8.2.2.4.1.1.3.?.0	relOutPulseWidth.?	read-write	Relay Pulse x100ms	Integer32
!.8.2.2.4.1.1.4.?.0	relOutActivation.?	read-write	Relay activated from	INTEGER {webHttpApi(0), al01(1), al02(2), al03(3), al04(4), al05(5), al06(6), al07(7), al08(8), al09(9), al10(10), al11(11), al12(12), al13(13), al14(14), al15(15), al16(16), al17(17), al18(18), al19(19), al20(20), al21(21), al22(22), al23(23), al24(24), anyAlarm(25), schedule1(26), schedule2(27), schedule3(28), schedule4(29) }
!.8.2.2.4.1.1.5.?.0	relOutAction.?	read-write	Relay action on alarm condition	INTEGER {on(0), pulse(2)}

OID	Name	Access	Description	Syntax
!.8.2.2.4.2.0	relOutAfterRst	read-write	Relay state after restart	INTEGER {off(0), on(1), laststate(2)}

monitorNcontrol -> channels -> chanTable -> chanEntry -> chIndex 1 to 24
replace "**?**" with a number from 1 to 24 and "**!**" with "1.3.6.1.4.1.38783" in the table below

OID	Name	Access	Description	Syntax
!.8.3.1.1.1.2.?.0	chType.?	read-write	Channel type	INTEGER {general(0), cumulative(1), discrete(2), counter(3)}
!.8.3.1.1.1.3.?.0	chdescription.?	read-write	Channel description	DisplayString
!.8.3.1.1.1.4.?.0	chParam1.?	read-write	Channel parameter 1	INTEGER {none(0), s01(3), s02(4), s03(5), s04(6), s05(7), s06(8), s07(9), s08(10),s09(11), s10(12), s11(13), s12(14), s13(15), s14(16), s15(17), s16(18), s17(19), s18(20), s19(21), s20(22), s21(23), s22(24), s23(25), s24(26), a01(27), a02(28), a03(29), a04(30), a05(31), a06(32), d01(33), d02(34), d03(35), d04(36)}
!.8.3.1.1.1.5.?.0	chOP1.?	read-write	Channel operand 1	INTEGER{none(0), multiplication(1), division(2), sum(3), subtract(4)}
!.8.3.1.1.1.6.?.0	chParam2.?	read-write	Channel parameter 2	INTEGER {none(0), one(1), null(2), s01(3), s02(4), s03(5), s04(6), s05(7), s06(8), s07(9), s08(10),s09(11), s10(12), s11(13), s12(14), s13(15), s14(16), s15(17),

				s16(18), s17(19), s18(20), s19(21), s20(22), s21(23), s22(24), s23(25), s24(26),a01(27), a02(28), a03(29), a04(30), a05(31), a06(32),d01(33), d02(34), d03(35), d04(36)}
!.8.3.1.1.1.7.?.0	chOP2.?	read-write	Channel operand 2	INTEGER {none(0), multiplication(1), division(2), sum(3), subtract(4)}
!.8.3.1.1.1.8.?.0	chCoef1.?	read-write	Channel coefficient 1 x1000 in Integer format	Integer32
!.8.3.1.1.1.9.?.0	chOP3.?	read-write	Channel operand 3	INTEGER {none(0), multiplication(1), division(2), sum(3), subtract(4)}
!.8.3.1.1.1.10.?.0	chCoef2.?	read-write	Channel coefficient 2 x1000 in Integer format	Integer32
!.8.3.1.1.1.11.?.0	chUnit.?	read-write	Channel unit	DisplayString
!.8.3.1.1.1.12.?.0	chCumulInitValue.?	read-write	Channel cumulative initial value	Integer32
!.8.3.1.1.1.13.?.0	chValue.?	read-only	Channel value x1000 in Integer format	Integer32
!.8.3.1.1.1.14.?.0	chCounter.?	read-only	Channel as 32-bit counter	Counter32
!.8.3.1.1.1.15.?.0	chAlarmStatus.?	read-only	Channel alarm status	INTEGER {undefined(0), normal(1), indeterminate(2), warning(3), minor(4), major(5), critical(6)}

monitorNcontrol -> relays -> relOutTable -> relOutEntry -> relayIndex 1 to 4

replace "**?**" with a number from 1 to 4 and "**!**" with "1.3.6.1.4.1.38783" in the table below

OID	Name	Access	Description	Syntax
!.8.3.2.1.1.2.?.0	relayState.?	read-write	Relay state	INTEGER {off(0), on(1)}
!.8.3.2.1.1.3.?.0	relayPulse.?	read-write	Relay pulse	INTEGER {off(0), on(1)}

monitorNcontrol -> alarmsTable -> alarmsEntry -> alIndex 1 to 24

replace "**?**" with a number from 1 to 24 and "**!**" with "1.3.6.1.4.1.38783" in the table below

OID	Name	Access	Description	Syntax
!.8.3.3.1.1.2.?.0	alDescription.?	read-write	Alarm description	DisplayString
!.8.3.3.1.1.3.?.0	alCond1Channel.?	read-write	Alarm condition 1 channel	INTEGER {none(0), v01(1), v02(2), v03(3), v04(4), v05(5), v06(6), v07(7), v08(8), v09(9), v10(10), v11(11), v12(12), v13(13), v14(14), v15(15), v16(16), v17(17), v18(18), v19(19), v20(20), v21(21), v22(22), v23(23), v24(24)}
!.8.3.3.1.1.4.?.0	alCond1Operand.?	read-write	Alarm condition 1 operand	INTEGER{larger(1), less(2)}
!.8.3.3.1.1.5.?.0	alCond1Limit.?	read-write	Alarm condition 1 limit x1000 in Integer format	Integer32
!.8.3.3.1.1.6.?.0	alCond1Hys.?	read-write	Alarm condition 1 hysteresis x1000 in Integer format	Integer32
!.8.3.3.1.1.7.?.0	alCond1AlarmState.?	read-write	Alarm condition 1 discrete alarm state	INTEGER {open(0), closed(1)}
!.8.3.3.1.1.8.?.0	alCondLogic.?	read-write	Alarm conditions logic	INTEGER{none(0), and(1), or(2)}
!.8.3.3.1.1.9.?.0	alCond2Channel.?	read-write	Alarm condition 2 channel	INTEGER {none(0), v01(1), v02(2), v03(3), v04(4), v05(5), v06(6), v07(7), v08(8), v09(9), v10(10), v11(11), v12(12),

				v13(13), v14(14), v15(15), v16(16), v17(17), v18(18), v19(19), v20(20), v21(21), v22(22), v23(23), v24(24)}
!.8.3.3.1.1.10.?.0	alCond2Operand.?	read-write	Alarm condition 2 operand	INTEGER{larger(1), less(2)}
!.8.3.3.1.1.11.?.0	alCond2Limit.?	read-write	Alarm condition 2 limit x1000 in Integer format	Integer32
!.8.3.3.1.1.12.?.0	alCond2Hys.?	read-write	Alarm condition 2 hysteresis x1000 in Integer format	Integer32
!.8.3.3.1.1.13.?.0	alCond2AlarmState.?	read-write	Alarm condition 2 discrete alarm state	INTEGER {open(0), closed(1)}
!.8.3.3.1.1.14.?.0	alType.?	read-write	Alarm type	INTEGER {warning(3), minor(4), major(5), critical(6)}
!.8.3.3.1.1.15.?.0	alAssigned.?	read-write	Alarm assigned to	INTEGER {none(0), v01(1), v02(2), v03(3), v04(4), v05(5), v06(6), v07(7), v08(8),v09(9), v10(10), v11(11), v12(12), v13(13), v14(14), v15(15), v16(16),v17(17), v18(18), v19(19), v20(20), v21(21), v22(22), v23(23), v24(24)}
!.8.3.3.1.1.16.?.0	alActionDelay.?	read-write	Alarm action delay	Integer32
!.8.3.3.1.1.17.?.0	alActionOnReturn.?	read-write	Alarm action on return	INTEGER {no(0), yes(1)}
!.8.3.3.1.1.18.?.0	alAction1.?	read-write	Alarm action 1	INTEGER {none(0), trapcond1(1), trapcond2(2), trapcond1and2(3), postiostate(4)}
!.8.3.3.1.1.19.?.0	alAction2.?	read-write	Alarm action 2	INTEGER {none(0), trapcond1(1), trapcond2(2), trapcond1and2(3), postiostate(4)}

!.8.3.3.1.1.20.?.0	alAction3.?	read-write	Alarm action 3	INTEGER {none(0), trapcond1(1), trapcond2(2), trapcond1and2(3), postiostate(4)}
!.8.3.3.1.1.21.?.0	alStatus.?	read-write	Alarm status	INTEGER {undefined(0), normal(1), indeterminate(2), warning(3), minor(4), major(5), critical(6)}

monitorNcontrol

replace "**!**" with "1.3.6.1.4.1.38783" in the table below

OID	Name	Access	Description	Syntax
!.8.3.4.0	configurationSaved	read-write	Configuration save status SAVED/UNSAVED	INTEGER { unsaved(0), saved(1) }
!.8.3.5.0	restartDevice	read-write	Restart Device	INTEGER { cancel(0), restart(1) }
!.8.3.6.0	hardwareErr	read only	Hardware Error	INTEGER { noErr(0), hwErr(1) }

monitorNcontrol-> funcTable -> funcEntry -> funcIndex 1 to 4

replace "**?**" with a number from 1 to 4 and "**!**" with "1.3.6.1.4.1.38783" in the table below

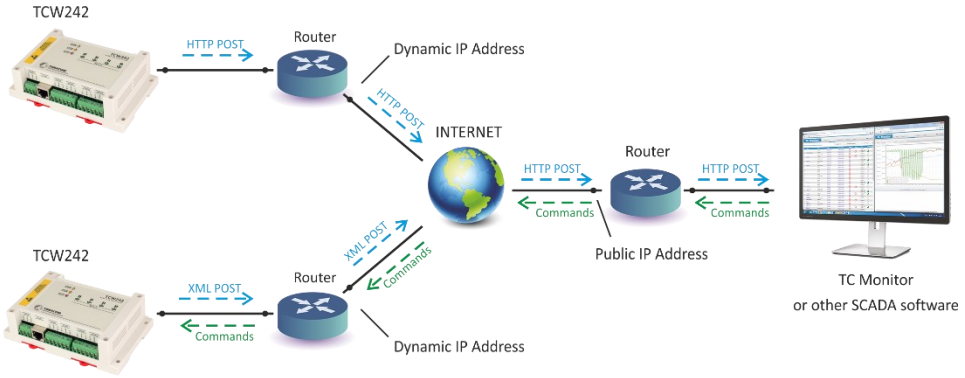
OID	Name	Access	Description	Syntax
!.8.3.7.1.1.2.?.0	funcState	read only	Function state	INTEGER { false(0), true(1) }
!.8.3.7.1.1.3.?.0	funcAl	read only	Function alarm	INTEGER { normal(0), alarm(1) }

9.2. HTTP API

9.2.1. HTTP Post

The TCW242 can perform HTTP/HTTPS Posts to upload XML/JSON files to a designated server. This feature is useful when the controller is located behind a router without a public IP address or when access to the router configuration is unavailable. The server must have a public IP address.

A typical monitoring application is depicted in the following illustration:



HTTP/HTTPS Posts can be sent at set intervals or upon the trigger of an alarm.

The server can also send HTTP Gets with specific commands in response, as described in section 9.2.3. HTTP commands.

Steps to test HTTP/HTTPS Post:

- Save the 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!";
        }
?>
```

- Copy the post.php file to a public web server with PHP support. Verify the script by typing the URL (e.g. www.yourserverURL.com/post.php) in a web browser. If successful, a page with "The PHP script is working!" will be displayed.
- Configure the controller to send HTTP/HTTPS POST to your web server. Enter the URL (yourserverURL.com/post.php) in the URL field and click "Test HTTP Post".
- If the HTTP/HTTPS POST is received and processed, "OK" will appear near the button. Additionally, an XML file with time information (e.g. 20220420103318_status.xml) will be created in the same directory as post.php.

9.1.1. HTTP/HTTPS Get

HTTP/HTTPS Get can be used to monitor TCW242 via XML or JSON files. The format is as follows:

http(s)://device.ip.address/status.xml

http(s)://device.ip.address/status.json

For more details on the structure of the files, see Appendix A XML file structure and Appendix B JSON file structure.

HTTP/HTTPS Get can be sent at any time to TCW242 if it is on the same network or it has appropriate routing.

If there isn't direct access to the device, HTTP/HTTPS Get can be sent immediately after HTTP/HTTPS Post receiving from the same device.

9.1.1.1. Commands

All command used with HTTP/HTTPS Post can be used also with HTTP/HTTPS Get. The right format is:

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

Where:

yyy is the command;

xxx is the parameter.

Example:

http(s)://device.ip.address/status.xml?pper=300, will set Post period = 300 sec.

9.1.1.2. HTTP/HTTPS 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 shown in the table below:

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

Example:

http(s)://device.ip.address/status.xml?a=admin:admin&pper=120 will set post period on 120 sec in case the username=admin and pass=admin

9.2.2. List of HTTP API commands

Command	Description
dataf=x	Data format XML/JSON for HHTP Post – 0 XML, 1 JSON
pushtls=x	HTTP(S) protocol, where x is 0 for HTTP and 1 for HTTPS
purl1=yyy	URL for HTTP Post to Server 1, where yyy is a full path to the PHP file. Example: purl=212.25.45.120:30181/xampp/test/posttest1.php
purl2=yyy	URL for HTTP Post to Server 1, where yyy is a full path to the PHP file. Example: purl=212.25.45.120:30181/xampp/test/posttest2.php
purl3=yyy	URL for HTTP Post to Server 1, where yyy is a full path to the PHP file. Example: purl=212.25.45.120:30181/xampp/test/posttest3.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
ron=**n**	Turn relay ON (**n** is 1,2,4 or 8) ron=1 - will turn ON relay 1 ron=2 - will turn ON relay 2 ron=4 - will turn ON relay 3 ron=8 - will turn ON relay 4
ron=1&ron=2&ron=4&ron=8	Turn four relays ON
rof=**n**	Turn relay OFF (**n** is 1,2,4 or 8) rof=1 - will turn OFF relay 1 rof=2 - will turn OFF relay 2 rof=4 - will turn OFF relay 3 rof=8 - will turn OFF relay 4
rof=1&rof=2&rof=4&rof=8	Turn four relays OFF
rpl=**n**	Pulse relay **n** (**n** is 1,2,4 or 8) rpl=1 – will pulse relay 1 rpl=2 – will pulse relay 2 rpl=4 – will pulse relay 3 rpl=8 – will pulse relay 4
mdata=x	Data format JSON/Plain text for MQTT Publish – 0 JSON, 1 Plain text
mmode=x	Publish protocol, where x is 0 for unsecure and 1 for TLS/SSL
muser=xxxx	Username authentication for MQTT, where xxxx is a username
mpass=xxxx	Password authentication for MQTT, where xxxx is a password
murl=yyy	URL for MQTT publish, where yyy is a path murl=212.25.45.120
mport=yyyy	Port for MQTT publish, where yyyy is a port mport=1883
mper=x	MQTT publish period in seconds (x is between 60 and 172800) mper=600 – will set MQTT publish period to 600 seconds
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.)

9.3. MODBUS TCP/IP

The Modbus protocol is a communication standard that was first introduced in 1979 by Modicon. It enables communication between a master device and a slave device in SCADA systems

9.3.1. Codes and answers

9.3.1.1. Read Coil Status (FC=01)

Request
Command to obtain ON/OFF status of discrete coils at address 100.
01 0064 0001
Response
01 01 01

9.3.1.2. Force Single Coil (FC=05)

Request
This command sets the state of discrete address 100 to "ON".
05 0064 FF00
Response
The typical response is a copy of the request, sent back after writing to the coil.
05 0064 FF00

9.3.1.1. Read Discrete Inputs (FC=02)

Request
This command is asking for the status of 4 discrete inputs.

02 0064 0004
Response
02 01 0C

9.3.1.3. Read Holding Registers (FC=03)

Request
This command requests the value of holding register 19300.
03 4B64 0002
Response
03 04 41DD 4210

Request
This command requests the value of holding register 18100.
03 46B4 0008
Response
03 10 54 65 6D 70 65 72 61 74 75 72 65 00 00 00 00 00

9.3.1.2. Write Single Register (FC=06)

Request
This command writes a value to the single register at address 18300.
06 477C 0003
Response
06 477C 0003

9.3.1.3. Write Multiple Registers (FC=16)

Request
This command writes a value to the single register at address 18300.
10 477C 0002 04 0003 0004
Response
10 477C 0002

9.3.1.4. Exception codes

All exceptions are indicated by adding 0x80 to the function code in the request, followed by a single reason byte. For example:

01 Illegal function:

The function code received in the query is not a valid action for the controller.

02 Illegal data address:

The data address received in the query is not a valid address for the slave. Specifically, the reference number and transfer length combination is invalid. A request with offset 96 and length 4 would succeed, but a request with offset 96 and length 5 would result in exception code 02.

9.3.2. Short address table

Note: Changes can be saved by setting "Configuration Saved" to 1.

Parameter	FC	PDU decimal address	Data size	Data
Relay 1	01,05,15	100	Discrete	
Relay 2	01,05,15	101	Discrete	
Relay 3	01,05,15	102	Discrete	
Relay 4	01,05,15	103	Discrete	
Digital input 1	02	100	Discrete	
Digital input 2	02	101	Discrete	
Digital input 3	02	102	Discrete	
Digital input 4	02	103	Discrete	
Analog input 1 value	03	14400	32-bit Float	
Analog input 2 value	03	14402	32-bit Float	
Analog input 3 value	03	14404	32-bit Float	
Analog input 4 value	03	14406	32-bit Float	
Channel 1 value	03	19300	32-bit Float	
Channel 2 value	03	19302	32-bit Float	
Channel 3 value	03	19304	32-bit Float	
Channel 4 value	03	19306	32-bit Float	
Channel 5 value	03	19308	32-bit Float	
Channel 6 value	03	19310	32-bit Float	
Channel 7 value	03	19312	32-bit Float	
Channel 8 value	03	19314	32-bit Float	
Channel 9 value	03	19316	32-bit Float	
Channel 10 value	03	19318	32-bit Float	
Channel 11 value	03	19320	32-bit Float	

Channel 12 value	03	19322	32-bit Float	
Channel 13 value	03	19324	32-bit Float	
Channel 14 value	03	19326	32-bit Float	
Channel 15 value	03	19328	32-bit Float	
Channel 16 value	03	19330	32-bit Float	
Channel 17 value	03	19332	32-bit Float	
Channel 18 value	03	19334	32-bit Float	
Channel 19 value	03	19336	32-bit Float	
Channel 20 value	03	19338	32-bit Float	
Channel 21 value	03	19340	32-bit Float	
Channel 22 value	03	19342	32-bit Float	
Channel 23 value	03	19344	32-bit Float	
Channel 24 value	03	19346	32-bit Float	
Alarm 1 status	03	22000	16-bit unsign int	undefined(0), normal(1), indeterminate(2), warning(3), minor(4), major(5), critical(6)
Alarm 2 status	03	22001	16-bit unsign int	—"—
Alarm 3 status	03	22002	16-bit unsign int	—"—
Alarm 4 status	03	22003	16-bit unsign int	—"—
Alarm 5 status	03	22004	16-bit unsign int	—"—
Alarm 6 status	03	22005	16-bit unsign int	—"—
Alarm 7 status	03	22006	16-bit unsign int	—"—
Alarm 8 status	03	22007	16-bit unsign int	—"—
Alarm 9 status	03	22008	16-bit unsign int	—"—
Alarm 10 status	03	22009	16-bit unsign int	—"—
Alarm 11 status	03	22010	16-bit unsign int	—"—
Alarm 12 status	03	22011	16-bit unsign int	—"—
Alarm 13 status	03	22012	16-bit unsign int	—"—
Alarm 14 status	03	22013	16-bit unsign int	—"—
Alarm 15 status	03	22014	16-bit unsign int	—"—
Alarm 16 status	03	22015	16-bit unsign int	—"—
Alarm 17 status	03	22016	16-bit unsign int	—"—
Alarm 18 status	03	22017	16-bit unsign int	—"—
Alarm 19 status	03	22018	16-bit unsign int	—"—
Alarm 20 status	03	22019	16-bit unsign int	—"—
Alarm 21 status	03	22020	16-bit unsign int	—"—
Alarm 22 status	03	22021	16-bit unsign int	—"—
Alarm 23 status	03	22022	16-bit unsign int	—"—
Alarm 24 status	03	22023	16-bit unsign int	—"—
Save configuration	03,06	24000	16-bit unsign int	unsaved(0), saved(1)

The full address table is available in Appendix C.

10. Data Logger

The logger uses a FLASH memory circular buffer for storage, where new data overwrites the oldest when full. The full log is always available for download and cannot be cleared.
The number of records stored varies depending on the description length and characters used (up to 52371 in the worst case, enough for 36 days with 1-minute records). Typically, the logger can store 71400 records (enough for 49 days with 1-minute records).
The records can be uploaded as a CSV file with a semicolon delimiter to an HTTP server in intervals of 1 to 24 hours, with the first row being a header including the record ID and time stamp.

The structure of one row (a record) of the log is as follows:

ID	Time	Type of record	Channels - values/units	Relays - conditions	Channels - states/units	Alarms - values/descriptions

ID	32-bit unique number for every row (record).
Time	a time stamp of record, in format dd.mm.yyyy, hh:mm:ss.
Type of record	following types of records are available: "Time" for periodical record; "Event" for record initiated by alarm condition; "Type" for header record; "Start" after power-up condition; "Restart" after reset condition; "Power Down" after power-down condition; "Bad" for a problematic record
Channels - values/units	Channels 1 to 24 values/units
Relays - conditions	Relays 1 to 4 conditions
Channels - states/units	Channels 1 to 24 states/units For chanel types General, Cumulative and Discrete following states are available: 0 – “Undefined” 1 – “Normal” 2 – “Indeterminate” 3 – “Warning” 4 – “Minor” 5 – “Major” 6 – “Critical” For channel type Counter following states are available: 8 – “Undefined” 9 – “Normal” 10 – “Indeterminate” 11 – “Warning” 12 – “Minor” 13 – “Major” 14 – “Critical”
Alarms - values/descriptions	Alarms 1 to 24 values/descriptions The wollowing alarm values are available: 0 – “Undefined” 1 – “Normal” 2 – “Indeterminate” 3 – “Warning” 4 – “Minor” 5 – “Major”

6 – “Critical”

An example of the log file /fragment channels - values/units/:

ID;Time;Type;Ch1/°C;Ch2/%RH;Ch3;Ch4;Ch5;Ch6;Ch7;Ch8;Ch9;Ch10;Ch11;Ch12;Ch13;Ch14;Ch15;Ch16;Ch17;Ch18;Ch19;Ch20;Ch21;Ch22;Ch23;Ch24....
25114;19.07.2022,16:49:49; Time;25.319;54.512;118.833;229.877;0.000;6587.396;;;;;;;;;;;;;;;;;;;;....

An example of the log file /fragment relays - conditions/:

ID;Time;Type;...R1,R2,R3,R4....
25114;19.07.2022,16:49:49; Time;1,1,0,0;;;;;;;;;;;;;;;;;;;;....

An example of the log file /fragment channels - states/units/:

ID;Time;Type;....ChSt1/°C;ChSt2/%RH;ChSt3;ChSt4;ChSt5;ChSt6;ChSt7;ChSt8;ChSt9;ChSt10;ChSt11;ChSt12;ChSt13;ChSt14;ChSt15;ChSt16;ChSt17;ChSt18;ChSt19;ChSt20;ChSt21;ChSt22;ChSt23;ChSt24....
25114;19.07.2022,16:49:49;Time;....3;5;0;2;2;2;;;;;;;;;;;;;;;;;;;;....

An example of the log file /fragment alarms - values/descriptions/:

ID;Time;Type;...Al1/AL1-T;Al2/AL2-H;Al3;Al4;Al5;Al6;Al7;Al8;Al9;Al10;Al11;Al12;Al13;Al14;Al15;Al16;Al17;Al18;Al19;Al20;Al21;Al22;Al23;Al24....
25114;19.07.2022,16:49:49;Time;....3;5;0;0;0;0;;;;;;;;;;;;;;;;;;

11. MQTT

MQTT is a publish/subscribe messaging protocol for client-server communication. It is lightweight, open-source, simple, and easy to implement. MQTT is widely used across industries including automotive, manufacturing, telecommunications, oil and gas.

Learn more about MQTT at www.mqtt.org.

12. Factory default settings

TCW242 can be reset to its original factory settings in three ways:

12.1. Factory reset without network settings

The "Factory Default" button under Administration > Backup/Restore restores all parameters to factory settings, excluding network settings.

12.2. Factory reset for network settings only

Pressing the reset button for over 5 seconds while the device is running will set all network settings to their factory defaults.

12.3. Full factory reset

For a full factory reset, follow these steps:

- Press and hold the RESET button and turn on the power supply
- The yellow LED lights and the red LED flashes about 5 times per second
- Release the reset button after about 5 seconds when the red LED turns off
- The yellow LED flashes once per second and the red LED lights, indicating the device is in working mode with factory defaults.

The factory default settings are:

Username	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
Analog inputs mode	Voltage
Digital inputs mode	Open/Closed

13. Operating environment

The device is designed for use in a pollution level 2 environment and at elevations up to 2000 meters. Any other equipment used in conjunction with the controller must also meet EMC requirements and be suitable for the same operating conditions.

14. Safety precautions

Do not use this device for medical or life-saving purposes or in any situation where its failure could result in serious injury or death.

Use only flexible stranded wire with a cross-section of 0.5mm² or greater for digital and analog inputs and relay output wiring to reduce the risk of fire.

To avoid electric shock and fire hazards, keep the device away from liquids, rain, and moisture. Do not place objects filled with liquids, such as vases, on the device.

Ensure recommended spacing between adjacent devices to prevent overheating of the controller. Leave space for cable attachment/removal after installation.

Teracom does not guarantee proper operation if the product is used outside of its specifications.

For proper device operation, follow these steps:

- Properly install the device, refer to the manual;
- Log in to the device using a browser program;
- Configure the device properly;
- Short the "Din1" and "GND";
- Go to the "Input/Output page" of the WEB interface, and check that the proper parameter value is displayed in the Digital Inputs section;
- Go to the "Modbus sensors page" of the WEB interface and install the Modbus RTU (TSH3XX/TST3XX or third-party) sensor, and check that the proper parameters are displayed;
- The flashing "STS" LED should indicate proper operation.

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.

15. Maintenance

Upon completion of any service or repairs to the device or once per year, a safety check must be performed to determine that this product is in proper operating condition.

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

XML file structure

```
<Monitor>
  <DeviceInfo>
    <DeviceName>TCW242</DeviceName>
    <HostName>TCW242 </HostName>
    <ID>5c:32:c5:00:ac:57</ID>
    <FwVer>TCW242-v3.002</FwVer>
    <MnfInfo>www.teracomsystems.com</MnfInfo>
    <SysContact>info@teracomsystems.com</SysContact>
    <SysName>TCW242</SysName>
    <SysLocation>Location</SysLocation>
  </DeviceInfo>
  <CH>
    <CH1>
      <type>0</type>
      <description>V01-TST300v4-T</description>
      <value>25.601</value>
      <valuebin/>
      <unit/>
      <alarmbin>3</alarmbin>
      <alarm>Warning</alarm>
      <selch>3</selch>
    </CH1>
    <CH2>
      <type>2</type>
      <description>V02-D1</description>
      <value>OPEN</value>
      <valuebin>1</valuebin>
      <unit/>
      <alarmbin>1</alarmbin>
      <alarm>Normal</alarm>
      <selch>31</selch>
    </CH2>
    <CH3>
      <type>2</type>
      <description>V03-D2</description>
      <value>OPEN</value>
      <valuebin>1</valuebin>
      <unit/>
      <alarmbin>1</alarmbin>
      <alarm>Normal</alarm>
      <selch>32</selch>
    </CH3>
    <CH4>
      <type>0</type>
      <description>V04</description>
      <value>---</value>
      <valuebin/>
      <unit/>
      <alarmbin>0</alarmbin>
      <alarm/>
      <selch>0</selch>
    </CH4>
    <CH5>
      <type>0</type>
      <description>V05</description>
      <value>---</value>
      <valuebin/>
      <unit/>
      <alarmbin>0</alarmbin>
      <alarm/>
      <selch>0</selch>
    </CH5>
    <CH6>
      <type>0</type>
      <description>V06</description>
      <value>---</value>
      <valuebin/>
      <unit/>
      <alarmbin>0</alarmbin>
      <alarm/>
      <selch>0</selch>
    </CH6>
    <CH7>
```

```
    <type>0</type>
    <description>V07</description>
    <value>---</value>
    <valuebin/>
    <unit/>
    <alarmbin>0</alarmbin>
    <alarm/>
    <selch>0</selch>
</CH7>
<CH8>
    <type>0</type>
    <description>V08</description>
    <value>---</value>
    <valuebin/>
    <unit/>
    <alarmbin>0</alarmbin>
    <alarm/>
    <selch>0</selch>
</CH8>
<CH9>
    <type>0</type>
    <description>V09</description>
    <value>---</value>
    <valuebin/>
    <unit/>
    <alarmbin>0</alarmbin>
    <alarm/>
    <selch>0</selch>
</CH9>
<CH10>
    <type>0</type>
    <description>V10</description>
    <value>---</value>
    <valuebin/>
    <unit/>
    <alarmbin>0</alarmbin>
    <alarm/>
    <selch>0</selch>
</CH10>
<CH11>
    <type>0</type>
    <description>V11</description>
    <value>---</value>
    <valuebin/>
    <unit/>
    <alarmbin>0</alarmbin>
    <alarm/>
    <selch>0</selch>
</CH11>
<CH12>
    <type>0</type>
    <description>V12</description>
    <value>---</value>
    <valuebin/>
    <unit/>
    <alarmbin>0</alarmbin>
    <alarm/>
    <selch>0</selch>
</CH12>
<CH13>
    <type>0</type>
    <description>V13</description>
    <value>---</value>
    <valuebin/>
    <unit/>
    <alarmbin>0</alarmbin>
    <alarm/>
    <selch>0</selch>
</CH13>
<CH14>
    <type>0</type>
    <description>V14</description>
    <value>---</value>
    <valuebin/>
    <unit/>
    <alarmbin>0</alarmbin>
    <alarm/>
```

```
    <selch>0</selch>
</CH14>
<CH15>
    <type>0</type>
    <description>V15</description>
    <value>---</value>
    <valuebin/>
    <unit/>
    <alarmbin>0</alarmbin>
    <alarm/>
    <selch>0</selch>
</CH15>
<CH16>
    <type>0</type>
    <description>V16</description>
    <value>---</value>
    <valuebin/>
    <unit/>
    <alarmbin>0</alarmbin>
    <alarm/>
    <selch>0</selch>
</CH16>
<CH17>
    <type>0</type>
    <description>V17</description>
    <value>---</value>
    <valuebin/>
    <unit/>
    <alarmbin>0</alarmbin>
    <alarm/>
    <selch>0</selch>
</CH17>
<CH18>
    <type>0</type>
    <description>V18</description>
    <value>---</value>
    <valuebin/>
    <unit/>
    <alarmbin>0</alarmbin>
    <alarm/>
    <selch>0</selch>
</CH18>
<CH19>
    <type>0</type>
    <description>V19</description>
    <value>---</value>
    <valuebin/>
    <unit/>
    <alarmbin>0</alarmbin>
    <alarm/>
    <selch>0</selch>
</CH19>
<CH20>
    <type>0</type>
    <description>V20</description>
    <value>---</value>
    <valuebin/>
    <unit/>
    <alarmbin>0</alarmbin>
    <alarm/>
    <selch>0</selch>
</CH20>
<CH21>
    <type>0</type>
    <description>V21</description>
    <value>---</value>
    <valuebin/>
    <unit/>
    <alarmbin>0</alarmbin>
    <alarm/>
    <selch>0</selch>
</CH21>
<CH22>
    <type>0</type>
    <description>V22</description>
    <value>---</value>
    <valuebin/>
```

```
        <unit/>
        <alarmbin>0</alarmbin>
        <alarm/>
        <selch>0</selch>
    </CH22>
    <CH23>
        <type>0</type>
        <description>V23</description>
        <value>---</value>
        <valuebin/>
        <unit/>
        <alarmbin>0</alarmbin>
        <alarm/>
        <selch>0</selch>
    </CH23>
    <CH24>
        <type>0</type>
        <description>V24</description>
        <value>---</value>
        <valuebin/>
        <unit/>
        <alarmbin>0</alarmbin>
        <alarm/>
        <selch>0</selch>
    </CH24>
</CH>
<AL>
    <AL1>
        <description>AL01-TST300-T</description>
        <alarmbin>3</alarmbin>
        <alarm>Warning</alarm>
        <assign>1</assign>
    </AL1>
    <AL2>
        <description>AL02-D1</description>
        <alarmbin>1</alarmbin>
        <alarm>Normal</alarm>
        <assign>2</assign>
    </AL2>
    <AL3>
        <description>AL03-D2</description>
        <alarmbin>1</alarmbin>
        <alarm>Normal</alarm>
        <assign>3</assign>
    </AL3>
    <AL4>
        <description>AL04</description>
        <alarmbin>0</alarmbin>
        <alarm/>
        <assign>0</assign>
    </AL4>
    <AL5>
        <description>AL05</description>
        <alarmbin>0</alarmbin>
        <alarm/>
        <assign>0</assign>
    </AL5>
    <AL6>
        <description>AL06</description>
        <alarmbin>0</alarmbin>
        <alarm/>
        <assign>0</assign>
    </AL6>
    <AL7>
        <description>AL07</description>
        <alarmbin>0</alarmbin>
        <alarm/>
        <assign>0</assign>
    </AL7>
    <AL8>
        <description>AL08</description>
        <alarmbin>0</alarmbin>
        <alarm/>
        <assign>0</assign>
    </AL8>
    <AL9>
        <description>AL09</description>
```

