TCW260 Energy monitoring module
Version 1.5 / July 2020

USER MANUAL

www.teracomsystems.com
1. Introduction

TCW260 is an energy monitoring module with Ethernet connectivity and data logging capability. All monitored parameters can be seen as numbers and as graphs.

The device has 4 digital inputs, S0+ compatible. The inputs can work in two modes – OPEN/CLOSED for a reading of “dry contact” outputs and COUNTER mode for direct connection of energy meters with pulse outputs.

The energy monitoring module has also 6 analog inputs. Every analog input can work either in voltage (0/10V) or current loop (0/20mA) modes. The mode is changed over the user interface.

All digital and analog inputs are fully isolated from the power supply.

TCW260 supports MODBUS RTU interface for up to 24 Teracom and third-party sensors. The used RS-485 interface is fully isolated from the power supply.

The user can arrange up to 24 independent monitoring channels from the inputs and sensor readings. Every channel can be set up using up to 2 input parameters and/or constants. There are three type of channels – discrete (for OPEN/CLOSED outputs monitoring), general (for general monitoring) and cumulative (for energy, volume, etc. monitoring).

The user can also arrange up to 24 independent alarms with 5 different user selectable states. Every alarm can be set up using up to 2 limits and hysteresis. The alarm 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.

The device supports SNMP, HTTP API and MODBUS/TCP as machine-to-machine (M2M) communication.

2. Features

- 10/100 Mb Ethernet connectivity with Auto-MDIX;
- Password protected, web-based configuration and control;
- 4 digital inputs, S0-pulse interface compatible;
- 6 analog inputs with 0/10V and 0/20mA modes;
- MODBUS RTU interface for up to 24 sensors (registers);
- Up to 24 channels for monitoring;
- Up to 24 independent alarms;
- SNMP v.2 support;
- SNMP traps sending for alert conditions;
- HTTP and SNMP port changing;
- HTTP API commands;
- Periodical HTTP Post of XML/JSON status files for client-server systems;
- MODBUS TCP/IP support;
- Dynamic DNS with DynDNS, No-IP and DNS-O-Matic support;
- NTP protocol support;
- Data logger for up to 70000 records;
- DIN Rail Mountable;
- Wide power supply voltage range;
- Backup/restore of settings;
- Remote firmware update.
3. Applications

The energy monitoring module TCW260 is dedicated to monitoring and recording the parameters of the measurement of resources - electricity meters, gas meters, water meters, and others. Rising resource costs require reliable analysis and optimization. Depending on the search result, this can be done at a micro level (separate machine) or macro level (company).

The monitoring module can integrate seamlessly into already working objects. With proper selection of sensors, this can happen even without interrupting the production process.

Of course, the module can be used also for general monitoring of industrial processes.

A few example applications include:

- Energy analysis and optimization;
- Water consumption analysis;
- Gas consumption optimization;
- A building management system;
- Industrial processes monitoring;
- General SCADA systems.

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 requirements
  Supply Voltage: 10 to 32 VDC
  Input Current: 220 mA @ 12 VDC (without RS-485 powering)

- RS-485 interface
  Isolation: Isolated (1000VDC)
  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: Isolated (1000VDC)
  Mode: OPEN/CLOSED (“Dry contact”) or COUNTER (S0-pulse interface outputs)
  Maximum input voltage: +5.5VDC
  Sampling rate: 1mS
  Digital filtering time interval: 5 to 60000mS
• Analog inputs
  Isolation: Isolated (1000VDC)
  Type: Single ended
  Resolution: 10 bits
  Mode: Voltage or current loop
  Input Range: 0/10V or 0/20mA
  Accuracy: ±1%
  Sampling Rate: 600mS per channel (averaged value of 100 samples)
  Input Impedance: 1 mega-ohm (min.)

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

5. LED indicators
   The following indicators show the status of the controller:
   • PWR (red) – in working mode shines, blinks together with STS if there is a hardware error;
   • STS (yellow) – flashes when the main program of the controller is executed;
   • NET (orange) – network status - ON when a link is established, blinks if there is an activity.

6. Installation and setup
   This device must be installed by qualified personnel.
   This device must not be installed directly outdoors.
   The Installation consists of mounting the device, connecting to an IP network, connecting inputs and outputs, providing power and configuring via a web browser.

6.1. Mounting
   TCW260 should be mounted in a clean and dry location on a not flammable surface. Ventilation is recommended for installations where the ambient air temperature is expected to be high.
   Mount the device to a wall by using two plastic dowels 8x60mm (example Würth GmbH 0912 802 002) and two dowel screws 6x70mm (example Würth GmbH 01 57 06 70). Attach the screws to the surface vertically. See Appendix-A, fig. 1 for mechanical details.
   Maintain spacing from adjacent equipment. Allow 50 mm of space on all sides, as shown in fig. 2 in Appendix A, this provides ventilation and electrical isolation.
   TCW260 can be mounted to a standard (35mm by 7.55mm) DIN rail. Attach the controller to the DIN rail by hooking the hook on the back of the enclosure to the DIN rail and then snap the bottom hook into place.

6.2. Connection
   Attention! Disconnect power supply before wiring.
   The correct wiring procedure is as follows:
   • Make sure power is turned off;
   • Make wiring connections to the terminals;
   • Apply power.
Make sure that the wires are properly attached to the terminals and that the terminals are tightened. Not the proper wiring and configuration can cause permanent damage to TCW260 or the equipment to which it is connected or both.

**Connector 1**
- Ethernet - RJ45

**Connector 2**
- Power - 2.1x5.5mm connector, (Central positive)

**Connector 3**
- Pin1 – Power positive
- Pin2 – Power negative (GND)

**Connector 4**
- Pin1 – S04- (SGND)
- Pin2 – not connected
- Pin3 – S04+ (Digital in 4)
- Pin4 – S03- (SGND)
- Pin5 – not connected
- Pin6 – S03+ (Digital in 3)
- Pin7 – S02- (SGND)
- Pin8 – not connected
- Pin9 – S02+ (Digital in 2)
- Pin10 – S01- (SGND)
- Pin11 – not connected
- Pin12 – S01+ (Digital in 1)

**Connector 5**
- Pin1 – not connected (most left)
- Pin2 – not connected
- Pin3 – not connected

**Connector 6**
- Pin1 – D0/A+
- Pin2 – SGND
- Pin3 – D1/B-

**Connector 7**
- Pin1 – Analog in 1
- Pin2 – SGND
- Pin3 – Analog in 2
- Pin4 – SGND
- Pin5 – Analog in 3
- Pin6 – SGND
- Pin7 – Analog in 4
- Pin8 – SGND
- Pin9 – Analog in 5
- Pin10 – SGND
- Pin11 – Analog in 6
- Pin12 – SGND

It is recommended using grounding configuration with isolated local ground and PE.
6.2.1. Power supply connection

TCW260 is designed to be supplied by adapter SYS1421-0612-W2E or similar, intended for use in the conditions of overvoltage category II, and prior assessed for compliance with safety requirements. The power supply equipment shall be resistant to short circuit and overload in a secondary circuit.

When in use, do not position the equipment so that it is difficult to disconnect the device from the power supply.

6.2.2. Digital inputs connection

All inputs are galvanic isolated from the power supply.

The digital inputs of TCW260 in OPEN/CLOSED mode can be used for monitoring of devices with “dry contact” outputs – door contact switch, push button, PIR detector, etc.

The following picture illustrates how a dry contact switch can be connected to the input of TCW260. One side of the contact is connected to “S0+” and another side is connected to “S0-” terminals.

The digital inputs of TCW260 in COUNTER mode can be used for monitoring of devices with S0-pulse interface – energy meters, water meters, etc.

The following picture illustrates how an energy meter is connected to the input of TCW260. One side of the contact is connected to “S0+” and another side is connected to “S0-” terminals.

The maximum cable length should be up to 30 meters.
6.2.3. **Analog inputs connection**

All inputs are galvanic isolated from the power supply. Analog inputs of TCW260 can be used for monitoring of devices with voltage and current loop outputs. They can be connected directly to analog sensors for temperature, humidity, current/voltage transducers, etc.

The following picture illustrates how a battery can be connected to the analog input of TCW260 in voltage mode. The positive terminal is connected to “Analog In” and the negative terminal to “GND”.

![Battery Connection](image1.png)

The following picture illustrates how an analog sensor for temperature with current loop output is connected to the analog input. The active terminal is connected to “Analog In” while the shield terminal to “GND”.

![Sensor Connection](image2.png)

The maximum cable length should be up to 30 meters.

6.2.4. **RS-485 connection**

RS-485 interface is galvanic isolated from the power supply. Up to 24 MODBUS RTU sensors can be connected to TCW260. The device supports Teracom and third-party sensors.

Connections can be realized by a standard RJ-45 connector. The used pinout is that one recommended in the document “MODBUS over Serial Line Specification and Implementation Guide” available on www.modbus.org.
It is mandatory to use 120ohms line terminators at both ends of the bus. TCW260 incorporates one of the terminators and should be placed at one end. So the client should only take care of terminating at the other end of the line.

It is strongly recommended using “daisy-chained” (linear topology) for multiple sensors:

For sensors with screw terminals following connection is possible:

It is recommended to use only UTP/FTP cables and keep total cable length up to 30 m, although functionality has been achieved on a longer distance.
6.2.5. Network connection

The Ethernet port of TCW260 should be connected to 10/100 Base-T Ethernet hub, switch or router.

For configuration, TCW260 may be connected directly to the Ethernet port on a computer. The device support Auto-MDI and it is not necessary to use “crossover” cable, standard “straight-through” can be also used.

TCW260 can be used in a wireless network by connecting through a wireless router.

6.3. Communication setup

By default TCW260 is delivered with the following network settings:

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

Communication with TCW260 can be established by assigning a temporary IP address to the computer. For computers with Windows OS assigning of IP address is made in “Local area connection properties”:
This address should be on the same network - for example 192.168.1.3:

To get access to the web interface, you should type http://192.168.1.2 into the browser.
If the network settings are correct, the login pop-up window will appear:

Authorization data must be entered (by default username=admin, password=admin).
It is recommended to change the username and password to prevent unauthorized access to the controller.
All TCW controllers connected to LAN can be easily found by a free tool “TCW discoverer”. It is available for Win and Mac operating systems and can be downloaded from www.teracomsystems.com.

7. Set up concept
The main set up channel of the device is the web interface. It is recommended to use it, even more of the settings are available through SNMP and HTTP API commands.
The device set up begin with network settings.
Then the so-called primary parameters (MODBUS RTU sensors/registers, analog inputs, and digital inputs) are set up.
The channels are formed from already set up primary parameters. Every channel can combine up to two primary parameters and constants with math operations. It is possible that a channel is formed by only one primary parameter.
Once all channels have been tuned, the alarm can be set up. It is important to remember that alarms work with channels, not with primary parameters. Each alarm can have up to two conditions. Different channels may be involved in the conditions. Alarms are independent of the channels but can also be assigned to a channel.
In other words, the right set up should follow the following order:

Once everything above is set correctly, the wanted services (data logger, SNMP, HTTP API, etc.) can be turned on.

8. Web interface
The web interface allows configuration and monitoring. All pages are UTF-8 encoded.
The controller supports a few active session.

8.1. Monitoring
This section displays the status of all channels and alarms – textually and graphically.
The pages “Channels” and “Alarms” are automatically refreshed on an interval of 0 to 253 seconds. Zero means no automatic refresh. This parameter is set in section “Setup-System-Refresh of channels and alarms pages”. By default, the refresh interval is 1 second.
8.1.1. Channels
This page displays the status of all monitored channels – their values and alarm status. The information is updated on the refresh interval.

<table>
<thead>
<tr>
<th>Channels</th>
<th>Description</th>
<th>Value</th>
<th>Unit</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Temperature</td>
<td>25.876</td>
<td>°C</td>
<td>Warning</td>
</tr>
<tr>
<td>2</td>
<td>Humidity</td>
<td>45.967</td>
<td>RH</td>
<td>Normal</td>
</tr>
<tr>
<td>3</td>
<td>Digital Input</td>
<td>OPEN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>V95-Voltage</td>
<td>229.877</td>
<td>V</td>
<td>Minor</td>
</tr>
<tr>
<td>5</td>
<td>V96-Current</td>
<td>4.006</td>
<td>A</td>
<td>Normal</td>
</tr>
<tr>
<td>6</td>
<td>V97-Energy</td>
<td>6122.271</td>
<td>kWh</td>
<td></td>
</tr>
</tbody>
</table>

8.1.2. Alarms
This page displays the status of all alarms. The information is updated on the refresh interval.

<table>
<thead>
<tr>
<th>Alarms</th>
<th>Description</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>AL01-Temp.</td>
<td>Warning</td>
</tr>
<tr>
<td>2</td>
<td>AL02</td>
<td>Minor</td>
</tr>
<tr>
<td>3</td>
<td>AL03</td>
<td>Normal</td>
</tr>
<tr>
<td>4</td>
<td>AL04-D1</td>
<td>Critical</td>
</tr>
<tr>
<td>5</td>
<td>AL05-Humidity</td>
<td>Normal</td>
</tr>
</tbody>
</table>

8.1.3. Graphs
All channels together with their alarm statuses can be monitored graphically in tabs Graph-1 to Graph-6. Every graph supports up to 4 channels with up to 2 different dimensions. The curve’s color for every channel is selectable. The alarms status colors are fixed. There are a few checkboxes for display modification.

Graphs show information from the past, so it is mandatory that the logger is active.
It is important to know that the information on graphs is static, it is not updated with the newest values. If you want to see the last information, the page should be reloaded. The information can be exported in CSV file.
8.2. Setup

8.2.1. Network

The network parameters can be set on this page.

The controller supports static and dynamic IP addresses. It is good practice to change the default IP address of the controller immediately after first power-on. This will avoid collisions if many devices are used on the same network. It may be necessary to clear the arp cache, each time you connect a new device to the network. This is done by typing `arp -d` in the command prompt window of the computer.

The “Hostname” is up to 15 characters. It is shown in search results of TCW discoverer. It is recommended to use public DNS server (8.8.8.8, 8.8.4.4, etc.) rather than the default gateway.

8.2.2. Modbus sensors

8.2.2.1. Modbus RTU communication setup

This section allows you to set the communication parameters of the RS-485 interface – bit rate, parity, and a number of stop bits. By default, the settings are 19200, even parity, and 1 stop bit. It is mandatory that all sensors on the bus use the same bit rate, parity, and a number of stop bits. Before to add any sensor to the interface its parameters should be set up properly.

In the right part of the section, there is a tool which scans the bus and reports the number of the found sensor together with their addresses. It is very useful at a time when you add new sensors. It is recommended to use a small address segment to speed up the scan process.

Playing with “Scan time-out for sensor answer” it is possible to find this parameter for an unknown sensor. The test starts with a large time-out (for example 500ms) and gradually decreases the time until the sensor stops responding. In order for the operation sustainability, the found time should be increased with for example 20%.
8.2.2.2. Modbus RTU sensors

This section allows you to add, delete or edit MODBUS RTU sensors/registers. All they are primary parameters and can be used in forming of channels.

It is recommended to add sensors/registers one by one using the scan tool described in 8.2.2.1. Up to 24 sensors/registers can be added. All they are shown in the table.

According to MODBUS convention, possible addresses for sensors are in the range from 1 to 247.

The Multiplier and offset works as follows:

\[ \text{Value} = (\text{Raw Value} \times \text{Multiplier}) + \text{Offset}. \]

If you want to see the raw value of the sensor/register set Multiplier=1 and Offset=0.

During operation, all sensors are polled consecutively. The controller expects an answer in "response time-out". If the sensor does not respond for that time, the controller addresses the next sensor. If the same sensor does not respond in series 3 times, the device assumes that it is not present in the system but continues to polls it.

According to the above paragraph, special attention should be paid for response time-out. On the one hand, the time-out should be large enough for the sensor to respond, but on the other hand, the sum of all sensor time-outs forms the "maximum response time-out" of the whole system. The "maximum response time-out" determines the response of the system.

For the sustainable system operation, polling of every sensor is made at a fixed time - "Polling time". It is user selectable and can be 1, 2, 3 or 4 seconds. By default, it is 1 second.

It is recommended that "Polling time" \( \geq "\text{Maximum response time-out}" \).
8.2.2.3. **Sensor setup tool**

The device has also a simple but useful tool for configuration and control of MODBUS RTU sensors. It can change the addresses and communication parameters of sensors from different manufacturers.

8.2.3. **Inputs**

This page is used for parameterization of analog and digital inputs

**8.2.3.1. Analog inputs**

TCW260 has 6 analog inputs. All they are isolated from the power supply but use the same ground. Every analog input can work in voltage (0-10V) or current loop mode (0-20mA).

For every analog input Multiplier and Offset can be set. They work as follows:

\[
\text{Value} = (\text{Raw\_Value} \times \text{Multiplier}) + \text{Offset}.
\]

If you want to see the raw value of the sensor/register set Multiplier=1 and Offset=0.

By default and after “Factory default settings” procedure: Multiplier=1.00, Offset=0.00 and Mode=0-10V
Example:
For humidity sensor HIH-4000-003 the following parameter (coming from datasheet) should be set for fine work:
- Multiplier: -31.74
- Offset: -0.826
- Mode: 0-10V
- Unit: %RH (is set in the channels web page)
The value of multiplier is inverted of slope parameter (1/0.0315);
If the output voltage of this sensor is 3.198V the value of analog input will be 75.28% RH:
\[ 75.28 = \frac{3.198 - 0.826}{0.0315} \times 31.74 \]

8.2.3.2. Digital inputs
TCW260 has 4 digital inputs. All they are isolated from the power supply but use the same ground. Every digital input can work in OPEN/CLOSE or COUNTER mode. In COUNTER mode, counting can be made on rising, falling or both edges.
For the counter mode, there is a possibility for setting up of the counter initial value.

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

There are two delays – Open-to Close and Close-to-Open. The delays are in the range of 5 to 60000mS. These delays can be used for additional digital filtering. The delays are applied in both modes.

On the picture above Open-to Close and Close-to-Open delays are set to 13mS.