```
    <alarmbin>0</alarmbin>
    <alarm/>
    <assign>0</assign>
</AL9>
<AL10>
    <description>AL10</description>
    <alarmbin>0</alarmbin>
    <alarm/>
    <assign>0</assign>
</AL10>
<AL11>
    <description>AL11</description>
    <alarmbin>0</alarmbin>
    <alarm/>
    <assign>0</assign>
</AL11>
<AL12>
    <description>AL12</description>
    <alarmbin>0</alarmbin>
    <alarm/>
    <assign>0</assign>
</AL12>
<AL13>
    <description>AL13</description>
    <alarmbin>0</alarmbin>
    <alarm/>
    <assign>0</assign>
</AL13>
<AL14>
    <description>AL14</description>
    <alarmbin>0</alarmbin>
    <alarm/>
    <assign>0</assign>
</AL14>
<AL15>
    <description>AL15</description>
    <alarmbin>0</alarmbin>
    <alarm/>
    <assign>0</assign>
</AL15>
<AL16>
    <description>AL16</description>
    <alarmbin>0</alarmbin>
    <alarm/>
    <assign>0</assign>
</AL16>
<AL17>
    <description>AL17</description>
    <alarmbin>0</alarmbin>
    <alarm/>
    <assign>0</assign>
</AL17>
<AL18>
    <description>AL18</description>
    <alarmbin>0</alarmbin>
    <alarm/>
    <assign>0</assign>
</AL18>
<AL19>
    <description>AL19</description>
    <alarmbin>0</alarmbin>
    <alarm/>
    <assign>0</assign>
</AL19>
<AL20>
    <description>AL20</description>
    <alarmbin>0</alarmbin>
    <alarm/>
    <assign>0</assign>
</AL20>
<AL21>
    <description>AL21</description>
    <alarmbin>0</alarmbin>
    <alarm/>
    <assign>0</assign>
</AL21>
<AL22>
```

```
            <description>AL22</description>
            <alarmbin>0</alarmbin>
            <alarm/>
            <assign>0</assign>
        </AL22>
        <AL23>
            <description>AL23</description>
            <alarmbin>0</alarmbin>
            <alarm/>
            <assign>0</assign>
        </AL23>
        <AL24>
            <description>AL24</description>
            <alarmbin>0</alarmbin>
            <alarm/>
            <assign>0</assign>
        </AL24>
    </AL>
    <R>
        <R1>
            <description>Relay 1</description>
            <value>OFF</value>
            <valuebin>0</valuebin>
            <pulseWidth>0.3</pulseWidth>
            <control>Any Alarm</control>
        </R1>
        <R2>
            <description>Relay 2</description>
            <value>OFF</value>
            <valuebin>0</valuebin>
            <pulseWidth>1.0</pulseWidth>
            <control>Watchdog1</control>
        </R2>
        <R3>
            <description>Relay 3</description>
            <value>OFF</value>
            <valuebin>0</valuebin>
            <pulseWidth>1.0</pulseWidth>
            <control>Watchdog3</control>
        </R3>
        <R4>
            <description>Relay 4</description>
            <value>OFF</value>
            <valuebin>0</valuebin>
            <pulseWidth>1.0</pulseWidth>
            <control>Watchdog4</control>
        </R4>
    </R>
    <FUNC>
        <FUNC1>
            <alarmbin>0</alarmbin>
            <alarm>Normal</alarm>
        </FUNC1>
        <FUNC2>
            <alarmbin>1</alarmbin>
            <alarm>Alarm</alarm>
        </FUNC2>
        <FUNC3>
            <alarmbin>0</alarmbin>
            <alarm>Normal</alarm>
        </FUNC3>
        <FUNC4>
            <alarmbin>0</alarmbin>
            <alarm>Normal</alarm>
        </FUNC4>
    </FUNC>
    <WDG>
        <WDG1>
            <description>Watchdog1</description>
            <modebin>3</modebin>
            <mode>incoming ping</mode>
            <host>0.0.0.0</host>
            <state>0</state>
        </WDG1>
        <WDG2>
            <description>Watchdog2</description>
            <modebin>2</modebin>
```

```
            <mode>outgoing ping</mode>
            <host>192.168.32.112</host>
            <state>0</state>
        </WDG2>
        <WDG3>
            <description>Watchdog3</description>
            <modebin>2</modebin>
            <mode>outgoing ping</mode>
            <host>192.168.32.2</host>
            <state>0</state>
        </WDG3>
        <WDG4>
            <description>Watchdog4</description>
            <modebin>2</modebin>
            <mode>outgoing ping</mode>
            <host>192.168.32.30</host>
            <state>0</state>
        </WDG4>
    </WDG>
    <HTTPPost>
        <Key/>
        <PostPeriod>60</PostPeriod>
    </HTTPPost>
    <MQTT>
        <Period>300</Period>
    </MQTT>
    <Sys>
        <hwerr/>
        <HighAlarmbin>3</HighAlarmbin>
        <HighAlarm>Warning</HighAlarm>
    </Sys>
    <Time>
        <Date>04.01.2024</Date>
        <Time>16:06:23</Time>
    </Time>
</Monitor>
```

Where:

<CH1>… <CH24> - channels;

<AL1> … <AL24> - alarms;

<R1> … <R4> - relays

<FUNC1> … <FUNC4> - functions

<WDG1> … <WDG4> - watchdogs

<valuebin> - number values 0 or 1;

<alarmbin> - number values from 0 to 6;

<alarm> - Undefined, Normal, Indeterminate, Warning, Minor, Major, Critical;

<assign>0</assign> - alarm not assigned to any channel;

<selch>0</selch> - channel is not displayed on Monitoring -> Channels section

<state>1</state> - watchdog is activated

JSON file structure


```
{
  "Monitor": {
    "DeviceInfo": {
      "DeviceName": "TCW242",
      "HostName": "TCW242-E      ",
      "ID": "5c:32:c5:00:ac:57",
      "FwVer": "TCW242-v3.002",
      "MnfInfo": "www.teracomsystems.com",
      "SysContact": "info@teracomsystems.com",
      "SysName": "TCW242",
      "SysLocation": "Location"
    },
    "CH": {
      "CH1": {
        "type": "0",
        "description": "V01-TST300v4-T",
        "value": "26.070",
        "valuebin": "",
        "unit": "",
        "alarmbin": "1",
        "alarm": "Normal",
        "selch":"3"
      },
      "CH2": {
        "type": "2",
        "description": "V02-D1",
        "value": "OPEN",
        "valuebin": "1",
        "unit": "",
        "alarmbin": "1",
        "alarm": "Normal",
        "selch":"31"
      },
      "CH3": {
        "type": "2",
        "description": "V03-D2",
        "value": "OPEN",
        "valuebin": "1",
        "unit": "",
        "alarmbin": "1",
        "alarm": "Normal",
        "selch":"32"
      },
      "CH4": {
        "type": "0",
        "description": "V04",
        "value": "---",
        "valuebin": "",
        "unit": "",
        "alarmbin": "0",
        "alarm": "",
        "selch":"0"
      },
      "CH5": {
        "type": "0",
        "description": "V05",
        "value": "---",
        "valuebin": "",
        "unit": "",
        "alarmbin": "0",
        "alarm": "",
        "selch":"0"
      },
      "CH6": {
        "type": "0",
        "description": "V06",
        "value": "---",
        "valuebin": "",
        "unit": "",
        "alarmbin": "0",
        "alarm": "",
        "selch":"0"
      },
```

```
"CH7": {
   "type": "0",
   "description": "V07",
   "value": "---",
   "valuebin": "",
   "unit": "",
   "alarmbin": "0",
   "alarm": "",
   "selch":"0"
},
"CH8": {
   "type": "0",
   "description": "V08",
   "value": "---",
   "valuebin": "",
   "unit": "",
   "alarmbin": "0",
   "alarm": "",
   "selch":"0"
},
"CH9": {
   "type": "0",
   "description": "V09",
   "value": "---",
   "valuebin": "",
   "unit": "",
   "alarmbin": "0",
   "alarm": "",
   "selch":"0"
},
"CH10": {
   "type": "0",
   "description": "V10",
   "value": "---",
   "valuebin": "",
   "unit": "",
   "alarmbin": "0",
   "alarm": "",
   "selch":"0"
},
"CH11": {
   "type": "0",
   "description": "V11",
   "value": "---",
   "valuebin": "",
   "unit": "",
   "alarmbin": "0",
   "alarm": "",
   "selch":"0"
},
"CH12": {
   "type": "0",
   "description": "V12",
   "value": "---",
   "valuebin": "",
   "unit": "",
   "alarmbin": "0",
   "alarm": "",
   "selch":"0"
},
"CH13": {
   "type": "0",
   "description": "V13",
   "value": "---",
   "valuebin": "",
   "unit": "",
   "alarmbin": "0",
   "alarm": "",
   "selch":"0"
},
"CH14": {
   "type": "0",
   "description": "V14",
   "value": "---",
   "valuebin": "",
   "unit": "",
   "alarmbin": "0",
```

```
    "alarm": "",
    "selch":"0"
  },
  "CH15": {
    "type": "0",
    "description": "V15",
    "value": "---",
    "valuebin": "",
    "unit": "",
    "alarmbin": "0",
    "alarm": "",
    "selch":"0"
  },
  "CH16": {
    "type": "0",
    "description": "V16",
    "value": "---",
    "valuebin": "",
    "unit": "",
    "alarmbin": "0",
    "alarm": "",
    "selch":"0"
  },
  "CH17": {
    "type": "0",
    "description": "V17",
    "value": "---",
    "valuebin": "",
    "unit": "",
    "alarmbin": "0",
    "alarm": "",
    "selch":"0"
  },
  "CH18": {
    "type": "0",
    "description": "V18",
    "value": "---",
    "valuebin": "",
    "unit": "",
    "alarmbin": "0",
    "alarm": "",
    "selch":"0"
  },
  "CH19": {
    "type": "0",
    "description": "V19",
    "value": "---",
    "valuebin": "",
    "unit": "",
    "alarmbin": "0",
    "alarm": "",
    "selch":"0"
  },
  "CH20": {
    "type": "0",
    "description": "V20",
    "value": "---",
    "valuebin": "",
    "unit": "",
    "alarmbin": "0",
    "alarm": "",
    "selch":"0"
  },
  "CH21": {
    "type": "0",
    "description": "V21",
    "value": "---",
    "valuebin": "",
    "unit": "",
    "alarmbin": "0",
    "alarm": "",
    "selch":"0"
  },
  "CH22": {
    "type": "0",
    "description": "V22",
    "value": "---",
```

```
      "valuebin": "",
      "unit": "",
      "alarmbin": "0",
      "alarm": "",
      "selch":"0"
    },
    "CH23": {
      "type": "0",
      "description": "V23",
      "value": "---",
      "valuebin": "",
      "unit": "",
      "alarmbin": "0",
      "alarm": "",
      "selch":"0"
    },
    "CH24": {
      "type": "0",
      "description": "V24",
      "value": "---",
      "valuebin": "",
      "unit": "",
      "alarmbin": "0",
      "alarm": "",
      "selch":"0"
    }
  },
  "AL": {
    "AL1": {
      "description": "AL01-TST300-T",
      "alarmbin": "1",
      "alarm": "Normal",
      "assign":"1"
    },
    "AL2": {
      "description": "AL02-D1",
      "alarmbin": "1",
      "alarm": "Normal",
      "assign":"2"
    },
    "AL3": {
      "description": "AL03-D2",
      "alarmbin": "1",
      "alarm": "Normal",
      "assign":"3"
    },
    "AL4": {
      "description": "AL04",
      "alarmbin": "0",
      "alarm": "",
      "assign":"0"
    },
    "AL5": {
      "description": "AL05",
      "alarmbin": "0",
      "alarm": "",
      "assign":"0"
    },
    "AL6": {
      "description": "AL06",
      "alarmbin": "0",
      "alarm": "",
      "assign":"0"
    },
    "AL7": {
      "description": "AL07",
      "alarmbin": "0",
      "alarm": "",
      "assign":"0"
    },
    "AL8": {
      "description": "AL08",
      "alarmbin": "0",
      "alarm": "",
      "assign":"0"
    },
    "AL9": {
```

```
    "description": "AL09",
    "alarmbin": "0",
    "alarm": "",
    "assign":"0"
  },
  "AL10": {
    "description": "AL10",
    "alarmbin": "0",
    "alarm": "",
    "assign":"0"
  },
  "AL11": {
    "description": "AL11",
    "alarmbin": "0",
    "alarm": "",
    "assign":"0"
  },
  "AL12": {
    "description": "AL12",
    "alarmbin": "0",
    "alarm": "",
    "assign":"0"
  },
  "AL13": {
    "description": "AL13",
    "alarmbin": "0",
    "alarm": "",
    "assign":"0"
  },
  "AL14": {
    "description": "AL14",
    "alarmbin": "0",
    "alarm": "",
    "assign":"0"
  },
  "AL15": {
    "description": "AL15",
    "alarmbin": "0",
    "alarm": "",
    "assign":"0"
  },
  "AL16": {
    "description": "AL16",
    "alarmbin": "0",
    "alarm": "",
    "assign":"0"
  },
  "AL17": {
    "description": "AL17",
    "alarmbin": "0",
    "alarm": "",
    "assign":"0"
  },
  "AL18": {
    "description": "AL18",
    "alarmbin": "0",
    "alarm": "",
    "assign":"0"
  },
  "AL19": {
    "description": "AL19",
    "alarmbin": "0",
    "alarm": "",
    "assign":"0"
  },
  "AL20": {
    "description": "AL20",
    "alarmbin": "0",
    "alarm": "",
    "assign":"0"
  },
  "AL21": {
    "description": "AL21",
    "alarmbin": "0",
    "alarm": "",
    "assign":"0"
  },
```

```
    "AL22": {
      "description": "AL22",
      "alarmbin": "0",
      "alarm": "",
      "assign":"0"
    },
    "AL23": {
      "description": "AL23",
      "alarmbin": "0",
      "alarm": "",
      "assign":"0"
    },
    "AL24": {
      "description": "AL24",
      "alarmbin": "0",
      "alarm": "",
      "assign":"0"
    }
  },
  "R": {
    "R1": {
      "description": "Relay 1",
      "value": "OFF",
      "valuebin": "0",
      "pulseWidth": "0.3",
      "control": "Any Alarm"
    },
    "R2": {
      "description": "Relay 2",
      "value": "OFF",
      "valuebin": "0",
      "pulseWidth": "1.0",
      "control": "Watchdog1"
    },
    "R3": {
      "description": "Relay 3",
      "value": "OFF",
      "valuebin": "0",
      "pulseWidth": "1.0",
      "control": "Watchdog3"
    },
    "R4": {
      "description": "Relay 4",
      "value": "OFF",
      "valuebin": "0",
      "pulseWidth": "1.0",
      "control": "Watchdog4"
    }
  },
  "FUNC": {
    "FUNC1": {
      "alarmbin": "0",
      "alarm": "Normal"
    },
    "FUNC2": {
      "alarmbin": "0",
      "alarm": "Normal"
    },
    "FUNC3": {
      "alarmbin": "0",
      "alarm": "Normal"
    },
    "FUNC4": {
      "alarmbin": "0",
      "alarm": "Normal"
    }
  },
  "WDG": {
    "WDG1": {
      "description": "Watchdog1",
      "modebin": "3",
      "mode": "incoming ping",
      "host": "0.0.0.0",
      "state": "0"
    },
    "WDG2": {
      "description": "Watchdog2",
```