8.2.4. Channels
This section allows you to add, delete or edit channels. All they can be monitored on monitoring pages and their values can be recorded periodically by the data logger.
Up to 24 channels can be set up.
There are three types of channels – discrete (it forms by one digital input in OPEN/CLOSE mode),
general (it forms by up to 2 primary parameters and constants) and cumulative (same as a general
channel but with cumulating the value in time).
Cumulative channels are used for energy, volume, etc. monitoring. For them, there is a possibility
for setting up of the initial value.
It is important to know that for general and cumulative channels, the sequence of operations is
OP1, then OP2 and finally OP3.
It is possible to use a digital input in OPEN/CLOSE mode to form a general or cumulative channel.
In this case, its values for operations are 0 (for CLOSE) and 1 (for OPEN). For example, this is useful
if you want to stop the integration for the cumulative channel from an external signal.

8.2.5. Alarms
This section allows you to add, delete or edit alarms. Up to 24 alarms can be set up.
There is four types of alarms – Warning, Minor, Major and Critical.
Only channels can be used to form the alarms.
The alarms are independent but can be assigned to a channel. There isn’t a restriction to which
channel to assign the alarm.
For every alarm up to 2 conditions can be used. Both conditions can be logically joined using the
logical operators AND and OR.
It is not mandatory to use the same channel in both conditions. In this case, conditions for two
different channels can be combined in one alarm.

The “Limit” indicates the border of the working range for the observed channel.
A trigger condition occurs when the value goes higher (“Sign” is “≥”) and lower (“Sign” is “<”) than
the limit. In both cases, the monitored parameter goes out of range.
Coming back in range for the observed channel is considered when the value goes higher than
Limit + Hysteresis (“Sign” is “<”) or lower than Limit – Hysteresis (“Sign” is “≥”).
The hysteresis is used to prevent excessively triggering when the value vacillates around the trigger
point.
It is strongly recommended not to use Hysteresis=0.0.
For every alarm “Return notification” option is available. If this option is chosen there will be notification also when parameter returns in range.

For every alarm, there is “Notification delay” parameter. It is very useful as a filter for short alarm conditions.
8.2.6. System

On this page, some general system parameters can be set up.

System name, system location, and system contact are used for device identification. They are presented in SNMP three and XML/JSON status file.

WEB access authentication by default is active with admin/admin details.

HTTP port for WEB access by default is 80 but it can be changed. This is useful for some routers which don’t support different outside/inside ports for port forwarding.

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 real-time clock of the controller can be set either manually or automatically. For automatic clock synchronization, the controller supports NTP (Network Time Protocol).

By default NTP synchronization is disabled, server – time.google.com, Time zone +00:00 and interval of 12 hours.

8.3.2. SNMP

TCW260 supports SNMP V2. With this M2M protocol, the device can be part of SNMP monitoring and control systems.

By default SNMP is disabled, the port is 161, read community is public and write community is private.
In an alarm condition, an SNMP trap can be sent up to 5 independent recipients. For every trap different port and community can be used.
SNMP trap is sent after reset.
Actual MIB file can be downloaded from here.

8.3.3. Logger

The logger works in three modes – Time, Alarm and Time&Alarm. The mode specifies what initiates a record in the logger’s memory.

In Time mode, records are made periodically on “Log interval” time. In Alarm mode, records are made on every alarm condition. In Time&Alarm mode, a mix of both conditions for records is used.

The log interval determines the time between two log entries. It is good to remember that by reducing the log interval, we increase the resolution, but we also reduce the past period for which we have records.

The logger records can be synchronized with a specific minute in an hour. Synchronization is very useful when monitoring electricity, water, gas meters, etc. The log interval can be chosen from a drop-down menu between 1 and 60 minutes. The field „Sync to the minute” determines which minute of every hour is used for synchronization. Although any minute can be selected, it is better 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.

The settings determine 4 records per hour in **HH:00**, **HH:15**, **HH:30**, and **HH:45**.

The device is powered up.
The first record will be immediately after power-up - 09:12. The next records will be in 09:15, 09:30, 09:45, 10:00, 10:15, etc.
There are two ways to reach the logger records:

- download of the full log file, using “Download full log” in the WEB interface;
- periodical upload the last unsent records to the dedicated HTTP server.

The records are uploaded in CSV file format. The period of the upload can be chosen from the menu between 1 and 24 hours. If you enable this service, take care of the real-time clock (NTP service).

The HTTP server for upload can be domain or IP address but take care about DNS settings.

“Sync time” is a moment in the day when a period of upload is synchronized.

Example:

Current time is 19:31, Upload period is 3 hours and Sync time is 9:00.

To synchronize the logger to 9:00 it means that time for uploads will be: 09:00, 12:00, 15:00, 18:00, 21:00, 24:00, 03:00 and 06:00. The first upload, after enabling the logger in 19:31, will be in 21:00.

The button “Force upload” initiates upload recorded information between previous periodical upload and now.

By default, the logger is disabled.

More about the logger can be found in the Data logger section.

8.3.4. HTTP Post

HTTP Post is used for periodically upload an XML/JSON file to HTTP server by HTTP requests (POST method). The XML/JSON file contains the current status of all monitored parameters and extra system information. The file format is chosen from the drop-down menu.

The HTTP Post can be sent up to 3 independent servers. The HTTP servers can be addressed either by domain name or IP address.

The “Period” can be set between 10 and 14400 seconds. This parameter can be changed remotely also by HTTP API. The "Period" determines at what time the control software receives up-to-date information from TCW260 and can, therefore, make changes to some of the parameters. The shorter is "Period", the closer to the real-time operation is the system. On the other hand, as shorter is the "Period" as higher is the data traffic through the network.

If the checkbox “Connect on any alarm” is selected, the HTTP Post request will be sent in an alarm condition.
The “Key” field is user defined. Its value is sent in XML/JSON file and can be used for device identification.

If “Process Answer” option is enabled, TCW260 will execute the commands, sent by the remote server as an answer of HTTP Post.

More about HTTP Post can be read in HTTP API section.

8.3.5. Dynamic DNS

TCW260 supports the following DNS services – DynDNS, No-IP, and DNS-O-Matic.

With dynamic DNS, TCW260 can be accessed from the Internet having only a dynamic public IP address.

8.3.6. MODBUS

TCW260 supports MODBUS TCP/IP.

The standard port for this protocol is 502.
By default, Modbus is disabled.
More about this functionality can be read at MODBUS section.
8.4. Administration

8.4.1. User/Pass
The TCW260 supports one user only. It has administrative rights.

![Admin access form](image)

8.4.2. Backup/Restore
The TCW260 supports backup and restore of all user setting. All settings are saved in XML backup file. This file can be used after this to restore on many devices. This is very useful for multiplying similar settings to a batch of controllers.

![Backup/Restore configuration](image)

8.4.3. FW update
The TCW260 can be updated via a WEB interface.

![Firmware update](image)

To update the device follow the steps below:
- Go to [www.teracomsystems.com](http://www.teracomsystems.com) and download the latest firmware;
- From Administration->FW update select downloaded .cod file and press “UPLOAD” button;
- After the firmware update is completed, the Login page will appear.

**Attention! Don’t turn off the power supply during the update. Turning off the power supply will damage the device.**

8.5. Logout
The TCW260 support a few sessions, but the good practice is to log out after finishing the work.
9. Protocols and API

9.1. SNMP

Simple Network Management Protocol (SNMP) is a standard internet protocol for managing devices on IP networks. In typical usage of SNMP, one or more administrative computers, called managers, monitor and control devices on LAN. Each controlled device, at all times, executes a software component called an agent which reports information via SNMP to the manager.

The TCW260 can be configured and monitored through SNMP.

This could be done using every SNMP v.2 compatible program. Parameters that can be changed, are grouped according to their functions in the tables below. To obtain a valid OID number, replace symbol “I” with “1.3.6.1.4.1.38783”.

To save the changes **configurationSaved** (OID !.6.6.3.0) should be set to "1".

**Setup -> network**

replace “!” with “1.3.6.1.4.1.38783” in the table below

<table>
<thead>
<tr>
<th>OID</th>
<th>Name</th>
<th>Access</th>
<th>Description</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>.6.2.1.1.0</td>
<td>deviceID</td>
<td>read-only</td>
<td>Device ID (default MAC address)</td>
<td>MacAddress</td>
</tr>
<tr>
<td>.6.2.1.2.0</td>
<td>hostName</td>
<td>read-only</td>
<td>Hostname</td>
<td>DisplayString</td>
</tr>
<tr>
<td>.6.2.1.3.0</td>
<td>deviceIP</td>
<td>read-only</td>
<td>Device IP address</td>
<td>IpAddress</td>
</tr>
</tbody>
</table>

**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

<table>
<thead>
<tr>
<th>OID</th>
<th>Name</th>
<th>Access</th>
<th>Description</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>.6.2.2.1.1.1.2.0</td>
<td>mbSenDescription</td>
<td>read-write</td>
<td>Sensor description</td>
<td>DisplayString</td>
</tr>
<tr>
<td>.6.2.2.1.1.1.3.0</td>
<td>mbSenMult</td>
<td>read-write</td>
<td>Sensor multiplier x1000 in Integer format</td>
<td>Integer32</td>
</tr>
<tr>
<td>.6.2.2.1.1.1.4.0</td>
<td>mbSenOffset</td>
<td>read-write</td>
<td>Sensor offset x1000 in Integer format</td>
<td>Integer32</td>
</tr>
<tr>
<td>.6.2.2.1.1.1.5.0</td>
<td>mbSenVal</td>
<td>read-only</td>
<td>Sensor value x1000 in Integer format</td>
<td>Integer32</td>
</tr>
<tr>
<td>.6.2.2.1.1.1.6.0</td>
<td>mbSenCounter</td>
<td>read-only</td>
<td>Sensor as 32-bit Counter</td>
<td>Counter32</td>
</tr>
</tbody>
</table>

**Setup -> parameters -> analogInputs -> analogInpTable -> analogInpEntry -> analogInpIndex 1 to 6**

replace “?” with a number from 1 to 6 and “!” with “1.3.6.1.4.1.38783” in the table below

<table>
<thead>
<tr>
<th>OID</th>
<th>Name</th>
<th>Access</th>
<th>Description</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>.6.2.2.2.1.1.2.0</td>
<td>analogInpDescription</td>
<td>read-write</td>
<td>Analog input description</td>
<td>DisplayString</td>
</tr>
<tr>
<td>.6.2.2.2.1.1.3.0</td>
<td>analogInpMult</td>
<td>read-write</td>
<td>Analog input multiplier x1000 in Integer format</td>
<td>Integer32</td>
</tr>
<tr>
<td>.6.2.2.2.1.1.4.0</td>
<td>analogInpOffset</td>
<td>read-write</td>
<td>Analog input offset x1000 in Integer format</td>
<td>Integer32</td>
</tr>
<tr>
<td>.6.2.2.2.1.1.5.0</td>
<td>analogInpMode</td>
<td>read-write</td>
<td>Analog input mode - 0-10V or 4-20mA</td>
<td>Integer32</td>
</tr>
<tr>
<td>.6.2.2.2.1.1.6.0</td>
<td>analogInpValue</td>
<td>read-only</td>
<td>Analog input value x1000 in Integer format</td>
<td>Integer32</td>
</tr>
</tbody>
</table>

**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

<table>
<thead>
<tr>
<th>OID</th>
<th>Name</th>
<th>Access</th>
<th>Description</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>.6.2.2.3.1.1.2.0</td>
<td>digInpDescription</td>
<td>read-write</td>
<td>Digital input description</td>
<td>DisplayString</td>
</tr>
<tr>
<td>.6.2.2.3.1.1.3.0</td>
<td>digInpLowLevel</td>
<td>read-write</td>
<td>Digital Input closed state</td>
<td>DisplayString</td>
</tr>
<tr>
<td>.6.2.2.3.1.1.4.0</td>
<td>digInpHighLevel</td>
<td>read-write</td>
<td>Digital Input open state</td>
<td>DisplayString</td>
</tr>
<tr>
<td>.6.2.2.3.1.1.5.0</td>
<td>digInpMode</td>
<td>read-write</td>
<td>Digital input mode - Discrete or Counter</td>
<td>INTEGER {</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>openclosed(0),</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>risingEdge(1),</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>fallingEdge(2),</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>bothEdges(3)</td>
</tr>
<tr>
<td>.6.2.2.3.1.1.6.0</td>
<td>digInpCloseToOpenDelay</td>
<td>read-write</td>
<td>Digital input Close To Open delay</td>
<td>Integer32(0..60000)</td>
</tr>
<tr>
<td>.6.2.2.3.1.1.7.0</td>
<td>digInpOpenToCloseDelay</td>
<td>read-write</td>
<td>Digital input Open To Close delay</td>
<td>Integer32(0..60000)</td>
</tr>
<tr>
<td>.6.2.2.3.1.1.8.0</td>
<td>digInpCounterInitialValue</td>
<td>read-only</td>
<td>Digital input counter initial value</td>
<td>Integer32</td>
</tr>
<tr>
<td>.6.2.2.3.1.1.9.0</td>
<td>digInpValue</td>
<td>read-only</td>
<td>Digital input value</td>
<td>Unsigned32</td>
</tr>
</tbody>
</table>
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

<table>
<thead>
<tr>
<th>OID</th>
<th>Name</th>
<th>Access</th>
<th>Description</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.6.3.1.1.2.7.0</td>
<td>chType.</td>
<td>read-write</td>
<td>Channel type</td>
<td>INTEGER [general(0), cumulative(1), discrete(2), counter(3)]</td>
</tr>
<tr>
<td>1.6.3.1.1.3.7.0</td>
<td>chdescription.</td>
<td>read-write</td>
<td>Channel description</td>
<td>DisplayString</td>
</tr>
<tr>
<td>1.6.3.1.1.4.7.0</td>
<td>chParam1.</td>
<td>read-write</td>
<td>Channel parameter 1</td>
<td>INTEGER [general(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)]</td>
</tr>
<tr>
<td>1.6.3.1.1.5.7.0</td>
<td>chOP1.</td>
<td>read-write</td>
<td>Channel operand 1</td>
<td>INTEGER [none(0), multiplication(1), division(2), sum(3), subtract(4)]</td>
</tr>
<tr>
<td>1.6.3.1.1.6.7.0</td>
<td>chParam2.</td>
<td>read-write</td>
<td>Channel parameter 2</td>
<td>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)]</td>
</tr>
<tr>
<td>1.6.3.1.1.7.7.0</td>
<td>chOP2.</td>
<td>read-write</td>
<td>Channel operand 2</td>
<td>INTEGER [none(0), multiplication(1), division(2), sum(3), subtract(4)]</td>
</tr>
<tr>
<td>1.6.3.1.1.8.7.0</td>
<td>chCoef1.</td>
<td>read-write</td>
<td>Channel coefficient 1 x1000 in Integer format</td>
<td>Integer32</td>
</tr>
<tr>
<td>1.6.3.1.1.9.7.0</td>
<td>chOP3.</td>
<td>read-write</td>
<td>Channel operand 3</td>
<td>INTEGER [none(0), multiplication(1), division(2), sum(3), subtract(4)]</td>
</tr>
<tr>
<td>1.6.3.1.1.10.7.0</td>
<td>chCoef2.</td>
<td>read-write</td>
<td>Channel coefficient 2 x1000 in Integer format</td>
<td>Integer32</td>
</tr>
<tr>
<td>1.6.3.1.1.11.7.0</td>
<td>chUnit.</td>
<td>read-write</td>
<td>Channel unit</td>
<td>DisplayString</td>
</tr>
<tr>
<td>1.6.3.1.1.12.7.0</td>
<td>chCumulInitValue.</td>
<td>read-write</td>
<td>Channel cumulative initial value</td>
<td>Integer32</td>
</tr>
<tr>
<td>1.6.3.1.1.13.7.0</td>
<td>chValue.</td>
<td>read-only</td>
<td>Channel value x1000 in Integer format</td>
<td>Integer32</td>
</tr>
<tr>
<td>1.6.3.1.1.14.7.0</td>
<td>chCounter.</td>
<td>read-only</td>
<td>Channel as 32-bit counter</td>
<td>Counter32</td>
</tr>
<tr>
<td>1.6.3.1.1.15.7.0</td>
<td>chAlarmStatus.</td>
<td>read-only</td>
<td>Channel alarm status</td>
<td>INTEGER [undefined(0), normal(1), indeterminate(2), warning(3), minor(4), major(5), critical(6)]</td>
</tr>
</tbody>
</table>
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