```
            "modebin": "2",
            "mode": "outgoing ping",
            "host": "192.168.32.112",
            "state": "0"
          },
          "WDG3": {
            "description": "Watchdog3",
            "modebin": "2",
            "mode": "outgoing ping",
            "host": "192.168.32.2",
            "state": "0"
          },
          "WDG4": {
            "description": "Watchdog4",
            "modebin": "2",
            "mode": "outgoing ping",
            "host": "192.168.32.30",
            "state": "0"
          }
        },
        "HTTPPost": {
          "Key": "",
          "PostPeriod": "60"
        },
        "MQTT": {
          "Period": "300"
        },
        "Sys": {
          "hwerr": "",
          "HighAlarmbin": "1",
          "HighAlarm": "Normal"
        },
        "Time": {
          "Date": "04.01.2024",
          "Time": "16:36:26"
        }
      }
}
```

MODBUS TCP/IP full address table

Parameter	FC	PDU decimal address	Data size	Data
Relay 1	01,05,15	100	Discrete	
Relay 2	01,05,15	101	Discrete	
Relay 3	01,05,15	102	Discrete	
Relay 4	01,05,15	103	Discrete	
Digital input 1	02	100	Discrete	
Digital input 2	02	101	Discrete	
Digital input 3	02	102	Discrete	
Digital input 4	02	103	Discrete	
Relay 1 description	03,16	10000	16 bytes UTF-8	
Relay 2 description	03,16	10008	16 bytes UTF-8	
Relay 3 description	03,16	10016	16 bytes UTF-8	
Relay 4 description	03,16	10024	16 bytes UTF-8	
Relay 1 pulse width	03,16	10100	32-bit Float	
Relay 2 pulse width	03,16	10102	32-bit Float	
Relay 3 pulse width	03,16	10104	32-bit Float	
Relay 4 pulse width	03,16	10106	32-bit Float	
Relay 1 activated from	03,06,16	10200	16-bit unsign int	webHttpApi(0), al01(1), al02(2), al03(3), al04(4), al05(5), al06(6), al07(7), al08(8), al09(9), al10(10), al11(11), al12(12), al13(13), al14(14), al15(15), al16(16), al17(17), al18(18), al19(19), al20(20), al21(21), al22(22), al23(23), al24(24), anyAlarm(25), schedule1(26), schedule2(27), schedule3(28), schedule4(29)
Relay 2 activated from	03,06,16	10201	16-bit unsign int	—"—
Relay 3 activated from	03,06,16	10202	16-bit unsign int	—"—
Relay 4 activated from	03,06,16	10203	16-bit unsign int	—"—
Relay 1 action on alarm	03,06,16	10300	16-bit unsign int	on(0), pulse(2)
Relay 2 action on alarm	03,06,16	10301	16-bit unsign int	on(0), pulse(2)
Relay 3 action on alarm	03,06,16	10302	16-bit unsign int	on(0), pulse(2)
Relay 4 action on alarm	03,06,16	10303	16-bit unsign int	on(0), pulse(2)
Relays state after restart	03,06	10400	16-bit unsign int	off(0), on(1), laststate(2)

Digital input 1 Description	03,16	12000	16 bytes UTF-8	
Digital input 2 Description	03,16	12008	16 bytes UTF-8	
Digital input 3 Description	03,16	12016	16 bytes UTF-8	
Digital input 4 Description	03,16	12024	16 bytes UTF-8	
Digital input 1 closed state description	03,16	12100	16 bytes UTF-8	
Digital input 2 closed state description	03,16	12108	16 bytes UTF-8	
Digital input 3 closed state description	03,16	12116	16 bytes UTF-8	
Digital input 4 closed state description	03,16	12124	16 bytes UTF-8	
Digital input 1 open state description	03,16	12200	16 bytes UTF-8	
Digital input 2 open state description	03,16	12208	16 bytes UTF-8	
Digital input 3 open state description	03,16	12216	16 bytes UTF-8	
Digital input 4 open state description	03,16	12224	16 bytes UTF-8	
Digital input 1 mode	03,06,16	12300	16-bit unsign int	openclosed(0), risingEdge(1), fallingEdge(2), bothEdges(3)
Digital input 2 mode	03,06,16	12301	16-bit unsign int	—"—
Digital input 3 mode	03,06,16	12302	16-bit unsign int	—"—
Digital input 4 mode	03,06,16	12303	16-bit unsign int	—"—
Digital input 1 close to open delay	03,16	12400	32-bit unsign int	
Digital input 2 close to open delay	03,16	12402	32-bit unsign int	
Digital input 3 close to open delay	03,16	12404	32-bit unsign int	
Digital input 4 close to open delay	03,16	12406	32-bit unsign int	
Digital input 1 open to close delay	03,16	12500	32-bit unsign int	
Digital input 2 open to close delay	03,16	12502	32-bit unsign int	
Digital input 3 open to close delay	03,16	12504	32-bit unsign int	
Digital input 4 open to close delay	03,16	12506	32-bit unsign int	
Digital input 1 counter init value	03,16	12600	32-bit unsign int	
Digital input 2 counter init value	03,16	12602	32-bit unsign int	
Digital input 3 counter init value	03,16	12604	32-bit unsign int	

Digital input 4 counter init value	03,16	12606	32-bit unsign int	
Digital input 1 counter value	03	12700	32-bit unsign int	
Digital input 2 counter value	03	12702	32-bit unsign int	
Digital input 3 counter value	03	12704	32-bit unsign int	
Digital input 4 counter value	03	12706	32-bit unsign int	
Analog input 1 description	03,16	14000	16 bytes UTF-8	
Analog input 2 description	03,16	14008	16 bytes UTF-8	
Analog input 3 description	03,16	14016	16 bytes UTF-8	
Analog input 4 description	03,16	14024	16 bytes UTF-8	
Analog input 1 multiplier	03,16	14100	32-bit Float	
Analog input 2 multiplier	03,16	14102	32-bit Float	
Analog input 3 multiplier	03,16	14104	32-bit Float	
Analog input 4 multiplier	03,16	14106	32-bit Float	
Analog input 1 offset	03,16	14200	32-bit Float	
Analog input 2 offset	03,16	14202	32-bit Float	
Analog input 3 offset	03,16	14204	32-bit Float	
Analog input 4 offset	03,16	14206	32-bit Float	
Analog input 1 mode(V/mA)	03,06,16	14300	16-bit unsign int	0-10V(0), 4-20mA(1)
Analog input 2 mode(V/mA)	03,06,16	14301	16-bit unsign int	0-10V(0), 4-20mA(1)
Analog input 3 mode(V/mA)	03,06,16	14302	16-bit unsign int	0-10V(0), 4-20mA(1)
Analog input 4 mode(V/mA)	03,06,16	14303	16-bit unsign int	0-10V(0), 4-20mA(1)
Analog input 1 value	03	14400	32-bit Float	
Analog input 2 value	03	14402	32-bit Float	
Analog input 3 value	03	14404	32-bit Float	
Analog input 4 value	03	14406	32-bit Float	
MB sensor 1 description	03,16	16000	16 bytes UTF-8	
MB sensor 2 description	03,16	16008	16 bytes UTF-8	
MB sensor 3 description	03,16	16016	16 bytes UTF-8	
MB sensor 4 description	03,16	16024	16 bytes UTF-8	
MB sensor 5 description	03,16	16032	16 bytes UTF-8	
MB sensor 6 description	03,16	16040	16 bytes UTF-8	
MB sensor 7 description	03,16	16048	16 bytes UTF-8	
MB sensor 8 description	03,16	16056	16 bytes UTF-8	
MB sensor 9 description	03,16	16064	16 bytes UTF-8	
MB sensor 10 description	03,16	16072	16 bytes UTF-8	
MB sensor 11 description	03,16	16080	16 bytes UTF-8	
MB sensor 12 description	03,16	16088	16 bytes UTF-8	
MB sensor 13 description	03,16	16096	16 bytes UTF-8	
MB sensor 14 description	03,16	16104	16 bytes UTF-8	
MB sensor 15 description	03,16	16112	16 bytes UTF-8	
MB sensor 16 description	03,16	16120	16 bytes UTF-8	

MB sensor 17 description	03,16	16128	16 bytes UTF-8	
MB sensor 18 description	03,16	16136	16 bytes UTF-8	
MB sensor 19 description	03,16	16144	16 bytes UTF-8	
MB sensor 20 description	03,16	16152	16 bytes UTF-8	
MB sensor 21 description	03,16	16160	16 bytes UTF-8	
MB sensor 22 description	03,16	16168	16 bytes UTF-8	
MB sensor 23 description	03,16	16176	16 bytes UTF-8	
MB sensor 24 description	03,16	16184	16 bytes UTF-8	
MB sensor 1 multiplier	03,16	16200	32-bit Float	
MB sensor 2 multiplier	03,16	16202	32-bit Float	
MB sensor 3 multiplier	03,16	16204	32-bit Float	
MB sensor 4 multiplier	03,16	16206	32-bit Float	
MB sensor 5 multiplier	03,16	16208	32-bit Float	
MB sensor 6 multiplier	03,16	16210	32-bit Float	
MB sensor 7 multiplier	03,16	16212	32-bit Float	
MB sensor 8 multiplier	03,16	16214	32-bit Float	
MB sensor 9 multiplier	03,16	16216	32-bit Float	
MB sensor 10 multiplier	03,16	16218	32-bit Float	
MB sensor 11 multiplier	03,16	16220	32-bit Float	
MB sensor 12 multiplier	03,16	16222	32-bit Float	
MB sensor 13 multiplier	03,16	16224	32-bit Float	
MB sensor 14 multiplier	03,16	16226	32-bit Float	
MB sensor 15 multiplier	03,16	16228	32-bit Float	
MB sensor 16 multiplier	03,16	16230	32-bit Float	
MB sensor 17 multiplier	03,16	16232	32-bit Float	
MB sensor 18 multiplier	03,16	16234	32-bit Float	
MB sensor 19 multiplier	03,16	16236	32-bit Float	
MB sensor 20 multiplier	03,16	16238	32-bit Float	
MB sensor 21 multiplier	03,16	16240	32-bit Float	
MB sensor 22 multiplier	03,16	16242	32-bit Float	
MB sensor 23 multiplier	03,16	16244	32-bit Float	
MB sensor 24 multiplier	03,16	16246	32-bit Float	
MB sensor 1 offset	03,16	16300	32-bit Float	
MB sensor 2 offset	03,16	16302	32-bit Float	
MB sensor 3 offset	03,16	16304	32-bit Float	
MB sensor 4 offset	03,16	16306	32-bit Float	
MB sensor 5 offset	03,16	16308	32-bit Float	
MB sensor 6 offset	03,16	16310	32-bit Float	
MB sensor 7 offset	03,16	16312	32-bit Float	
MB sensor 8 offset	03,16	16314	32-bit Float	
MB sensor 9 offset	03,16	16316	32-bit Float	
MB sensor 10 offset	03,16	16318	32-bit Float	
MB sensor 11 offset	03,16	16320	32-bit Float	
MB sensor 12 offset	03,16	16322	32-bit Float	
MB sensor 13 offset	03,16	16324	32-bit Float	
MB sensor 14 offset	03,16	16326	32-bit Float	

MB sensor 15 offset	03,16	16328	32-bit Float	
MB sensor 16 offset	03,16	16330	32-bit Float	
MB sensor 17 offset	03,16	16332	32-bit Float	
MB sensor 18 offset	03,16	16334	32-bit Float	
MB sensor 19 offset	03,16	16336	32-bit Float	
MB sensor 20 offset	03,16	16338	32-bit Float	
MB sensor 21 offset	03,16	16340	32-bit Float	
MB sensor 22 offset	03,16	16342	32-bit Float	
MB sensor 23 offset	03,16	16344	32-bit Float	
MB sensor 24 offset	03,16	16346	32-bit Float	
MB sensor 1 value	03	16400	32-bit Float	
MB sensor 2 value	03	16402	32-bit Float	
MB sensor 3 value	03	16404	32-bit Float	
MB sensor 4 value	03	16406	32-bit Float	
MB sensor 5 value	03	16408	32-bit Float	
MB sensor 6 value	03	16410	32-bit Float	
MB sensor 7 value	03	16412	32-bit Float	
MB sensor 8 value	03	16414	32-bit Float	
MB sensor 9 value	03	16416	32-bit Float	
MB sensor 10 value	03	16418	32-bit Float	
MB sensor 11 value	03	16420	32-bit Float	
MB sensor 12 value	03	16422	32-bit Float	
MB sensor 13 value	03	16424	32-bit Float	
MB sensor 14 value	03	16426	32-bit Float	
MB sensor 15 value	03	16428	32-bit Float	
MB sensor 16 value	03	16430	32-bit Float	
MB sensor 17 value	03	16432	32-bit Float	
MB sensor 18 value	03	16434	32-bit Float	
MB sensor 19 value	03	16436	32-bit Float	
MB sensor 20 value	03	16438	32-bit Float	
MB sensor 21 value	03	16440	32-bit Float	
MB sensor 22 value	03	16442	32-bit Float	
MB sensor 23 value	03	16444	32-bit Float	
MB sensor 24 value	03	16446	32-bit Float	
MB sensor 1 counter	03	16500	32-bit unsign int	
MB sensor 2 counter	03	16502	32-bit unsign int	
MB sensor 3 counter	03	16504	32-bit unsign int	
MB sensor 4 counter	03	16506	32-bit unsign int	
MB sensor 5 counter	03	16508	32-bit unsign int	
MB sensor 6 counter	03	16510	32-bit unsign int	
MB sensor 7 counter	03	16512	32-bit unsign int	
MB sensor 8 counter	03	16514	32-bit unsign int	
MB sensor 9 counter	03	16516	32-bit unsign int	
MB sensor 10 counter	03	16518	32-bit unsign int	
MB sensor 11 counter	03	16520	32-bit unsign int	
MB sensor 12 counter	03	16522	32-bit unsign int	

MB sensor 13 counter	03	16524	32-bit unsign int	
MB sensor 14 counter	03	16526	32-bit unsign int	
MB sensor 15 counter	03	16528	32-bit unsign int	
MB sensor 16 counter	03	16530	32-bit unsign int	
MB sensor 17 counter	03	16532	32-bit unsign int	
MB sensor 18 counter	03	16534	32-bit unsign int	
MB sensor 19 counter	03	16536	32-bit unsign int	
MB sensor 20 counter	03	16538	32-bit unsign int	
MB sensor 21 counter	03	16540	32-bit unsign int	
MB sensor 22 counter	03	16542	32-bit unsign int	
MB sensor 23 counter	03	16544	32-bit unsign int	
MB sensor 24 counter	03	16546	32-bit unsign int	
Channel 1 type	03,06,16	18000	16-bit unsign int	general(0), discrete(2) , counter(3)
Channel 2 type	03,06,16	18001	16-bit unsign int	—"—
Channel 3 type	03,06,16	18002	16-bit unsign int	—"—
Channel 4 type	03,06,16	18003	16-bit unsign int	—"—
Channel 5 type	03,06,16	18004	16-bit unsign int	—"—
Channel 6 type	03,06,16	18005	16-bit unsign int	—"—
Channel 7 type	03,06,16	18006	16-bit unsign int	—"—
Channel 8 type	03,06,16	18007	16-bit unsign int	—"—
Channel 9 type	03,06,16	18008	16-bit unsign int	—"—
Channel 10 type	03,06,16	18009	16-bit unsign int	—"—
Channel 11 type	03,06,16	18010	16-bit unsign int	—"—