<table>
<thead>
<tr>
<th>OID</th>
<th>Name</th>
<th>Access</th>
<th>Description</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.6.3.2.1.2.7.0</td>
<td>alDescription.7</td>
<td>read-write</td>
<td>Alarm description</td>
<td>INTEGER {none(0), v01(1), v02(2), v03(3), v04(4), v05(5), v06(6),</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>v07(7), v08(8), v09(9), v10(10), v11(11), v12(12), v13(13), v14(14),</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>v15(15), v16(16), v17(17), v18(18), v19(19), v20(20), v21(21), v22(22),</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>v23(23), v24(24)}</td>
</tr>
<tr>
<td>1.6.3.2.1.3.7.0</td>
<td>alCond1Channel.7</td>
<td>read-write</td>
<td>Alarm condition 1 channel</td>
<td>INTEGER {none(0), v01(1), v02(2), v03(3), v04(4), v05(5), v06(6),</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>v07(7), v08(8), v09(9), v10(10), v11(11), v12(12), v13(13), v14(14),</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>v15(15), v16(16), v17(17), v18(18), v19(19), v20(20), v21(21), v22(22),</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>v23(23), v24(24)}</td>
</tr>
<tr>
<td>1.6.3.2.1.4.7.0</td>
<td>alCond1Operand.7</td>
<td>read-write</td>
<td>Alarm condition 1 operand</td>
<td>INTEGER {larger(1), less(2)}</td>
</tr>
<tr>
<td>1.6.3.2.1.5.7.0</td>
<td>alCond1Limit.7</td>
<td>read-write</td>
<td>Alarm condition 1 limit x1000 in Integer format</td>
<td>Integer32</td>
</tr>
<tr>
<td>1.6.3.2.1.6.7.0</td>
<td>alCond1Hys.7</td>
<td>read-write</td>
<td>Alarm condition 1 hysteresis x1000 in Integer format</td>
<td>Integer32</td>
</tr>
<tr>
<td>1.6.3.2.1.7.7.0</td>
<td>alCond1AlarmState.7</td>
<td>read-write</td>
<td>Alarm condition 1 discrete alarm state</td>
<td>INTEGER {open(0), closed(1)}</td>
</tr>
<tr>
<td>1.6.3.2.1.8.7.0</td>
<td>alCondLogic.7</td>
<td>read-write</td>
<td>Alarm conditions logic</td>
<td>INTEGER {none(0), and(1), or(2)}</td>
</tr>
<tr>
<td>1.6.3.2.1.9.7.0</td>
<td>alCond2Channel.7</td>
<td>read-write</td>
<td>Alarm condition 2 channel</td>
<td>INTEGER {none(0), v01(1), v02(2), v03(3), v04(4), v05(5), v06(6),</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>v07(7), v08(8), v09(9), v10(10), v11(11), v12(12), v13(13), v14(14),</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>v15(15), v16(16), v17(17), v18(18), v19(19), v20(20), v21(21), v22(22),</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>v23(23), v24(24)}</td>
</tr>
<tr>
<td>1.6.3.2.1.10.7.0</td>
<td>alCond2Operand.7</td>
<td>read-write</td>
<td>Alarm condition 2 operand</td>
<td>INTEGER {larger(1), less(2)}</td>
</tr>
<tr>
<td>1.6.3.2.1.11.7.0</td>
<td>alCond2Limit.7</td>
<td>read-write</td>
<td>Alarm condition 2 limit x1000 in Integer format</td>
<td>Integer32</td>
</tr>
<tr>
<td>1.6.3.2.1.12.7.0</td>
<td>alCond2Hys.7</td>
<td>read-write</td>
<td>Alarm condition 2 hysteresis x1000 in Integer format</td>
<td>Integer32</td>
</tr>
<tr>
<td>1.6.3.2.1.13.7.0</td>
<td>alCond2AlarmState.7</td>
<td>read-write</td>
<td>Alarm condition 2 discrete alarm state</td>
<td>INTEGER {open(0), closed(1)}</td>
</tr>
<tr>
<td>1.6.3.2.1.14.7.0</td>
<td>alType.7</td>
<td>read-write</td>
<td>Alarm type</td>
<td>INTEGER {warning(3), minor(4), major(5), critical(6)}</td>
</tr>
<tr>
<td>1.6.3.2.1.15.7.0</td>
<td>alAssigned.7</td>
<td>read-write</td>
<td>Alarm assigned to</td>
<td>INTEGER {none(0), v01(1), v02(2), v03(3), v04(4), v05(5), v06(6),</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>v07(7), v08(8), v09(9), v10(10), v11(11), v12(12), v13(13), v14(14),</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>v15(15), v16(16), v17(17), v18(18), v19(19), v20(20), v21(21), v22(22),</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>v23(23), v24(24)}</td>
</tr>
<tr>
<td>1.6.3.2.1.16.7.0</td>
<td>alActionDelay.7</td>
<td>read-write</td>
<td>Alarm action delay</td>
<td>Integer32</td>
</tr>
<tr>
<td>1.6.3.2.1.17.7.0</td>
<td>alActionOnReturn.7</td>
<td>read-write</td>
<td>Alarm action on return</td>
<td>INTEGER {no(0), yes(1)}</td>
</tr>
<tr>
<td>1.6.3.2.1.18.7.0</td>
<td>alAction1.7</td>
<td>read-write</td>
<td>Alarm action 1</td>
<td>INTEGER {none(0), trapcond1(1), trapcond2(2), trapcond1and2(3),</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>postiostate(4)}</td>
</tr>
<tr>
<td>1.6.3.2.1.19.7.0</td>
<td>alAction2.7</td>
<td>read-write</td>
<td>Alarm action 2</td>
<td>INTEGER {none(0), trapcond1(1), trapcond2(2), trapcond1and2(3),</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>postiostate(4)}</td>
</tr>
</tbody>
</table>
monitorNcontrol
replace “!” with “1.3.6.1.4.1.38783” in the table below

<table>
<thead>
<tr>
<th>OID</th>
<th>Name</th>
<th>Access</th>
<th>Description</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.6.3.3.0</td>
<td>configurationSaved</td>
<td>read-write</td>
<td>Configuration save status SAVED/UNSAVED</td>
<td>INTEGER { unsaved(0), saved(1) }</td>
</tr>
<tr>
<td>1.6.3.4.0</td>
<td>restartDevice</td>
<td>read-write</td>
<td>Restart Device</td>
<td>INTEGER { cancel(0), restart(1) }</td>
</tr>
<tr>
<td>1.6.3.5.0</td>
<td>hardwareErr</td>
<td>read-only</td>
<td>Hardware Error</td>
<td>INTEGER { noErr(0), hwErr(1) }</td>
</tr>
</tbody>
</table>

9.2. HTTP API

9.2.1. HTTP Post

TCW260 can execute HTTP Post to upload XML/JSON file to a dedicated server.

This functionality is very useful if the controller is behind the router without a public IP address or the user don’t have access to router configuration. The server should have a public IP address.

The typical monitoring application is shown in the picture below:

HTTP post can be sent periodically or periodically plus on an alarm condition. As an answer, the server can send HTTP Get with the appropriate command.

To test HTTP Post follow the steps below:

- Save following code like post.php:

```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)."\n";
```
Copy the post.php file on a public web server with PHP support. To verify that the script is working properly, you can type the URL (for example www.yourserverURL.com/post.php) in your web browser. If all is OK, a web page with “The PHP script is working!” will be shown.

- Set the controller to send an HTTP POST to your web server. Enter the address (yourserverURL.com/post.php) in the URL field. Click on “Test HTTP Post” button.
- If the HTTP POST is received and processed, “OK” will be shown close to the button. Along with this, an XML file will be created in the same directory, where post.php is located. The file name will contain time information and looks like 20190420103318_status.xml.

9.2.2. HTTP Get

HTTP Get can be used to monitor TCW260 via XML or JSON files. The format is as follows:

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

See sections 8.2.4 XML file structure and 8.2.5 JSON file structure for details of files.

HTTP Get can be sent at any time to TCW260 if it is on the same network or it has appropriate routing.

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

9.2.2.1. Commands

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

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

Where:
- yyy is the command;
- xxx is the parameter.

Example:

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

9.2.2.2. HTTP GET authentication

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

<table>
<thead>
<tr>
<th>XML/HTTP API authentication</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>enabled</td>
<td><a href="http://device.ip.address/status.xml?a=uuuu:pppp">http://device.ip.address/status.xml?a=uuuu:pppp</a></td>
</tr>
<tr>
<td>disabled</td>
<td><a href="http://device.ip.address/status.xml">http://device.ip.address/status.xml</a></td>
</tr>
</tbody>
</table>

Example:

http://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
### List of HTTP API commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>pmet=x</td>
<td>Enable/disable HTTP Post - 0 disable, 2 enable</td>
</tr>
<tr>
<td>dataf=x</td>
<td>Data format XML/JSON for HTTP Post – 0 XML, 1 JSON</td>
</tr>
<tr>
<td>purl1=yyy</td>
<td>URL for HTTP Post to Server 1, where yyy is a full path to php file. Example: purl=212.25.45.120:30181/xampp/test/posttest1.php</td>
</tr>
<tr>
<td>purl2=yyy</td>
<td>URL for HTTP Post to Server 1, where yyy is a full path to php file. Example: purl=212.25.45.120:30181/xampp/test/posttest2.php</td>
</tr>
<tr>
<td>purl3=yyy</td>
<td>URL for HTTP Post to Server 1, where yyy is a full path to php file. Example: purl=212.25.45.120:30181/xampp/test/posttest3.php</td>
</tr>
<tr>
<td>pper=x</td>
<td>HTTP Post period in seconds (x is between 10 and 14400)</td>
</tr>
<tr>
<td>dk=xxx</td>
<td>HTTP Post key – xxx is up to 17 characters</td>
</tr>
<tr>
<td>pans=x</td>
<td>HTTP Post process answer, where x=0 - No, x=1 – Server1, x=2 – Server2, x=3 – Server 3</td>
</tr>
<tr>
<td>FIN</td>
<td>Terminate session (it makes sense only for HTTP Post)</td>
</tr>
</tbody>
</table>
9.2.4. XML file structure

```xml
<Monitor>
  <DeviceInfo>
    <DeviceName>TCW260</DeviceName>
    <HostName>TCW260</HostName>
    <ID>54:10:ec:4f:59:f6</ID>
    <FwVer>TCW260-v1.004rc8-197</FwVer>
    <MnfInfo>www.teracomsystems.com</MnfInfo>
    <SysContact>info@teracomsystems.com</SysContact>
    <SysName>TCW260</SysName>
    <SysLocation>Location</SysLocation>
  </DeviceInfo>
  <CH1>
    <type>0</type>
    <description>Temperature</description>
    <value>24.386</value>
    <unit>°C</unit>
    <alarmbin>4</alarmbin>
    <alarm>Minor</alarm>
    <selch>3</selch>
  </CH1>
  <CH2>
    <type>0</type>
    <description>Humidity</description>
    <value>51.323</value>
    <unit>RH</unit>
    <alarmbin>1</alarmbin>
    <alarm>Normal</alarm>
    <selch>4</selch>
  </CH2>
  <CH3>
    <type>2</type>
    <description>Digital Input 1</description>
    <value>OPEN</value>
    <unit/>
    <alarmbin>1</alarmbin>
    <alarm>Normal</alarm>
    <selch>33</selch>
  </CH3>
  <CH4>
    <type>0</type>
    <description>V04</description>
    <value>-</value>
    <unit/>
    <alarmbin>0</alarmbin>
    <alarm/>
    <selch>0</selch>
  </CH4>
  <CH5>
    <type>0</type>
    <description>V05-Voltage</description>
    <value>0.000</value>
    <unit>V</unit>
    <alarmbin>1</alarmbin>
    <alarm>Normal</alarm>
    <selch>27</selch>
  </CH5>
  <CH6>
    <type>0</type>
    <description>V06-Current</description>
    <value>0.000</value>
    <unit>A</unit>
    <alarmbin>1</alarmbin>
    <alarm>Normal</alarm>
    <selch>28</selch>
  </CH6>
  <CH7>
    <type>1</type>
    <description>V07-Energy</description>
    <value>6587.396</value>
    <unit>kWh</unit>
    <alarmbin>0</alarmbin>
    <alarm/>
    <selch>27</selch>
  </CH7>
</Monitor>
```
<AL1>
<description>AL01-Temp.</description>
<alarmbin>4</alarmbin>
<alarm>Minor</alarm>
<assign>1</assign>
</AL1>
<AL2>
<description>AL02</description>
<alarmbin>1</alarmbin>
<alarm>Normal</alarm>
<assign>5</assign>
</AL2>
<AL3>
<description>AL03</description>
<alarmbin>1</alarmbin>
<alarm>Normal</alarm>
<assign>6</assign>
</AL3>
<AL4>
<description>AL04-DI1</description>
<alarmbin>1</alarmbin>
<alarm>Normal</alarm>
<assign>3</assign>
</AL4>
<AL5>
<description>AL05-Humidity</description>
<alarmbin>1</alarmbin>
<alarm>Normal</alarm>
<assign>2</assign>
</AL5>
<AL6>
<description>AL06</description>
<alarmbin>1</alarmbin>
<alarm>Normal</alarm>
<assign>3</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>
<HTTPPost>
<Key/>
<PostPeriod>300</PostPeriod>
</HTTPPost>
<Sys>
<hwerr/>
<HighAlarmbin>4</HighAlarmbin>
</Sys>
Where:

<CH1>…<CH24> - channels;
<AL1>…<AL24> - alarms;
<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

9.2.5. JSON file structure

```json
{
    "Monitor": {
        "DeviceInfo": {
            "DeviceName": "TCW260",
            "HostName": "TCW260",
            "ID": "54:10:ec:4f:59:f6",
            "FwVer": "TCW260+v1.004rc8-197",
            "MnfInfo": "www.teracomsystems.com",
            "SysContact": "info@teracomsystems.com",
            "SysName": "TCW260",
            "SysLocation": "Location"
        },
        "CH": {
            "CH1": {
                "type": "0",
                "description": "Temperature",
                "value": "24.268",
                "unit": "°C",
                "alarmbin": "4",
                "alarm": "Minor",
                "selch": "3"
            },
            "CH2": {
                "type": "0",
                "description": "Humidity",
                "value": "52.490",
                "unit": "RH",
                "alarmbin": "1",
                "alarm": "Normal",
                "selch": "4"
            },
            "CH3": {
                "type": "2",
                "description": "Digital Input 1",
                "value": "OPEN",
                "unit": "",
                "alarmbin": "1",
                "alarm": "Normal",
                "selch": "33"
            },
            "CH4": {
                "type": "0",
                "description": "V04",
                "value": "---",
                "unit": "",
                "alarmbin": "0",
                "alarm": "",
                "selch": "0"
            },
            "CH5": {
                "type": "0",
                "description": "V05-Voltage",
                "value": "0.000",
                "unit": "V",
                "alarmbin": "1",
                "alarm": "",
                "selch": "0"
            }
        }
    }
}
```


```json
"alarm": "Normal",
"selch": "27"
}

"CH6": {
  "type": "0",
  "description": "V06-Current",
  "value": "0.000",
  "unit": "A",
  "alarmbin": "1",
  "alarm": "Normal",
  "selch": "27"
}

"CH7": {
  "type": "1",
  "description": "V07-Energy",
  "value": "6587.396",
  "unit": "kWh",
  "alarmbin": "0",
  "alarm": "",
  "selch": "27"
}

"CH8": {
  "type": "0",
  "description": "V08",
  "value": "---",
  "unit": "",
  "alarmbin": "0",
  "alarm": "",
  "selch": "0"
}

"CH9": {
  "type": "0",
  "description": "V09",
  "value": "---",
  "unit": "",
  "alarmbin": "0",
  "alarm": "",
  "selch": "0"
}

"CH10": {
  "type": "0",
  "description": "V10",
  "value": "---",
  "unit": "",
  "alarmbin": "0",
  "alarm": "",
  "selch": "0"
}

"CH11": {
  "type": "0",
  "description": "V11",
  "value": "---",
  "unit": "",
  "alarmbin": "0",
  "alarm": "",
  "selch": "0"
}

"CH12": {
  "type": "0",
  "description": "V12",
  "value": "---",
  "unit": "",
  "alarmbin": "0",
  "alarm": "",
  "selch": "0"
}

"CH13": {
  "type": "0",
  "description": "V13",
  "value": "---",
  "unit": "",
  "alarmbin": "0",
  "alarm": "",
  "selch": "0"
}

"CH14": {
  "type": "0",
```

---

TCW260_R1.5 – July 2020
Page 37

---
"description": "V14",
"value": "---",
"unit": "",
"alarmbin": "0",
"alarm": "",
"selch": "0"
},

"CH15": {
"type": "0",
"description": "V15",
"value": "---",
"unit": "",
"alarmbin": "0",
"alarm": "",
"selch": "0"
},

"CH16": {
"type": "0",
"description": "V16",
"value": "---",
"unit": "",
"alarmbin": "0",
"alarm": "",
"selch": "0"
},

"CH17": {
"type": "0",
"description": "V17",
"value": "---",
"unit": "",
"alarmbin": "0",
"alarm": "",
"selch": "0"
},

"CH18": {
"type": "0",
"description": "V18",
"value": "---",
"unit": "",
"alarmbin": "0",
"alarm": "",
"selch": "0"
},

"CH19": {
"type": "0",
"description": "V19",
"value": "---",
"unit": "",
"alarmbin": "0",
"alarm": "",
"selch": "0"
},

"CH20": {
"type": "0",
"description": "V20",
"value": "---",
"unit": "",
"alarmbin": "0",
"alarm": "",
"selch": "0"
},

"CH21": {
"type": "0",
"description": "V21",
"value": "---",
"unit": "",
"alarmbin": "0",
"alarm": "",
"selch": "0"
},

"CH22": {
"type": "0",
"description": "V22",
"value": "---",
"unit": "",
"alarmbin": "0",
"alarm": ""
<table>
<thead>
<tr>
<th>CH23</th>
<th>Type: 0</th>
<th>Description: V23</th>
<th>Value: ---</th>
<th>Unit:</th>
<th>Alarm Bin: 0</th>
<th>Alarm:</th>
<th>Selected: 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>CH24</td>
<td>Type: 0</td>
<td>Description: V24</td>
<td>Value: ---</td>
<td>Unit:</td>
<td>Alarm Bin: 0</td>
<td>Alarm:</td>
<td>Selected: 0</td>
</tr>
<tr>
<td>AL1</td>
<td>Description: AL01-Temp</td>
<td>Alarm Bin: aa(0)</td>
<td>Alarm: Minor</td>
<td>Assign: 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AL2</td>
<td>Description: AL02</td>
<td>Alarm Bin: aa(1)</td>
<td>Alarm: Normal</td>
<td>Assign: 5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AL3</td>
<td>Description: AL03</td>
<td>Alarm Bin: aa(2)</td>
<td>Alarm: Normal</td>
<td>Assign: 6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AL4</td>
<td>Description: AL04-DI1</td>
<td>Alarm Bin: aa(3)</td>
<td>Alarm: Normal</td>
<td>Assign: 3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AL5</td>
<td>Description: AL05-Humidity</td>
<td>Alarm Bin: aa(4)</td>
<td>Alarm: Normal</td>
<td>Assign: 2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AL6</td>
<td>Description: AL06</td>
<td>Alarm Bin: aa(5)</td>
<td>Alarm: Normal</td>
<td>Assign: 3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AL7</td>
<td>Description: AL07</td>
<td>Alarm Bin: aa(6)</td>
<td>Alarm:</td>
<td>Assign: 0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AL8</td>
<td>Description: AL08</td>
<td>Alarm Bin: aa(7)</td>
<td>Alarm:</td>
<td>Assign: 0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AL9</td>
<td>Description: AL09</td>
<td>Alarm Bin: aa(8)</td>
<td>Alarm:</td>
<td>Assign: 0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AL10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
"description": "AL10",
"alarmbin": "aa(9)",
"alarm": "",
"assign": "0"
},
"AL11": {
"description": "AL11",
"alarmbin": "aa(10)",
"alarm": "",
"assign": "0"
},
"AL12": {
"description": "AL12",
"alarmbin": "aa(11)",
"alarm": "",
"assign": "0"
},
"AL13": {
"description": "AL13",
"alarmbin": "aa(12)",
"alarm": "",
"assign": "0"
},
"AL14": {
"description": "AL14",
"alarmbin": "aa(13)",
"alarm": "",
"assign": "0"
},
"AL15": {
"description": "AL15",
"alarmbin": "aa(14)",
"alarm": "",
"assign": "0"
},
"AL16": {
"description": "AL16",
"alarmbin": "aa(15)",
"alarm": "",
"assign": "0"
},
"AL17": {
"description": "AL17",
"alarmbin": "aa(16)",
"alarm": "",
"assign": "0"
},
"AL18": {
"description": "AL18",
"alarmbin": "aa(17)",
"alarm": "",
"assign": "0"
},
"AL19": {
"description": "AL19",
"alarmbin": "aa(18)",
"alarm": "",
"assign": "0"
},
"AL20": {
"description": "AL20",
"alarmbin": "aa(19)",
"alarm": "",
"assign": "0"
},
"AL21": {
"description": "AL21",
"alarmbin": "aa(20)",
"alarm": "",
"assign": "0"
},
"AL22": {
"description": "AL22",
"alarmbin": "aa(21)",
"alarm": "",
"assign": "0"
},
9.3. MODBUS TCP/IP