Channel 12 type	03,06,16	18011	16-bit unsign int	—"—
Channel 13 type	03,06,16	18012	16-bit unsign int	—"—
Channel 14 type	03,06,16	18013	16-bit unsign int	—"—
Channel 15 type	03,06,16	18014	16-bit unsign int	—"—
Channel 16 type	03,06,16	18015	16-bit unsign int	—"—
Channel 17 type	03,06,16	18016	16-bit unsign int	—"—
Channel 18 type	03,06,16	18017	16-bit unsign int	—"—
Channel 19 type	03,06,16	18018	16-bit unsign int	—"—
Channel 20 type	03,06,16	18019	16-bit unsign int	—"—
Channel 21 type	03,06,16	18020	16-bit unsign int	—"—
Channel 22 type	03,06,16	18021	16-bit unsign int	—"—
Channel 23 type	03,06,16	18022	16-bit unsign int	—"—
Channel 24 type	03,06,16	18023	16-bit unsign int	—"—
Channel 1 description	03,16	18100	16 bytes UTF-8	
Channel 2 description	03,16	18108	16 bytes UTF-8	
Channel 3 description	03,16	18116	16 bytes UTF-8	
Channel 4 description	03,16	18124	16 bytes UTF-8	
Channel 5 description	03,16	18132	16 bytes UTF-8	
Channel 6 description	03,16	18140	16 bytes UTF-8	
Channel 7 description	03,16	18148	16 bytes UTF-8	
Channel 8 description	03,16	18156	16 bytes UTF-8	
Channel 9 description	03,16	18164	16 bytes UTF-8	
Channel 10 description	03,16	18172	16 bytes UTF-8	

Channel 11 description	03,16	18180	16 bytes UTF-8	
Channel 12 description	03,16	18188	16 bytes UTF-8	
Channel 13 description	03,16	18196	16 bytes UTF-8	
Channel 14 description	03,16	18204	16 bytes UTF-8	
Channel 15 description	03,16	18212	16 bytes UTF-8	
Channel 16 description	03,16	18220	16 bytes UTF-8	
Channel 17 description	03,16	18228	16 bytes UTF-8	
Channel 18 description	03,16	18236	16 bytes UTF-8	
Channel 19 description	03,16	18244	16 bytes UTF-8	
Channel 20 description	03,16	18252	16 bytes UTF-8	
Channel 21 description	03,16	18260	16 bytes UTF-8	
Channel 22 description	03,16	18268	16 bytes UTF-8	
Channel 23 description	03,16	18276	16 bytes UTF-8	
Channel 24 description	03,16	18284	16 bytes UTF-8	
Channel 1 parameter 1	03,06,16	18300	16-bit unsign int	none(0), s01(3), s02(4), s03(5), s04(6), s05(7), s06(8), s07(9), s08(10), s09(11), s10(12), s11(13), s12(14), s13(15), s14(16), s15(17), s16(18), s17(19), s18(20), s19(21), s20(22), s21(23), s22(24), s23(25), s24(26), a01(27), a02(28), a03(29), a04(30), d01(31), d02(32), d03(33), d04(34)
Channel 2 parameter 1	03,06,16	18301	16-bit unsign int	—"—
Channel 3 parameter 1	03,06,16	18302	16-bit unsign int	—"—
Channel 4 parameter 1	03,06,16	18303	16-bit unsign int	—"—
Channel 5 parameter 1	03,06,16	18304	16-bit unsign int	—"—
Channel 6 parameter 1	03,06,16	18305	16-bit unsign int	—"—
Channel 7 parameter 1	03,06,16	18306	16-bit unsign int	—"—
Channel 8 parameter 1	03,06,16	18307	16-bit unsign int	—"—
Channel 9 parameter 1	03,06,16	18308	16-bit unsign int	—"—
Channel 10 parameter 1	03,06,16	18309	16-bit unsign int	—"—
Channel 11 parameter 1	03,06,16	18310	16-bit unsign int	—"—
Channel 12 parameter 1	03,06,16	18311	16-bit unsign int	—"—
Channel 13 parameter 1	03,06,16	18312	16-bit unsign int	—"—
Channel 14 parameter 1	03,06,16	18313	16-bit unsign int	—"—
Channel 15 parameter 1	03,06,16	18314	16-bit unsign int	—"—
Channel 16 parameter 1	03,06,16	18315	16-bit unsign int	—"—
Channel 17 parameter 1	03,06,16	18316	16-bit unsign int	—"—
Channel 18 parameter 1	03,06,16	18317	16-bit unsign int	—"—
Channel 19 parameter 1	03,06,16	18318	16-bit unsign int	—"—
Channel 20 parameter 1	03,06,16	18319	16-bit unsign int	—"—
Channel 21 parameter 1	03,06,16	18320	16-bit unsign int	—"—
Channel 22 parameter 1	03,06,16	18321	16-bit unsign int	—"—
Channel 23 parameter 1	03,06,16	18322	16-bit unsign int	—"—
Channel 24 parameter 1	03,06,16	18323	16-bit unsign int	—"—

Channel 1 op 1	03,06,16	18400	16-bit unsign int	none(0), multiplication(1), division(2), sum(3), subtract(4)
Channel 2 op 1	03,06,16	18401	16-bit unsign int	—"—
Channel 3 op 1	03,06,16	18402	16-bit unsign int	—"—
Channel 4 op 1	03,06,16	18403	16-bit unsign int	—"—
Channel 5 op 1	03,06,16	18404	16-bit unsign int	—"—
Channel 6 op 1	03,06,16	18405	16-bit unsign int	—"—
Channel 7 op 1	03,06,16	18406	16-bit unsign int	—"—
Channel 8 op 1	03,06,16	18407	16-bit unsign int	—"—
Channel 9 op 1	03,06,16	18408	16-bit unsign int	—"—
Channel 10 op 1	03,06,16	18409	16-bit unsign int	—"—
Channel 11 op 1	03,06,16	18410	16-bit unsign int	—"—
Channel 12 op 1	03,06,16	18411	16-bit unsign int	—"—
Channel 13 op 1	03,06,16	18412	16-bit unsign int	—"—
Channel 14 op 1	03,06,16	18413	16-bit unsign int	—"—
Channel 15 op 1	03,06,16	18414	16-bit unsign int	—"—
Channel 16 op 1	03,06,16	18415	16-bit unsign int	—"—
Channel 17 op 1	03,06,16	18416	16-bit unsign int	—"—
Channel 18 op 1	03,06,16	18417	16-bit unsign int	—"—
Channel 19 op 1	03,06,16	18418	16-bit unsign int	—"—
Channel 20 op 1	03,06,16	18419	16-bit unsign int	—"—
Channel 21 op 1	03,06,16	18420	16-bit unsign int	—"—
Channel 22 op 1	03,06,16	18421	16-bit unsign int	—"—
Channel 23 op 1	03,06,16	18422	16-bit unsign int	—"—
Channel 24 op 1	03,06,16	18423	16-bit unsign int	—"—
Channel 1 parameter 2	03,06,16	18500	16-bit unsign int	none(0), s01(3), s02(4), s03(5), s04(6), s05(7), s06(8), s07(9), s08(10), s09(11), s10(12), s11(13), s12(14), s13(15), s14(16), s15(17), s16(18), s17(19), s18(20), s19(21), s20(22), s21(23), s22(24), s23(25), s24(26), a01(27), a02(28), a03(29), a04(30), d01(31), d02(32), d03(33), d04(34)
Channel 2 parameter 2	03,06,16	18501	16-bit unsign int	—"—
Channel 3 parameter 2	03,06,16	18502	16-bit unsign int	—"—
Channel 4 parameter 2	03,06,16	18503	16-bit unsign int	—"—
Channel 5 parameter 2	03,06,16	18504	16-bit unsign int	—"—
Channel 6 parameter 2	03,06,16	18505	16-bit unsign int	—"—
Channel 7 parameter 2	03,06,16	18506	16-bit unsign int	—"—
Channel 8 parameter 2	03,06,16	18507	16-bit unsign int	—"—
Channel 9 parameter 2	03,06,16	18508	16-bit unsign int	—"—
Channel 10 parameter 2	03,06,16	18509	16-bit unsign int	—"—
Channel 11 parameter 2	03,06,16	18510	16-bit unsign int	—"—
Channel 12 parameter 2	03,06,16	18511	16-bit unsign int	—"—
Channel 13 parameter 2	03,06,16	18512	16-bit unsign int	—"—

Channel 14 parameter 2	03,06,16	18513	16-bit unsign int	—"—
Channel 15 parameter 2	03,06,16	18514	16-bit unsign int	—"—
Channel 16 parameter 2	03,06,16	18515	16-bit unsign int	—"—
Channel 17 parameter 2	03,06,16	18516	16-bit unsign int	—"—
Channel 18 parameter 2	03,06,16	18517	16-bit unsign int	—"—
Channel 19 parameter 2	03,06,16	18518	16-bit unsign int	—"—
Channel 20 parameter 2	03,06,16	18519	16-bit unsign int	—"—
Channel 21 parameter 2	03,06,16	18520	16-bit unsign int	—"—
Channel 22 parameter 2	03,06,16	18521	16-bit unsign int	—"—
Channel 23 parameter 2	03,06,16	18522	16-bit unsign int	—"—
Channel 24 parameter 2	03,06,16	18523	16-bit unsign int	—"—
Channel 1 op 2	03,06,16	18600	16-bit unsign int	none(0), multiplication(1), division(2), sum(3), subtract(4)
Channel 2 op 2	03,06,16	18601	16-bit unsign int	—"—
Channel 3 op 2	03,06,16	18602	16-bit unsign int	—"—
Channel 4 op 2	03,06,16	18603	16-bit unsign int	—"—
Channel 5 op 2	03,06,16	18604	16-bit unsign int	—"—
Channel 6 op 2	03,06,16	18605	16-bit unsign int	—"—
Channel 7 op 2	03,06,16	18606	16-bit unsign int	—"—
Channel 8 op 2	03,06,16	18607	16-bit unsign int	—"—
Channel 9 op 2	03,06,16	18608	16-bit unsign int	—"—
Channel 10 op 2	03,06,16	18609	16-bit unsign int	—"—
Channel 11 op 2	03,06,16	18610	16-bit unsign int	—"—
Channel 12 op 2	03,06,16	18611	16-bit unsign int	—"—
Channel 13 op 2	03,06,16	18612	16-bit unsign int	—"—
Channel 14 op 2	03,06,16	18613	16-bit unsign int	—"—
Channel 15 op 2	03,06,16	18614	16-bit unsign int	—"—
Channel 16 op 2	03,06,16	18615	16-bit unsign int	—"—
Channel 17 op 2	03,06,16	18616	16-bit unsign int	—"—
Channel 18 op 2	03,06,16	18617	16-bit unsign int	—"—
Channel 19 op 2	03,06,16	18618	16-bit unsign int	—"—
Channel 20 op 2	03,06,16	18619	16-bit unsign int	—"—
Channel 21 op 2	03,06,16	18620	16-bit unsign int	—"—
Channel 22 op 2	03,06,16	18621	16-bit unsign int	—"—
Channel 23 op 2	03,06,16	18622	16-bit unsign int	—"—
Channel 24 op 2	03,06,16	18623	16-bit unsign int	—"—
Channel 1 coeff 1	03,16	18700	32-bit Float	
Channel 2 coeff 1	03,16	18702	32-bit Float	
Channel 3 coeff 1	03,16	18704	32-bit Float	
Channel 4 coeff 1	03,16	18706	32-bit Float	
Channel 5 coeff 1	03,16	18708	32-bit Float	
Channel 6 coeff 1	03,16	18710	32-bit Float	
Channel 7 coeff 1	03,16	18712	32-bit Float	
Channel 8 coeff 1	03,16	18714	32-bit Float	
Channel 9 coeff 1	03,16	18716	32-bit Float	
Channel 10 coeff 1	03,16	18718	32-bit Float	

Channel 11 coeff 1	03,16	18720	32-bit Float	
Channel 12 coeff 1	03,16	18722	32-bit Float	
Channel 13 coeff 1	03,16	18724	32-bit Float	
Channel 14 coeff 1	03,16	18726	32-bit Float	
Channel 15 coeff 1	03,16	18728	32-bit Float	
Channel 16 coeff 1	03,16	18730	32-bit Float	
Channel 17 coeff 1	03,16	18732	32-bit Float	
Channel 18 coeff 1	03,16	18734	32-bit Float	
Channel 19 coeff 1	03,16	18736	32-bit Float	
Channel 20 coeff 1	03,16	18738	32-bit Float	
Channel 21 coeff 1	03,16	18740	32-bit Float	
Channel 22 coeff 1	03,16	18742	32-bit Float	
Channel 23 coeff 1	03,16	18744	32-bit Float	
Channel 24 coeff 1	03,16	18746	32-bit Float	
Channel 1 op 3	03,06,16	18800	16-bit unsign int	none(0), multiplication(1), division(2), sum(3), subtract(4)
Channel 2 op 3	03,06,16	18801	16-bit unsign int	—"—
Channel 3 op 3	03,06,16	18802	16-bit unsign int	—"—
Channel 4 op 3	03,06,16	18803	16-bit unsign int	—"—
Channel 5 op 3	03,06,16	18804	16-bit unsign int	—"—
Channel 6 op 3	03,06,16	18805	16-bit unsign int	—"—
Channel 7 op 3	03,06,16	18806	16-bit unsign int	—"—
Channel 8 op 3	03,06,16	18807	16-bit unsign int	—"—
Channel 9 op 3	03,06,16	18808	16-bit unsign int	—"—
Channel 10 op 3	03,06,16	18809	16-bit unsign int	—"—
Channel 11 op 3	03,06,16	18810	16-bit unsign int	—"—
Channel 12 op 3	03,06,16	18811	16-bit unsign int	—"—
Channel 13 op 3	03,06,16	18812	16-bit unsign int	—"—
Channel 14 op 3	03,06,16	18813	16-bit unsign int	—"—
Channel 15 op 3	03,06,16	18814	16-bit unsign int	—"—
Channel 16 op 3	03,06,16	18815	16-bit unsign int	—"—
Channel 17 op 3	03,06,16	18816	16-bit unsign int	—"—
Channel 18 op 3	03,06,16	18817	16-bit unsign int	—"—
Channel 19 op 3	03,06,16	18818	16-bit unsign int	—"—
Channel 20 op 3	03,06,16	18819	16-bit unsign int	—"—
Channel 21 op 3	03,06,16	18820	16-bit unsign int	—"—
Channel 22 op 3	03,06,16	18821	16-bit unsign int	—"—
Channel 23 op 3	03,06,16	18822	16-bit unsign int	—"—
Channel 24 op 3	03,06,16	18823	16-bit unsign int	—"—
Channel 1 coeff 2	03,16	18900	32-bit Float	
Channel 2 coeff 2	03,16	18902	32-bit Float	
Channel 3 coeff 2	03,16	18904	32-bit Float	
Channel 4 coeff 2	03,16	18906	32-bit Float	
Channel 5 coeff 2	03,16	18908	32-bit Float	
Channel 6 coeff 2	03,16	18910	32-bit Float	
Channel 7 coeff 2	03,16	18912	32-bit Float	

Channel 8 coeff 2	03,16	18914	32-bit Float	
Channel 9 coeff 2	03,16	18916	32-bit Float	
Channel 10 coeff 2	03,16	18918	32-bit Float	
Channel 11 coeff 2	03,16	18920	32-bit Float	
Channel 12 coeff 2	03,16	18922	32-bit Float	
Channel 13 coeff 2	03,16	18924	32-bit Float	
Channel 14 coeff 2	03,16	18926	32-bit Float	
Channel 15 coeff 2	03,16	18928	32-bit Float	
Channel 16 coeff 2	03,16	18930	32-bit Float	
Channel 17 coeff 2	03,16	18932	32-bit Float	
Channel 18 coeff 2	03,16	18934	32-bit Float	
Channel 29 coeff 2	03,16	18936	32-bit Float	
Channel 20 coeff 2	03,16	18938	32-bit Float	
Channel 21 coeff 2	03,16	18940	32-bit Float	
Channel 22 coeff 2	03,16	18942	32-bit Float	
Channel 23 coeff 2	03,16	18944	32-bit Float	
Channel 24 coeff 2	03,16	18946	32-bit Float	
Channel 1 unit	03,16	19000	16 bytes UTF-8	
Channel 2 unit	03,16	19008	16 bytes UTF-8	
Channel 3 unit	03,16	19016	16 bytes UTF-8	
Channel 4 unit	03,16	19024	16 bytes UTF-8	
Channel 5 unit	03,16	19032	16 bytes UTF-8	
Channel 6 unit	03,16	19040	16 bytes UTF-8	
Channel 7 unit	03,16	19048	16 bytes UTF-8	
Channel 8 unit	03,16	19056	16 bytes UTF-8	
Channel 9 unit	03,16	19064	16 bytes UTF-8	
Channel 10 unit	03,16	19072	16 bytes UTF-8	
Channel 11 unit	03,16	19080	16 bytes UTF-8	
Channel 12 unit	03,16	19088	16 bytes UTF-8	
Channel 13 unit	03,16	19096	16 bytes UTF-8	
Channel 14 unit	03,16	19104	16 bytes UTF-8	
Channel 15 unit	03,16	19112	16 bytes UTF-8	
Channel 16 unit	03,16	19120	16 bytes UTF-8	
Channel 17 unit	03,16	19128	16 bytes UTF-8	
Channel 18 unit	03,16	19136	16 bytes UTF-8	
Channel 19 unit	03,16	19144	16 bytes UTF-8	
Channel 20 unit	03,16	19152	16 bytes UTF-8	
Channel 21 unit	03,16	19160	16 bytes UTF-8	
Channel 22 unit	03,16	19168	16 bytes UTF-8	
Channel 23 unit	03,16	19176	16 bytes UTF-8	
Channel 24 unit	03,16	19184	16 bytes UTF-8	
Channel 1 value	03	19300	32-bit Float	
Channel 2 value	03	19302	32-bit Float	
Channel 3 value	03	19304	32-bit Float	
Channel 4 value	03	19306	32-bit Float	
Channel 5 value	03	19308	32-bit Float	

Channel 6 value	03	19310	32-bit Float	
Channel 7 value	03	19312	32-bit Float	
Channel 8 value	03	19314	32-bit Float	
Channel 9 value	03	19316	32-bit Float	
Channel 10 value	03	19318	32-bit Float	
Channel 11 value	03	19320	32-bit Float	
Channel 12 value	03	19322	32-bit Float	
Channel 13 value	03	19324	32-bit Float	
Channel 14 value	03	19326	32-bit Float	
Channel 15 value	03	19328	32-bit Float	
Channel 16 value	03	19330	32-bit Float	
Channel 17 value	03	19332	32-bit Float	
Channel 18 value	03	19334	32-bit Float	
Channel 19 value	03	19336	32-bit Float	
Channel 20 value	03	19338	32-bit Float	
Channel 21 value	03	19340	32-bit Float	
Channel 22 value	03	19342	32-bit Float	
Channel 23 value	03	19344	32-bit Float	
Channel 24 value	03	19346	32-bit Float	
Channel 1 counter	03	19400	32-bit unsign int	
Channel 2 counter	03	19402	32-bit unsign int	
Channel 3 counter	03	19404	32-bit unsign int	
Channel 4 counter	03	19406	32-bit unsign int	
Channel 5 counter	03	19408	32-bit unsign int	
Channel 6 counter	03	19410	32-bit unsign int	
Channel 7 counter	03	19412	32-bit unsign int	
Channel 8 counter	03	19414	32-bit unsign int	
Channel 9 counter	03	19416	32-bit unsign int	
Channel 10 counter	03	19418	32-bit unsign int	
Channel 11 counter	03	19420	32-bit unsign int	
Channel 12 counter	03	19422	32-bit unsign int	
Channel 13 counter	03	19424	32-bit unsign int	
Channel 14 counter	03	19426	32-bit unsign int	
Channel 15 counter	03	19428	32-bit unsign int	
Channel 16 counter	03	19430	32-bit unsign int	
Channel 17 counter	03	19432	32-bit unsign int	
Channel 18 counter	03	19434	32-bit unsign int	
Channel 19 counter	03	19436	32-bit unsign int	
Channel 20 counter	03	19438	32-bit unsign int	
Channel 21 counter	03	19440	32-bit unsign int	
Channel 22 counter	03	19442	32-bit unsign int	
Channel 23 counter	03	19444	32-bit unsign int	
Channel 24 counter	03	19446	32-bit unsign int	
Channel 1 alarm status	03	19500	16-bit unsign int	undefined(0), normal(1), indeterminate(2), warning(3), minor(4), major(5), critical(6)

Channel 2 alarm status	03	19501	16-bit unsign int	—"—
Channel 3 alarm status	03	19502	16-bit unsign int	—"—
Channel 4 alarm status	03	19503	16-bit unsign int	—"—
Channel 5 alarm status	03	19504	16-bit unsign int	—"—
Channel 6 alarm status	03	19505	16-bit unsign int	—"—
Channel 7 alarm status	03	19506	16-bit unsign int	—"—
Channel 8 alarm status	03	19507	16-bit unsign int	—"—
Channel 9 alarm status	03	19508	16-bit unsign int	—"—
Channel 10 alarm status	03	19509	16-bit unsign int	—"—
Channel 11 alarm status	03	19510	16-bit unsign int	—"—
Channel 12 alarm status	03	19511	16-bit unsign int	—"—
Channel 13 alarm status	03	19512	16-bit unsign int	—"—
Channel 14 alarm status	03	19513	16-bit unsign int	—"—
Channel 15 alarm status	03	19514	16-bit unsign int	—"—
Channel 16 alarm status	03	19515	16-bit unsign int	—"—
Channel 17 alarm status	03	19516	16-bit unsign int	—"—
Channel 18 alarm status	03	19517	16-bit unsign int	—"—
Channel 19 alarm status	03	19518	16-bit unsign int	—"—
Channel 20 alarm status	03	19519	16-bit unsign int	—"—
Channel 21 alarm status	03	19520	16-bit unsign int	—"—
Channel 22 alarm status	03	19521	16-bit unsign int	—"—
Channel 23 alarm status	03	19522	16-bit unsign int	—"—
Channel 24 alarm status	03	19523	16-bit unsign int	—"—
Alarm 1 description	03,16	20000	16 bytes UTF-8	
Alarm 2 description	03,16	20008	16 bytes UTF-8	
Alarm 3 description	03,16	20016	16 bytes UTF-8	
Alarm 4 description	03,16	20024	16 bytes UTF-8	
Alarm 5 description	03,16	20032	16 bytes UTF-8	
Alarm 6 description	03,16	20040	16 bytes UTF-8	
Alarm 7 description	03,16	20048	16 bytes UTF-8	
Alarm 8 description	03,16	20056	16 bytes UTF-8	
Alarm 9 description	03,16	20064	16 bytes UTF-8	
Alarm 10 description	03,16	20072	16 bytes UTF-8	
Alarm 11 description	03,16	20080	16 bytes UTF-8	
Alarm 12 description	03,16	20088	16 bytes UTF-8	
Alarm 13 description	03,16	20096	16 bytes UTF-8	
Alarm 14 description	03,16	20104	16 bytes UTF-8	
Alarm 15 description	03,16	20112	16 bytes UTF-8	
Alarm 16 description	03,16	20120	16 bytes UTF-8	
Alarm 17 description	03,16	20128	16 bytes UTF-8	
Alarm 18 description	03,16	20136	16 bytes UTF-8	
Alarm 19 description	03,16	20144	16 bytes UTF-8	
Alarm 20 description	03,16	20152	16 bytes UTF-8	
Alarm 21 description	03,16	20160	16 bytes UTF-8	
Alarm 22 description	03,16	20168	16 bytes UTF-8	
Alarm 23 description	03,16	20176	16 bytes UTF-8	
Alarm 24 description	03,16	20184	16 bytes UTF-8	

Alarm 1 condition 1 channel	03,06,16	20200	16-bit unsign int	none(0), v01(1), v02(2), v03(3), v04(4), v05(5), v06(6), v07(7), v08(8), v09(9), v10(10), v11(11), v12(12), v13(13), v14(14), v15(15), v16(16), v17(17), v18(18), v19(19), v20(20), v21(21), v22(22), v23(23), v24(24)
Alarm 2 condition 1 channel	03,06,16	20201	16-bit unsign int	—"—
Alarm 3 condition 1 channel	03,06,16	20202	16-bit unsign int	—"—
Alarm 4 condition 1 channel	03,06,16	20203	16-bit unsign int	—"—
Alarm 5 condition 1 channel	03,06,16	20204	16-bit unsign int	—"—
Alarm 6 condition 1 channel	03,06,16	20205	16-bit unsign int	—"—
Alarm 7 condition 1 channel	03,06,16	20206	16-bit unsign int	—"—
Alarm 8 condition 1 channel	03,06,16	20207	16-bit unsign int	—"—
Alarm 9 condition 1 channel	03,06,16	20208	16-bit unsign int	—"—
Alarm 10 condition 1 channel	03,06,16	20209	16-bit unsign int	—"—
Alarm 11 condition 1 channel	03,06,16	20210	16-bit unsign int	—"—
Alarm 12 condition 1 channel	03,06,16	20211	16-bit unsign int	—"—
Alarm 13 condition 1 channel	03,06,16	20212	16-bit unsign int	—"—
Alarm 14 condition 1 channel	03,06,16	20213	16-bit unsign int	—"—
Alarm 15 condition 1 channel	03,06,16	20214	16-bit unsign int	—"—
Alarm 16 condition 1 channel	03,06,16	20215	16-bit unsign int	—"—
Alarm 17 condition 1 channel	03,06,16	20216	16-bit unsign int	—"—
Alarm 18 condition 1 channel	03,06,16	20217	16-bit unsign int	—"—
Alarm 19 condition 1 channel	03,06,16	20218	16-bit unsign int	—"—
Alarm 20 condition 1 channel	03,06,16	20219	16-bit unsign int	—"—
Alarm 21 condition 1 channel	03,06,16	20220	16-bit unsign int	—"—
Alarm 22 condition 1 channel	03,06,16	20221	16-bit unsign int	—"—
Alarm 23 condition 1 channel	03,06,16	20222	16-bit unsign int	—"—
Alarm 24 condition 1 channel	03,06,16	20223	16-bit unsign int	—"—
Alarm 1 condition 1 operand	03,06,16	20300	16-bit unsign int	larger(0), less(1)
Alarm 2 condition 1 operand	03,06,16	20301	16-bit unsign int	larger(0), less(1)
Alarm 3 condition 1 operand	03,06,16	20302	16-bit unsign int	larger(0), less(1)
Alarm 4 condition 1 operand	03,06,16	20303	16-bit unsign int	larger(0), less(1)
Alarm 5 condition 1 operand	03,06,16	20304	16-bit unsign int	larger(0), less(1)
Alarm 6 condition 1 operand	03,06,16	20305	16-bit unsign int	larger(0), less(1)
Alarm 7 condition 1 operand	03,06,16	20306	16-bit unsign int	larger(0), less(1)
Alarm 8 condition 1 operand	03,06,16	20307	16-bit unsign int	larger(0), less(1)
Alarm 9 condition 1 operand	03,06,16	20308	16-bit unsign int	larger(0), less(1)
Alarm 10 condition 1 operand	03,06,16	20309	16-bit unsign int	larger(0), less(1)
Alarm 11 condition 1 operand	03,06,16	20310	16-bit unsign int	larger(0), less(1)
Alarm 12 condition 1 operand	03,06,16	20311	16-bit unsign int	larger(0), less(1)
Alarm 13 condition 1 operand	03,06,16	20312	16-bit unsign int	larger(0), less(1)
Alarm 14 condition 1 operand	03,06,16	20313	16-bit unsign int	larger(0), less(1)
Alarm 15 condition 1 operand	03,06,16	20314	16-bit unsign int	larger(0), less(1)
Alarm 16 condition 1 operand	03,06,16	20315	16-bit unsign int	larger(0), less(1)

Alarm 17 condition 1 operand	03,06,16	20316	16-bit unsign int	larger(0), less(1)
Alarm 18 condition 1 operand	03,06,16	20317	16-bit unsign int	larger(0), less(1)
Alarm 19 condition 1 operand	03,06,16	20318	16-bit unsign int	larger(0), less(1)
Alarm 20 condition 1 operand	03,06,16	20319	16-bit unsign int	larger(0), less(1)
Alarm 21 condition 1 operand	03,06,16	20320	16-bit unsign int	larger(0), less(1)
Alarm 22 condition 1 operand	03,06,16	20321	16-bit unsign int	larger(0), less(1)
Alarm 23 condition 1 operand	03,06,16	20322	16-bit unsign int	larger(0), less(1)
Alarm 24 condition 1 operand	03,06,16	20323	16-bit unsign int	larger(0), less(1)
Alarm 1 condition 1 limit	03,16	20400	32-bit Float	
Alarm 2 condition 1 limit	03,16	20402	32-bit Float	
Alarm 3 condition 1 limit	03,16	20404	32-bit Float	
Alarm 4 condition 1 limit	03,16	20406	32-bit Float	
Alarm 5 condition 1 limit	03,16	20408	32-bit Float	
Alarm 6 condition 1 limit	03,16	20410	32-bit Float	
Alarm 7 condition 1 limit	03,16	20412	32-bit Float	
Alarm 8 condition 1 limit	03,16	20414	32-bit Float	
Alarm 9 condition 1 limit	03,16	20416	32-bit Float	
Alarm 10 condition 1 limit	03,16	20418	32-bit Float	
Alarm 11 condition 1 limit	03,16	20420	32-bit Float	
Alarm 12 condition 1 limit	03,16	20422	32-bit Float	
Alarm 13 condition 1 limit	03,16	20424	32-bit Float	
Alarm 14 condition 1 limit	03,16	20426	32-bit Float	
Alarm 15 condition 1 limit	03,16	20428	32-bit Float	
Alarm 16 condition 1 limit	03,16	20430	32-bit Float	
Alarm 17 condition 1 limit	03,16	20432	32-bit Float	
Alarm 18 condition 1 limit	03,16	20434	32-bit Float	
Alarm 19 condition 1 limit	03,16	20436	32-bit Float	
Alarm 20 condition 1 limit	03,16	20438	32-bit Float	
Alarm 21 condition 1 limit	03,16	20440	32-bit Float	
Alarm 22 condition 1 limit	03,16	20442	32-bit Float	
Alarm 23 condition 1 limit	03,16	20444	32-bit Float	
Alarm 23 condition 1 limit	03,16	20446	32-bit Float	
Alarm 1 condition 1 hysteresis	03,16	20500	32-bit Float	
Alarm 2 condition 1 hysteresis	03,16	20502	32-bit Float	
Alarm 3 condition 1 hysteresis	03,16	20504	32-bit Float	
Alarm 4 condition 1 hysteresis	03,16	20506	32-bit Float	
Alarm 5 condition 1 hysteresis	03,16	20508	32-bit Float	
Alarm 6 condition 1 hysteresis	03,16	20510	32-bit Float	
Alarm 7 condition 1 hysteresis	03,16	20512	32-bit Float	
Alarm 8 condition 1 hysteresis	03,16	20514	32-bit Float	
Alarm 9 condition 1 hysteresis	03,16	20516	32-bit Float	
Alarm 10 condition 1 hysteresis	03,16	20518	32-bit Float	
Alarm 11 condition 1 hysteresis	03,16	20520	32-bit Float	
Alarm 12 condition 1 hysteresis	03,16	20522	32-bit Float	

Alarm 13 condition 1 hysteresis	03,16	20524	32-bit Float	
Alarm 14 condition 1 hysteresis	03,16	20526	32-bit Float	
Alarm 15 condition 1 hysteresis	03,16	20528	32-bit Float	
Alarm 16 condition 1 hysteresis	03,16	20530	32-bit Float	
Alarm 17 condition 1 hysteresis	03,16	20532	32-bit Float	
Alarm 18 condition 1 hysteresis	03,16	20534	32-bit Float	
Alarm 19 condition 1 hysteresis	03,16	20536	32-bit Float	
Alarm 20 condition 1 hysteresis	03,16	20538	32-bit Float	
Alarm 21 condition 1 hysteresis	03,16	20540	32-bit Float	
Alarm 22 condition 1 hysteresis	03,16	20542	32-bit Float	
Alarm 23 condition 1 hysteresis	03,16	20544	32-bit Float	
Alarm 24 condition 1 hysteresis	03,16	20546	32-bit Float	
Alarm 1 condition 1 discrete al. state	03,06,16	20600	16-bit unsign int	open(0), closed(1)
Alarm 2 condition 1 discrete al. state	03,06,16	20601	16-bit unsign int	open(0), closed(1)
Alarm 3 condition 1 discrete al. state	03,06,16	20602	16-bit unsign int	open(0), closed(1)
Alarm 4 condition 1 discrete al. state	03,06,16	20603	16-bit unsign int	open(0), closed(1)
Alarm 5 condition 1 discrete al. state	03,06,16	20604	16-bit unsign int	open(0), closed(1)
Alarm 6 condition 1 discrete al. state	03,06,16	20605	16-bit unsign int	open(0), closed(1)
Alarm 7 condition 1 discrete al. state	03,06,16	20606	16-bit unsign int	open(0), closed(1)