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

9.3.1. Codes and answers

9.3.1.1. Read Holding Registers (FC=03)

Request
This command is requesting the content of holding registers 400 – Channel 1 value..

03 0190 0002

03: The Function Code 3 (read Holding Registers)
0064: The Data Address of the first register requested (0190 hex = 400)
0002: The total number of registers requested. (read 2 registers each 2 bytes = 4 bytes)

Response

03 04 41 C3 B0 21

03: The Function Code 3 (read Channel 1 Holding Registers)
04: The number of data bytes to follow (2 registers x 2 bytes each = 4 bytes)
41C3 B021: 4 bytes value
All holding registers with float value are sent in big-endian.
In the example above, the value of 24.4610004 is sent.
Request
This command is requesting the content of holding registers 5000 – Channel 1 description.

**03 1388 0008**

- **03**: The Function Code 3 (read Holding Registers)
- **1388**: The Data Address of the first register requested (1388 hex = 5000)
- **0008**: The total number of registers requested (read 8 registers each 2 bytes = 16 bytes)

Response

**03 10 54 65 6D 70 65 72 61 74 75 72 65 00 00 00 00 00**

- **03**: The Function Code 3 (read Analog Output Holding Registers)
- **10**: The number of data bytes to follow (8 registers x 2 bytes each = 16 bytes)
- **54 65 6D 70 65 72 61 74 75 72 65 00 00 00 00 00**: 16 bytes value

All holding registers with strings are sent in big-endian. The answer is padded with 0. In the example above, the string “Temperature” is sent.

### 9.3.1.2. Exception codes

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

- **01**: Illegal function
  - The function code received in the query is not an allowable action for the controller.

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

### 9.3.2. Address table

<table>
<thead>
<tr>
<th>Parameter</th>
<th>FC</th>
<th>Address (Decimal)</th>
<th>Data size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read Channel 1</td>
<td>03</td>
<td>400</td>
<td>32-bit Float</td>
</tr>
<tr>
<td>Read Channel 2</td>
<td>03</td>
<td>402</td>
<td>32-bit Float</td>
</tr>
<tr>
<td>Read Channel 3</td>
<td>03</td>
<td>404</td>
<td>32-bit Float</td>
</tr>
<tr>
<td>Read Channel 4</td>
<td>03</td>
<td>406</td>
<td>32-bit Float</td>
</tr>
<tr>
<td>Read Channel 5</td>
<td>03</td>
<td>408</td>
<td>32-bit Float</td>
</tr>
<tr>
<td>Read Channel 6</td>
<td>03</td>
<td>410</td>
<td>32-bit Float</td>
</tr>
<tr>
<td>Read Channel 8</td>
<td>03</td>
<td>412</td>
<td>32-bit Float</td>
</tr>
<tr>
<td>Read Channel 9</td>
<td>03</td>
<td>414</td>
<td>32-bit Float</td>
</tr>
<tr>
<td>Read Channel 10</td>
<td>03</td>
<td>416</td>
<td>32-bit Float</td>
</tr>
<tr>
<td>Read Channel 11</td>
<td>03</td>
<td>418</td>
<td>32-bit Float</td>
</tr>
<tr>
<td>Read Channel 12</td>
<td>03</td>
<td>420</td>
<td>32-bit Float</td>
</tr>
<tr>
<td>Read Channel 13</td>
<td>03</td>
<td>422</td>
<td>32-bit Float</td>
</tr>
<tr>
<td>Read Channel 14</td>
<td>03</td>
<td>424</td>
<td>32-bit Float</td>
</tr>
<tr>
<td>Read Channel 15</td>
<td>03</td>
<td>426</td>
<td>32-bit Float</td>
</tr>
<tr>
<td>Read Channel 16</td>
<td>03</td>
<td>428</td>
<td>32-bit Float</td>
</tr>
<tr>
<td>Read Channel</td>
<td>Dimension</td>
<td>Value</td>
<td>Type</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------</td>
<td>-------</td>
<td>----------------</td>
</tr>
<tr>
<td>17</td>
<td>03</td>
<td>430</td>
<td>32-bit Float</td>
</tr>
<tr>
<td>18</td>
<td>03</td>
<td>432</td>
<td>32-bit Float</td>
</tr>
<tr>
<td>19</td>
<td>03</td>
<td>434</td>
<td>32-bit Float</td>
</tr>
<tr>
<td>20</td>
<td>03</td>
<td>436</td>
<td>32-bit Float</td>
</tr>
<tr>
<td>21</td>
<td>03</td>
<td>438</td>
<td>32-bit Float</td>
</tr>
<tr>
<td>22</td>
<td>03</td>
<td>440</td>
<td>32-bit Float</td>
</tr>
<tr>
<td>23</td>
<td>03</td>
<td>442</td>
<td>32-bit Float</td>
</tr>
<tr>
<td>24</td>
<td>03</td>
<td>444</td>
<td>32-bit Float</td>
</tr>
</tbody>
</table>

Channels 1-24, type Counter

<table>
<thead>
<tr>
<th>Read Channel</th>
<th>Dimension</th>
<th>Value</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>03</td>
<td>500</td>
<td>32-bit unsigned int</td>
</tr>
<tr>
<td>2</td>
<td>03</td>
<td>502</td>
<td>32-bit unsigned int</td>
</tr>
<tr>
<td>3</td>
<td>03</td>
<td>504</td>
<td>32-bit unsigned int</td>
</tr>
<tr>
<td>4</td>
<td>03</td>
<td>506</td>
<td>32-bit unsigned int</td>
</tr>
<tr>
<td>5</td>
<td>03</td>
<td>508</td>
<td>32-bit unsigned int</td>
</tr>
<tr>
<td>6</td>
<td>03</td>
<td>510</td>
<td>32-bit unsigned int</td>
</tr>
<tr>
<td>7</td>
<td>03</td>
<td>512</td>
<td>32-bit unsigned int</td>
</tr>
<tr>
<td>8</td>
<td>03</td>
<td>514</td>
<td>32-bit unsigned int</td>
</tr>
<tr>
<td>9</td>
<td>03</td>
<td>516</td>
<td>32-bit unsigned int</td>
</tr>
<tr>
<td>10</td>
<td>03</td>
<td>518</td>
<td>32-bit unsigned int</td>
</tr>
<tr>
<td>11</td>
<td>03</td>
<td>520</td>
<td>32-bit unsigned int</td>
</tr>
<tr>
<td>12</td>
<td>03</td>
<td>522</td>
<td>32-bit unsigned int</td>
</tr>
<tr>
<td>13</td>
<td>03</td>
<td>524</td>
<td>32-bit unsigned int</td>
</tr>
<tr>
<td>14</td>
<td>03</td>
<td>526</td>
<td>32-bit unsigned int</td>
</tr>
<tr>
<td>15</td>
<td>03</td>
<td>528</td>
<td>32-bit unsigned int</td>
</tr>
<tr>
<td>16</td>
<td>03</td>
<td>530</td>
<td>32-bit unsigned int</td>
</tr>
<tr>
<td>17</td>
<td>03</td>
<td>532</td>
<td>32-bit unsigned int</td>
</tr>
<tr>
<td>18</td>
<td>03</td>
<td>534</td>
<td>32-bit unsigned int</td>
</tr>
<tr>
<td>19</td>
<td>03</td>
<td>536</td>
<td>32-bit unsigned int</td>
</tr>
<tr>
<td>20</td>
<td>03</td>
<td>538</td>
<td>32-bit unsigned int</td>
</tr>
<tr>
<td>21</td>
<td>03</td>
<td>540</td>
<td>32-bit unsigned int</td>
</tr>
<tr>
<td>22</td>
<td>03</td>
<td>542</td>
<td>32-bit unsigned int</td>
</tr>
<tr>
<td>23</td>
<td>03</td>
<td>544</td>
<td>32-bit unsigned int</td>
</tr>
<tr>
<td>24</td>
<td>03</td>
<td>1000</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>1 Dimension</td>
<td>03</td>
<td>1008</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>2 Dimension</td>
<td>03</td>
<td>1016</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>3 Dimension</td>
<td>03</td>
<td>1024</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>4 Dimension</td>
<td>03</td>
<td>1032</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>5 Dimension</td>
<td>03</td>
<td>1040</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>6 Dimension</td>
<td>03</td>
<td>1048</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>7 Dimension</td>
<td>03</td>
<td>1056</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>8 Dimension</td>
<td>03</td>
<td>1064</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>9 Dimension</td>
<td>03</td>
<td>1072</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>10 Dimension</td>
<td>03</td>
<td>1080</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>11 Dimension</td>
<td>03</td>
<td>1088</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>12 Dimension</td>
<td>03</td>
<td>1096</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>13 Dimension</td>
<td>03</td>
<td>1104</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>14 Dimension</td>
<td>03</td>
<td>1112</td>
<td>16 bytes UTF-8</td>
</tr>
</tbody>
</table>
### Data Logger

The logger utilizes circular buffer in FLASH memory. When it is full, the new data overwrites the oldest one. In this manner FLASH memory stores full log all the time. There isn’t a command to clear the log. A copy of the full log is always available for download.