Alarm 8 condition 1 discrete al. state	03,06,16	20607	16-bit unsign int	open(0), closed(1)
Alarm 9 condition 1 discrete al. state	03,06,16	20608	16-bit unsign int	open(0), closed(1)
Alarm 10 condition 1 discrete al. state	03,06,16	20609	16-bit unsign int	open(0), closed(1)
Alarm 11 condition 1 discrete al. state	03,06,16	20610	16-bit unsign int	open(0), closed(1)
Alarm 12 condition 1 discrete al. state	03,06,16	20611	16-bit unsign int	open(0), closed(1)
Alarm 13 condition 1 discrete al. state	03,06,16	20612	16-bit unsign int	open(0), closed(1)
Alarm 14 condition 1 discrete al. state	03,06,16	20613	16-bit unsign int	open(0), closed(1)
Alarm 15 condition 1 discrete al. state	03,06,16	20614	16-bit unsign int	open(0), closed(1)

Alarm 16 condition 1 discrete al. state	03,06,16	20615	16-bit unsign int	open(0), closed(1)
Alarm 17 condition 1 discrete al. state	03,06,16	20616	16-bit unsign int	open(0), closed(1)
Alarm 18 condition 1 discrete al. state	03,06,16	20617	16-bit unsign int	open(0), closed(1)
Alarm 19 condition 1 discrete al. state	03,06,16	20618	16-bit unsign int	open(0), closed(1)
Alarm 20 condition 1 discrete al. state	03,06,16	20619	16-bit unsign int	open(0), closed(1)
Alarm 21 condition 1 discrete al. state	03,06,16	20620	16-bit unsign int	open(0), closed(1)
Alarm 22 condition 1 discrete al. state	03,06,16	20621	16-bit unsign int	open(0), closed(1)
Alarm 23 condition 1 discrete al. state	03,06,16	20622	16-bit unsign int	open(0), closed(1)
Alarm 24 condition 1 discrete al. state	03,06,16	20623	16-bit unsign int	open(0), closed(1)
Alarm 1 function	03,06,16	20700	16-bit unsign int	none(0}, and(1), or(2)
Alarm 2 function	03,06,16	20701	16-bit unsign int	none(0}, and(1), or(2)
Alarm 3 function	03,06,16	20702	16-bit unsign int	none(0}, and(1), or(2)
Alarm 4 function	03,06,16	20703	16-bit unsign int	none(0}, and(1), or(2)
Alarm 5 function	03,06,16	20704	16-bit unsign int	none(0}, and(1), or(2)
Alarm 6 function	03,06,16	20705	16-bit unsign int	none(0}, and(1), or(2)
Alarm 7 function	03,06,16	20706	16-bit unsign int	none(0}, and(1), or(2)
Alarm 8 function	03,06,16	20707	16-bit unsign int	none(0}, and(1), or(2)
Alarm 9 function	03,06,16	20708	16-bit unsign int	none(0}, and(1), or(2)
Alarm 10 function	03,06,16	20709	16-bit unsign int	none(0}, and(1), or(2)
Alarm 11 function	03,06,16	20710	16-bit unsign int	none(0}, and(1), or(2)
Alarm 12 function	03,06,16	20711	16-bit unsign int	none(0}, and(1), or(2)
Alarm 13 function	03,06,16	20712	16-bit unsign int	none(0}, and(1), or(2)
Alarm 14 function	03,06,16	20713	16-bit unsign int	none(0}, and(1), or(2)
Alarm 15 function	03,06,16	20714	16-bit unsign int	none(0}, and(1), or(2)
Alarm 16 function	03,06,16	20715	16-bit unsign int	none(0}, and(1), or(2)
Alarm 17 function	03,06,16	20716	16-bit unsign int	none(0}, and(1), or(2)
Alarm 18 function	03,06,16	20717	16-bit unsign int	none(0}, and(1), or(2)
Alarm 19 function	03,06,16	20718	16-bit unsign int	none(0}, and(1), or(2)
Alarm 20 function	03,06,16	20719	16-bit unsign int	none(0}, and(1), or(2)
Alarm 21 function	03,06,16	20720	16-bit unsign int	none(0}, and(1), or(2)
Alarm 22 function	03,06,16	20721	16-bit unsign int	none(0}, and(1), or(2)
Alarm 23 function	03,06,16	20722	16-bit unsign int	none(0}, and(1), or(2)
Alarm 24 function	03,06,16	20723	16-bit unsign int	none(0}, and(1), or(2)
Alarm 1 condition 2 channel	03,06,16	20800	16-bit unsign int	none(0), v01(1), v02(2), v03(3), v04(4), v05(5), v06(6), v07(7), v08(8), v09(9), v10(10), v11(11), v12(12), v13(13), v14(14), v15(15), v16(16),

				v17(17), v18(18), v19(19), v20(20), v21(21), v22(22), v23(23), v24(24)
Alarm 2 condition 2 channel	03,06,16	20801	16-bit unsign int	—"—
Alarm 3 condition 2 channel	03,06,16	20802	16-bit unsign int	—"—
Alarm 4 condition 2 channel	03,06,16	20803	16-bit unsign int	—"—
Alarm 5 condition 2 channel	03,06,16	20804	16-bit unsign int	—"—
Alarm 6 condition 2 channel	03,06,16	20805	16-bit unsign int	—"—
Alarm 7 condition 2 channel	03,06,16	20806	16-bit unsign int	—"—
Alarm 8 condition 2 channel	03,06,16	20807	16-bit unsign int	—"—
Alarm 9 condition 2 channel	03,06,16	20808	16-bit unsign int	—"—
Alarm 10 condition 2 channel	03,06,16	20809	16-bit unsign int	—"—
Alarm 11 condition 2 channel	03,06,16	20810	16-bit unsign int	—"—
Alarm 12 condition 2 channel	03,06,16	20811	16-bit unsign int	—"—
Alarm 13 condition 2 channel	03,06,16	20812	16-bit unsign int	—"—
Alarm 14 condition 2 channel	03,06,16	20813	16-bit unsign int	—"—
Alarm 15 condition 2 channel	03,06,16	20814	16-bit unsign int	—"—
Alarm 16 condition 2 channel	03,06,16	20815	16-bit unsign int	—"—
Alarm 17 condition 2 channel	03,06,16	20816	16-bit unsign int	—"—
Alarm 18 condition 2 channel	03,06,16	20817	16-bit unsign int	—"—
Alarm 19 condition 2 channel	03,06,16	20818	16-bit unsign int	—"—
Alarm 20 condition 2 channel	03,06,16	20819	16-bit unsign int	—"—
Alarm 21 condition 2 channel	03,06,16	20820	16-bit unsign int	—"—
Alarm 22 condition 2 channel	03,06,16	20821	16-bit unsign int	—"—
Alarm 23 condition 2 channel	03,06,16	20822	16-bit unsign int	—"—
Alarm 24 condition 2 channel	03,06,16	20823	16-bit unsign int	—"—
Alarm 1 condition 2 operand	03,06,16	20900	16-bit unsign int	larger(0), less(1)
Alarm 2 condition 2 operand	03,06,16	20901	16-bit unsign int	larger(0), less(1)
Alarm 3 condition 2 operand	03,06,16	20902	16-bit unsign int	larger(0), less(1)
Alarm 4 condition 2 operand	03,06,16	20903	16-bit unsign int	larger(0), less(1)
Alarm 5 condition 2 operand	03,06,16	20904	16-bit unsign int	larger(0), less(1)
Alarm 6 condition 2 operand	03,06,16	20905	16-bit unsign int	larger(0), less(1)
Alarm 7 condition 2 operand	03,06,16	20906	16-bit unsign int	larger(0), less(1)
Alarm 8 condition 2 operand	03,06,16	20907	16-bit unsign int	larger(0), less(1)
Alarm 9 condition 2 operand	03,06,16	20908	16-bit unsign int	larger(0), less(1)
Alarm 10 condition 2 operand	03,06,16	20909	16-bit unsign int	larger(0), less(1)
Alarm 11 condition 2 operand	03,06,16	20910	16-bit unsign int	larger(0), less(1)
Alarm 12 condition 2 operand	03,06,16	20911	16-bit unsign int	larger(0), less(1)
Alarm 13 condition 2 operand	03,06,16	20912	16-bit unsign int	larger(0), less(1)
Alarm 14 condition 2 operand	03,06,16	20913	16-bit unsign int	larger(0), less(1)
Alarm 15 condition 2 operand	03,06,16	20914	16-bit unsign int	larger(0), less(1)
Alarm 16 condition 2 operand	03,06,16	20915	16-bit unsign int	larger(0), less(1)
Alarm 17 condition 2 operand	03,06,16	20916	16-bit unsign int	larger(0), less(1)
Alarm 18 condition 2 operand	03,06,16	20917	16-bit unsign int	larger(0), less(1)
Alarm 19 condition 2 operand	03,06,16	20918	16-bit unsign int	larger(0), less(1)
Alarm 20 condition 2 operand	03,06,16	20919	16-bit unsign int	larger(0), less(1)
Alarm 21 condition 2 operand	03,06,16	20920	16-bit unsign int	larger(0), less(1)
Alarm 22 condition 2 operand	03,06,16	20921	16-bit unsign int	larger(0), less(1)

Alarm 23 condition 2 operand	03,06,16	20922	16-bit unsign int	larger(0), less(1)
Alarm 24 condition 2 operand	03,06,16	20923	16-bit unsign int	larger(0), less(1)
Alarm 1 condition 2 limit	03,16	21000	32-bit Float	
Alarm 2 condition 2 limit	03,16	21002	32-bit Float	
Alarm 3 condition 2 limit	03,16	21004	32-bit Float	
Alarm 4 condition 2 limit	03,16	21006	32-bit Float	
Alarm 5 condition 2 limit	03,16	21008	32-bit Float	
Alarm 6 condition 2 limit	03,16	21010	32-bit Float	
Alarm 7 condition 2 limit	03,16	21012	32-bit Float	
Alarm 8 condition 2 limit	03,16	21014	32-bit Float	
Alarm 9 condition 2 limit	03,16	21016	32-bit Float	
Alarm 10 condition 2 limit	03,16	21018	32-bit Float	
Alarm 11 condition 2 limit	03,16	21020	32-bit Float	
Alarm 12 condition 2 limit	03,16	21022	32-bit Float	
Alarm 13 condition 2 limit	03,16	21024	32-bit Float	
Alarm 14 condition 2 limit	03,16	21026	32-bit Float	
Alarm 15 condition 2 limit	03,16	21028	32-bit Float	
Alarm 16 condition 2 limit	03,16	21030	32-bit Float	
Alarm 17 condition 2 limit	03,16	21032	32-bit Float	
Alarm 18 condition 2 limit	03,16	21034	32-bit Float	
Alarm 19 condition 2 limit	03,16	21036	32-bit Float	
Alarm 20 condition 2 limit	03,16	21038	32-bit Float	
Alarm 21 condition 2 limit	03,16	21040	32-bit Float	
Alarm 22 condition 2 limit	03,16	21042	32-bit Float	
Alarm 23 condition 2 limit	03,16	21044	32-bit Float	
Alarm 24 condition 2 limit	03,16	21046	32-bit Float	
Alarm 1 condition 2 hysteresis	03,16	21100	32-bit Float	
Alarm 2 condition 2 hysteresis	03,16	21102	32-bit Float	
Alarm 3 condition 2 hysteresis	03,16	21104	32-bit Float	
Alarm 4 condition 2 hysteresis	03,16	21106	32-bit Float	
Alarm 5 condition 2 hysteresis	03,16	21108	32-bit Float	
Alarm 6 condition 2 hysteresis	03,16	21110	32-bit Float	
Alarm 7 condition 2 hysteresis	03,16	21112	32-bit Float	
Alarm 8 condition 2 hysteresis	03,16	21114	32-bit Float	
Alarm 9 condition 2 hysteresis	03,16	21116	32-bit Float	
Alarm 10 condition 2 hysteresis	03,16	21118	32-bit Float	
Alarm 11 condition 2 hysteresis	03,16	21120	32-bit Float	
Alarm 12 condition 2 hysteresis	03,16	21122	32-bit Float	
Alarm 13 condition 2 hysteresis	03,16	21124	32-bit Float	
Alarm 14 condition 2 hysteresis	03,16	21126	32-bit Float	
Alarm 15 condition 2 hysteresis	03,16	21128	32-bit Float	

Alarm 16 condition 2 hysteresis	03,16	21130	32-bit Float	
Alarm 17 condition 2 hysteresis	03,16	21132	32-bit Float	
Alarm 18 condition 2 hysteresis	03,16	21134	32-bit Float	
Alarm 19 condition 2 hysteresis	03,16	21136	32-bit Float	
Alarm 20 condition 2 hysteresis	03,16	21138	32-bit Float	
Alarm 21 condition 2 hysteresis	03,16	21140	32-bit Float	
Alarm 22 condition 2 hysteresis	03,16	21142	32-bit Float	
Alarm 23 condition 2 hysteresis	03,16	21144	32-bit Float	
Alarm 24 condition 2 hysteresis	03,16	21146	32-bit Float	
Alarm 1 condition 2 discrete al. state	03,06,16	21200	16-bit unsign int	open(0), closed(1)
Alarm 2 condition 2 discrete al. state	03,06,16	21201	16-bit unsign int	open(0), closed(1)
Alarm 3 condition 2 discrete al. state	03,06,16	21202	16-bit unsign int	open(0), closed(1)
Alarm 4 condition 2 discrete al. state	03,06,16	21203	16-bit unsign int	open(0), closed(1)
Alarm 5 condition 2 discrete al. state	03,06,16	21204	16-bit unsign int	open(0), closed(1)
Alarm 6 condition 2 discrete al. state	03,06,16	21205	16-bit unsign int	open(0), closed(1)
Alarm 7 condition 2 discrete al. state	03,06,16	21206	16-bit unsign int	open(0), closed(1)
Alarm 8 condition 2 discrete al. state	03,06,16	21207	16-bit unsign int	open(0), closed(1)
Alarm 9 condition 2 discrete al. state	03,06,16	21208	16-bit unsign int	open(0), closed(1)
Alarm 10 condition 2 discrete al. state	03,06,16	21209	16-bit unsign int	open(0), closed(1)
Alarm 11 condition 2 discrete al. state	03,06,16	21210	16-bit unsign int	open(0), closed(1)
Alarm 12 condition 2 discrete al. state	03,06,16	21211	16-bit unsign int	open(0), closed(1)
Alarm 13 condition 2 discrete al. state	03,06,16	21212	16-bit unsign int	open(0), closed(1)
Alarm 14 condition 2 discrete al. state	03,06,16	21213	16-bit unsign int	open(0), closed(1)
Alarm 15 condition 2 discrete al. state	03,06,16	21214	16-bit unsign int	open(0), closed(1)
Alarm 16 condition 2 discrete al. state	03,06,16	21215	16-bit unsign int	open(0), closed(1)
Alarm 17 condition 2 discrete al. state	03,06,16	21216	16-bit unsign int	open(0), closed(1)
Alarm 18 condition 2 discrete al. state	03,06,16	21217	16-bit unsign int	open(0), closed(1)

Alarm 19 condition 2 discrete al. state	03,06,16	21218	16-bit unsign int	open(0), closed(1)
Alarm 20 condition 2 discrete al. state	03,06,16	21219	16-bit unsign int	open(0), closed(1)
Alarm 21 condition 2 discrete al. state	03,06,16	21220	16-bit unsign int	open(0), closed(1)
Alarm 22 condition 2 discrete al. state	03,06,16	21221	16-bit unsign int	open(0), closed(1)
Alarm 23 condition 2 discrete al. state	03,06,16	21222	16-bit unsign int	open(0), closed(1)
Alarm 24 condition 2 discrete al. state	03,06,16	21223	16-bit unsign int	open(0), closed(1)
Alarm 1 type	03,06,16	21300	16-bit unsign int	warning(3), minor(4), major(5), critical(6)
Alarm 2 type	03,06,16	21301	16-bit unsign int	—"—
Alarm 3 type	03,06,16	21302	16-bit unsign int	—"—
Alarm 4 type	03,06,16	21303	16-bit unsign int	—"—
Alarm 5 type	03,06,16	21304	16-bit unsign int	—"—
Alarm 6 type	03,06,16	21305	16-bit unsign int	—"—
Alarm 7 type	03,06,16	21306	16-bit unsign int	—"—
Alarm 8 type	03,06,16	21307	16-bit unsign int	—"—
Alarm 9 type	03,06,16	21308	16-bit unsign int	—"—
Alarm 10 type	03,06,16	21309	16-bit unsign int	—"—
Alarm 11 type	03,06,16	21310	16-bit unsign int	—"—
Alarm 12 type	03,06,16	21311	16-bit unsign int	—"—
Alarm 13 type	03,06,16	21312	16-bit unsign int	—"—
Alarm 14 type	03,06,16	21313	16-bit unsign int	—"—
Alarm 15 type	03,06,16	21314	16-bit unsign int	—"—
Alarm 16 type	03,06,16	21315	16-bit unsign int	—"—
Alarm 17 type	03,06,16	21316	16-bit unsign int	—"—
Alarm 18 type	03,06,16	21317	16-bit unsign int	—"—
Alarm 19 type	03,06,16	21318	16-bit unsign int	—"—
Alarm 20 type	03,06,16	21319	16-bit unsign int	—"—
Alarm 21 type	03,06,16	21320	16-bit unsign int	—"—
Alarm 22 type	03,06,16	21321	16-bit unsign int	—"—
Alarm 23 type	03,06,16	21322	16-bit unsign int	—"—
Alarm 24 type	03,06,16	21323	16-bit unsign int	—"—
Alarm 1 assigned to channel	03,06,16	21400	16-bit unsign int	none(0), v01(1), v02(2), v03(3), v04(4), v05(5), v06(6), v07(7), v08(8),v09(9), v10(10), v11(11), v12(12), v13(13), v14(14), v15(15), v16(16),v17(17), v18(18), v19(19), v20(20), v21(21), v22(22), v23(23), v24(24)
Alarm 2 assigned to channel	03,06,16	21401	16-bit unsign int	—"—
Alarm 3 assigned to channel	03,06,16	21402	16-bit unsign int	—"—
Alarm 4 assigned to channel	03,06,16	21403	16-bit unsign int	—"—

Alarm 5 assigned to channel	03,06,16	21404	16-bit unsign int	—"—
Alarm 6 assigned to channel	03,06,16	21405	16-bit unsign int	—"—
Alarm 7 assigned to channel	03,06,16	21406	16-bit unsign int	—"—
Alarm 8 assigned to channel	03,06,16	21407	16-bit unsign int	—"—
Alarm 9 assigned to channel	03,06,16	21408	16-bit unsign int	—"—
Alarm 10 assigned to channel	03,06,16	21409	16-bit unsign int	—"—
Alarm 11 assigned to channel	03,06,16	21410	16-bit unsign int	—"—
Alarm 12 assigned to channel	03,06,16	21411	16-bit unsign int	—"—
Alarm 13 assigned to channel	03,06,16	21412	16-bit unsign int	—"—
Alarm 14 assigned to channel	03,06,16	21413	16-bit unsign int	—"—
Alarm 15 assigned to channel	03,06,16	21414	16-bit unsign int	—"—
Alarm 16 assigned to channel	03,06,16	21415	16-bit unsign int	—"—
Alarm 17 assigned to channel	03,06,16	21416	16-bit unsign int	—"—
Alarm 18 assigned to channel	03,06,16	21417	16-bit unsign int	—"—
Alarm 19 assigned to channel	03,06,16	21418	16-bit unsign int	—"—
Alarm 20 assigned to channel	03,06,16	21419	16-bit unsign int	—"—
Alarm 21 assigned to channel	03,06,16	21420	16-bit unsign int	—"—
Alarm 22 assigned to channel	03,06,16	21421	16-bit unsign int	—"—
Alarm 23 assigned to channel	03,06,16	21422	16-bit unsign int	—"—
Alarm 24 assigned to channel	03,06,16	21423	16-bit unsign int	—"—
Alarm 1 delay	03,16	21500	32-bit Float	
Alarm 2 delay	03,16	21502	32-bit Float	
Alarm 3 delay	03,16	21504	32-bit Float	
Alarm 4 delay	03,16	21506	32-bit Float	
Alarm 5 delay	03,16	21508	32-bit Float	
Alarm 6 delay	03,16	21510	32-bit Float	
Alarm 7 delay	03,16	21512	32-bit Float	
Alarm 8 delay	03,16	21514	32-bit Float	
Alarm 9 delay	03,16	21516	32-bit Float	
Alarm 10 delay	03,16	21518	32-bit Float	
Alarm 11 delay	03,16	21520	32-bit Float	
Alarm 12 delay	03,16	21522	32-bit Float	
Alarm 13 delay	03,16	21524	32-bit Float	
Alarm 14 delay	03,16	21526	32-bit Float	
Alarm 15 delay	03,16	21528	32-bit Float	
Alarm 16 delay	03,16	21530	32-bit Float	
Alarm 17 delay	03,16	21532	32-bit Float	
Alarm 18 delay	03,16	21534	32-bit Float	
Alarm 19 delay	03,16	21536	32-bit Float	
Alarm 20 delay	03,16	21538	32-bit Float	
Alarm 21 delay	03,16	21540	32-bit Float	
Alarm 22 delay	03,16	21542	32-bit Float	
Alarm 23 delay	03,16	21544	32-bit Float	
Alarm 24 delay	03,16	21546	32-bit Float	
Alarm 1 action on return	03,06,16	21600	16-bit unsign int	no(0), yes(1)
Alarm 2 action on return	03,06,16	21601	16-bit unsign int	no(0), yes(1)

Alarm 3 action on return	03,06,16	21602	16-bit unsign int	no(0), yes(1)
Alarm 4 action on return	03,06,16	21603	16-bit unsign int	no(0), yes(1)
Alarm 5 action on return	03,06,16	21604	16-bit unsign int	no(0), yes(1)
Alarm 6 action on return	03,06,16	21605	16-bit unsign int	no(0), yes(1)
Alarm 7 action on return	03,06,16	21606	16-bit unsign int	no(0), yes(1)
Alarm 8 action on return	03,06,16	21607	16-bit unsign int	no(0), yes(1)
Alarm 9 action on return	03,06,16	21608	16-bit unsign int	no(0), yes(1)
Alarm 10 action on return	03,06,16	21609	16-bit unsign int	no(0), yes(1)
Alarm 11 action on return	03,06,16	21610	16-bit unsign int	no(0), yes(1)
Alarm 12 action on return	03,06,16	21611	16-bit unsign int	no(0), yes(1)
Alarm 13 action on return	03,06,16	21612	16-bit unsign int	no(0), yes(1)
Alarm 14 action on return	03,06,16	21613	16-bit unsign int	no(0), yes(1)
Alarm 15 action on return	03,06,16	21614	16-bit unsign int	no(0), yes(1)
Alarm 16 action on return	03,06,16	21615	16-bit unsign int	no(0), yes(1)
Alarm 17 action on return	03,06,16	21616	16-bit unsign int	no(0), yes(1)
Alarm 18 action on return	03,06,16	21617	16-bit unsign int	no(0), yes(1)
Alarm 19 action on return	03,06,16	21618	16-bit unsign int	no(0), yes(1)
Alarm 20 action on return	03,06,16	21619	16-bit unsign int	no(0), yes(1)
Alarm 21 action on return	03,06,16	21620	16-bit unsign int	no(0), yes(1)
Alarm 22 action on return	03,06,16	21621	16-bit unsign int	no(0), yes(1)
Alarm 23 action on return	03,06,16	21622	16-bit unsign int	no(0), yes(1)
Alarm 24 action on return	03,06,16	21623	16-bit unsign int	no(0), yes(1)
Alarm 1 action 1	03,06,16	21700	16-bit unsign int	none(0), trapcond1(1), trapcond2(2), trapcond1and2(3), postiostate(4), mqttpublish(6)
Alarm 2 action 1	03,06,16	21701	16-bit unsign int	—"—
Alarm 3 action 1	03,06,16	21702	16-bit unsign int	—"—
Alarm 4 action 1	03,06,16	21703	16-bit unsign int	—"—
Alarm 5 action 1	03,06,16	21704	16-bit unsign int	—"—
Alarm 6 action 1	03,06,16	21705	16-bit unsign int	—"—
Alarm 7 action 1	03,06,16	21706	16-bit unsign int	—"—
Alarm 8 action 1	03,06,16	21707	16-bit unsign int	—"—
Alarm 9 action 1	03,06,16	21708	16-bit unsign int	—"—
Alarm 10 action 1	03,06,16	21709	16-bit unsign int	—"—
Alarm 11 action 1	03,06,16	21710	16-bit unsign int	—"—
Alarm 12 action 1	03,06,16	21711	16-bit unsign int	—"—
Alarm 13 action 1	03,06,16	21712	16-bit unsign int	—"—
Alarm 14 action 1	03,06,16	21713	16-bit unsign int	—"—
Alarm 15 action 1	03,06,16	21714	16-bit unsign int	—"—
Alarm 16 action 1	03,06,16	21715	16-bit unsign int	—"—
Alarm 17 action 1	03,06,16	21716	16-bit unsign int	—"—
Alarm 18 action 1	03,06,16	21717	16-bit unsign int	—"—
Alarm 19 action 1	03,06,16	21718	16-bit unsign int	—"—
Alarm 20 action 1	03,06,16	21719	16-bit unsign int	—"—
Alarm 21 action 1	03,06,16	21720	16-bit unsign int	—"—
Alarm 22 action 1	03,06,16	21721	16-bit unsign int	—"—

Alarm 23 action 1	03,06,16	21722	16-bit unsign int	—"—
Alarm 24 action 1	03,06,16	21723	16-bit unsign int	—"—
Alarm 1 action 2	03,06,16	21800	16-bit unsign int	none(0), trapcond1(1), trapcond2(2), trapcond1and2(3), postiostate(4), mqttpublish(6)
Alarm 2 action 2	03,06,16	21801	16-bit unsign int	—"—
Alarm 3 action 2	03,06,16	21802	16-bit unsign int	—"—
Alarm 4 action 2	03,06,16	21803	16-bit unsign int	—"—
Alarm 5 action 2	03,06,16	21804	16-bit unsign int	—"—
Alarm 6 action 2	03,06,16	21805	16-bit unsign int	—"—
Alarm 7 action 2	03,06,16	21806	16-bit unsign int	—"—
Alarm 8 action 2	03,06,16	21807	16-bit unsign int	—"—
Alarm 9 action 2	03,06,16	21808	16-bit unsign int	—"—
Alarm 10 action 2	03,06,16	21809	16-bit unsign int	—"—
Alarm 11 action 2	03,06,16	21810	16-bit unsign int	—"—
Alarm 12 action 2	03,06,16	21811	16-bit unsign int	—"—
Alarm 13 action 2	03,06,16	21812	16-bit unsign int	—"—
Alarm 14 action 2	03,06,16	21813	16-bit unsign int	—"—
Alarm 15 action 2	03,06,16	21814	16-bit unsign int	—"—
Alarm 16 action 2	03,06,16	21815	16-bit unsign int	—"—
Alarm 17 action 2	03,06,16	21816	16-bit unsign int	—"—
Alarm 18 action 2	03,06,16	21817	16-bit unsign int	—"—
Alarm 19 action 2	03,06,16	21818	16-bit unsign int	—"—
Alarm 20 action 2	03,06,16	21819	16-bit unsign int	—"—
Alarm 21 action 2	03,06,16	21820	16-bit unsign int	—"—
Alarm 22 action 2	03,06,16	21821	16-bit unsign int	—"—
Alarm 23 action 2	03,06,16	21822	16-bit unsign int	—"—
Alarm 24 action 2	03,06,16	21823	16-bit unsign int	—"—
Alarm 1 action 3	03,06,16	21900	16-bit unsign int	none(0), trapcond1(1), trapcond2(2), trapcond1and2(3), postiostate(4), mqttpublish(6)
Alarm 2 action 3	03,06,16	21901	16-bit unsign int	—"—
Alarm 3 action 3	03,06,16	21902	16-bit unsign int	—"—
Alarm 4 action 3	03,06,16	21903	16-bit unsign int	—"—
Alarm 5 action 3	03,06,16	21904	16-bit unsign int	—"—
Alarm 6 action 3	03,06,16	21905	16-bit unsign int	—"—
Alarm 7 action 3	03,06,16	21906	16-bit unsign int	—"—
Alarm 8 action 3	03,06,16	21907	16-bit unsign int	—"—
Alarm 9 action 3	03,06,16	21908	16-bit unsign int	—"—
Alarm 10 action 3	03,06,16	21909	16-bit unsign int	—"—
Alarm 11 action 3	03,06,16	21910	16-bit unsign int	—"—
Alarm 12 action 3	03,06,16	21911	16-bit unsign int	—"—
Alarm 13 action 3	03,06,16	21912	16-bit unsign int	—"—
Alarm 14 action 3	03,06,16	21913	16-bit unsign int	—"—

Alarm 15 action 3	03,06,16	21914	16-bit unsign int	—"—
Alarm 16 action 3	03,06,16	21915	16-bit unsign int	—"—
Alarm 17 action 3	03,06,16	21916	16-bit unsign int	—"—
Alarm 18 action 3	03,06,16	21917	16-bit unsign int	—"—
Alarm 19 action 3	03,06,16	21918	16-bit unsign int	—"—
Alarm 20 action 3	03,06,16	21919	16-bit unsign int	—"—
Alarm 21 action 3	03,06,16	21920	16-bit unsign int	—"—
Alarm 22 action 3	03,06,16	21921	16-bit unsign int	—"—
Alarm 23 action 3	03,06,16	21922	16-bit unsign int	—"—
Alarm 24 action 3	03,06,16	21923	16-bit unsign int	—"—
Alarm 1 status	03	22000	16-bit unsign int	undefined(0), normal(1), indeterminate(2), warning(3), minor(4), major(5), critical(6)
Alarm 2 status	03	22001	16-bit unsign int	—"—
Alarm 3 status	03	22002	16-bit unsign int	—"—
Alarm 4 status	03	22003	16-bit unsign int	—"—
Alarm 5 status	03	22004	16-bit unsign int	—"—
Alarm 6 status	03	22005	16-bit unsign int	—"—
Alarm 7 status	03	22006	16-bit unsign int	—"—
Alarm 8 status	03	22007	16-bit unsign int	—"—
Alarm 9 status	03	22008	16-bit unsign int	—"—
Alarm 10 status	03	22009	16-bit unsign int	—"—
Alarm 11 status	03	22010	16-bit unsign int	—"—
Alarm 12 status	03	22011	16-bit unsign int	—"—
Alarm 13 status	03	22012	16-bit unsign int	—"—
Alarm 14 status	03	22013	16-bit unsign int	—"—
Alarm 15 status	03	22014	16-bit unsign int	—"—
Alarm 16 status	03	22015	16-bit unsign int	—"—
Alarm 17 status	03	22016	16-bit unsign int	—"—
Alarm 18 status	03	22017	16-bit unsign int	—"—
Alarm 19 status	03	22018	16-bit unsign int	—"—
Alarm 20 status	03	22019	16-bit unsign int	—"—
Alarm 21 status	03	22020	16-bit unsign int	—"—
Alarm 22 status	03	22021	16-bit unsign int	—"—
Alarm 23 status	03	22022	16-bit unsign int	—"—
Alarm 24 status	03	22023	16-bit unsign int	—"—
Function 1 state	03	23000	16-bit unsign int	false(0), true(1)
Function 2 state	03	23001	16-bit unsign int	false(0), true(1)
Function 3 state	03	23002	16-bit unsign int	false(0), true(1)
Function 4 state	03	23003	16-bit unsign int	false(0), true(1)
Function 1 alarm state	03	23100	16-bit unsign int	normal(0), alarm(1)
Function 2 alarm state	03	23101	16-bit unsign int	normal(0), alarm(1)
Function 3 alarm state	03	23102	16-bit unsign int	normal(0), alarm(1)
Function 4 alarm state	03	23103	16-bit unsign int	normal(0), alarm(1)

Save configuration	03,06	24000	16-bit unsign int	unsaved(0), saved(1)
Restart device	03,06	24001	16-bit unsign int	cancel(0), restart(1)
HW error	03	24002	16-bit unsign int	noErr(0), hwErr(1)
Device ID	03	24100	18 bytes UTF-8	Example: 5c:32:c5:00:ac:52
Hostname	03	24200	16 bytes UTF-8	
Device IP	03	24300	16 bytes UTF-8	Example: 192.168.1.2

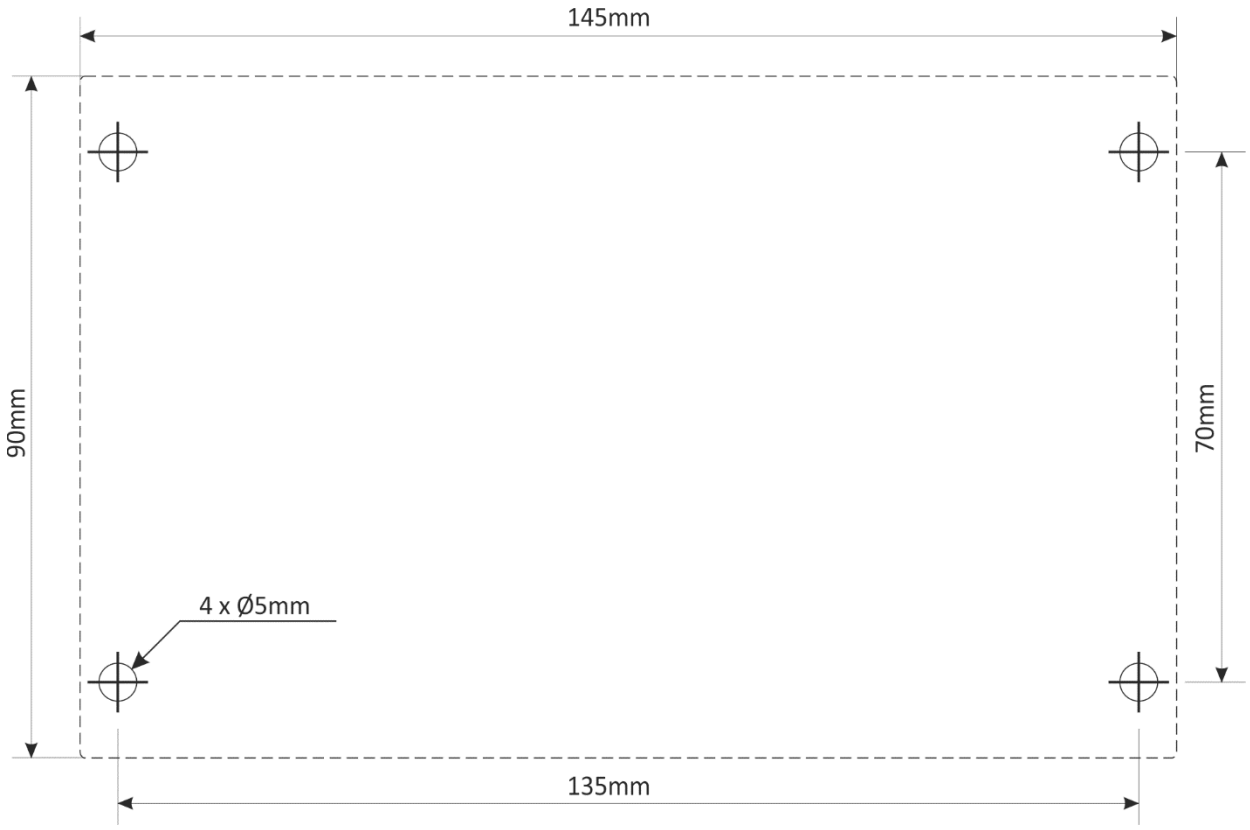

145mm
90mm
70mm
4 x Ø5mm
135mm

Fig.1

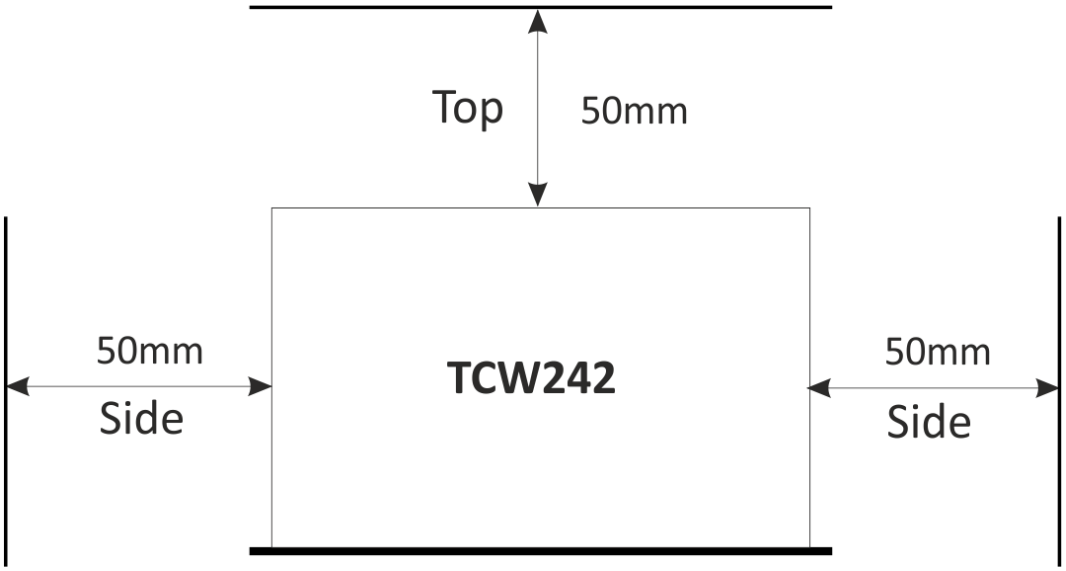

Top
50mm
50mm
Side
TCW242
50mm
Side

Fig.2