The number of records depends on how long descriptions and what kind of characters are used. In the worst case (15 bytes description with characters from the highest part of UTF-8) a number of records are about 52371. This number is enough for 36 days with records every 1 minute.

In most of the cases, the data logger can keep 71400 records, which is enough for 49 days with records every 1 minute.

The new records can be periodically uploaded as a file to a dedicated HTTP server in time intervals – 1, 2, 3, 4, 6, 8, 12 and 24 hours. The file is in CSV format. A semicolon is used as a delimiter.

The first row of the log file is always header. All rows, including the header, start with record ID and time stamp.

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

<table>
<thead>
<tr>
<th>Read Channel 15 Dimension</th>
<th>03</th>
<th>1112</th>
<th>16 bytes UTF-8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read Channel 16 Dimension</td>
<td>03</td>
<td>1120</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 17 Dimension</td>
<td>03</td>
<td>1128</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 18 Dimension</td>
<td>03</td>
<td>1136</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 19 Dimension</td>
<td>03</td>
<td>1144</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 20 Dimension</td>
<td>03</td>
<td>1152</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 21 Dimension</td>
<td>03</td>
<td>1160</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 22 Dimension</td>
<td>03</td>
<td>1168</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 23 Dimension</td>
<td>03</td>
<td>1176</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 24 Dimension</td>
<td>03</td>
<td>1184</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 1 Description</td>
<td>03</td>
<td>5000</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 2 Description</td>
<td>03</td>
<td>5008</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 3 Description</td>
<td>03</td>
<td>5016</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 4 Description</td>
<td>03</td>
<td>5024</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 5 Description</td>
<td>03</td>
<td>5032</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 6 Description</td>
<td>03</td>
<td>5040</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 7 Description</td>
<td>03</td>
<td>5048</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 8 Description</td>
<td>03</td>
<td>5056</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 9 Description</td>
<td>03</td>
<td>5064</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 10 Description</td>
<td>03</td>
<td>5072</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 11 Description</td>
<td>03</td>
<td>5080</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 12 Description</td>
<td>03</td>
<td>5088</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 13 Description</td>
<td>03</td>
<td>5096</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 14 Description</td>
<td>03</td>
<td>5104</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 15 Description</td>
<td>03</td>
<td>5112</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 16 Description</td>
<td>03</td>
<td>5120</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 17 Description</td>
<td>03</td>
<td>5128</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 18 Description</td>
<td>03</td>
<td>5136</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 19 Description</td>
<td>03</td>
<td>5144</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 20 Description</td>
<td>03</td>
<td>5152</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 21 Description</td>
<td>03</td>
<td>5160</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 22 Description</td>
<td>03</td>
<td>5168</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 23 Description</td>
<td>03</td>
<td>5176</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>Read Channel 24 Description</td>
<td>03</td>
<td>5184</td>
<td>16 bytes UTF-8</td>
</tr>
<tr>
<td>ID</td>
<td>Time</td>
<td>Type of record</td>
<td>Channels - values/units</td>
</tr>
<tr>
<td>----</td>
<td>------</td>
<td>----------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>ID</td>
<td>32-bit unique number for every row (record).</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Time</td>
<td>a time stamp of record, in format dd.mm.yyyy, hh:mm:ss.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Type of record</td>
<td>following types of records are available:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;Time&quot;</td>
<td>for periodical record;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;Event&quot;</td>
<td>for record initiated by alarm condition;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;Type&quot;</td>
<td>for header record;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;Start&quot;</td>
<td>after power-up condition;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;Restart&quot;</td>
<td>after reset condition;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;Power Down&quot;</td>
<td>after power-down condition;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;Bad&quot;</td>
<td>for a problematic record</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Channels - values/units</td>
<td>Channels 1 to 24 values/units</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Channels - states/units</td>
<td>Channels 1 to 24 states/units</td>
<td>For channel types General, Cumulative and Discrete following states are available:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 – “Undefined”</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 – “Normal”</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2 – “Indeterminate”</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3 – “Warning”</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4 – “Minor”</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>5 – “Major”</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>6 – “Critical”</td>
<td></td>
<td></td>
</tr>
<tr>
<td>For channel type Counter following states are available:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>8 – “Undefined”</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>9 – “Normal”</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>10 – “Indeterminate”</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>11 – “Warning”</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>12 – “Minor”</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>13 – “Major”</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>14 – “Critical”</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alarms - values/descriptions</td>
<td>Alarms 1 to 24 values/descriptions</td>
<td>Following alarm values are available:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 – “Undefined”</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 – “Normal”</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2 – “Indeterminate”</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3 – “Warning”</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4 – “Minor”</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>5 – “Major”</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>6 – “Critical”</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

An example of the log file /fragment channels - values/units/:
ID;Time;Type;Ch1/°C;Ch2;Ch3;Ch4;Ch5;Ch6;Ch7;Ch8;Ch9;Ch10;Ch11;Ch12;Ch13;Ch14;Ch15;Ch16;Ch17;Ch18;Ch19;Ch20;Ch21;Ch22;Ch23;Ch24;...
25114;14.05.2019,16:49:49;Time;25.319;1.000;118.833;229.877;0.000;6587.396;;;;;;;;;;;;; ;;;;;;….

An example of the log file /fragment channels - states/units/:
ID;Time;Type;….Ch1/°C;Ch2;Ch3;Ch4;Ch5;Ch6;Ch7;Ch8;Ch9;Ch10;Ch11;Ch12;Ch13;Ch14;Ch15;Ch16;Ch17;Ch18;Ch19;Ch20;Ch21;Ch22;Ch23;Ch24;...
25114;14.05.2019,16:49:49;Time;….1;1;3;6;10;11;;;;;;;;;;;;;;;;;;;….

An example of the log file /fragment alarms - values/descriptions/:
ID;Time;Type;…AL1/AL1-temp;AL2;AL3;AL4;AL5;AL6;AL7;AL8;AL9;AL10;AL11;AL12;AL13;AL14;AL15;AL16;AL17;AL18;AL19;AL20;AL21;AL22;AL23;AL24;...
25114;14.05.2019,16:49:49;Time;….1;1;3;6;10;11;;;;;;;;;;;;;;;;;;...
11. Factory default settings

TCW260 can be restored to its original factory default settings in 3 different ways.

11.1. Factory default from the WEB interface

If the button “Factory default” from Administration->Backup/Restore is pressed, all parameters return to factory default except Network settings.

11.2. Factory default with the reset button

If the reset button is pressed for more than 5 seconds, while the device is working, all Network settings go to factory default.

11.3. General factory default with the reset button

For factory default reset of all parameters following steps should be executed:

- Press and hold the RESET button, then turn on the power supply;
- Yellow LED shines and red LED blinks about 5 times on a second;
- After about 5 seconds red LED will turn off, the button can be released;
- Yellow LED flashes on 1 second and red LED shines – the device is in working mode, with factory default settings.

The factory default settings are:

<table>
<thead>
<tr>
<th>Setting</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Username</td>
<td>admin</td>
</tr>
<tr>
<td>Password</td>
<td>admin</td>
</tr>
<tr>
<td>IP Address</td>
<td>192.168.1.2</td>
</tr>
<tr>
<td>Subnet Mask</td>
<td>255.255.255.0</td>
</tr>
<tr>
<td>Default Gateway</td>
<td>192.168.1.1</td>
</tr>
<tr>
<td>SNMPConfiguration</td>
<td>disabled</td>
</tr>
<tr>
<td>readCommunity</td>
<td>public</td>
</tr>
<tr>
<td>writeCommunity</td>
<td>private</td>
</tr>
<tr>
<td>Analog inputs unit</td>
<td>voltage</td>
</tr>
<tr>
<td>Analog inputs multiplier</td>
<td>1.000</td>
</tr>
<tr>
<td>Analog inputs offset</td>
<td>0.000</td>
</tr>
<tr>
<td>Analog inputs mode</td>
<td>Voltage</td>
</tr>
<tr>
<td>Digital inputs mode</td>
<td>Open/Closed</td>
</tr>
</tbody>
</table>
12. Environment information

This equipment is intended for use in a Pollution Degree 2 environment, at altitudes up to 2000 meters. When the controller is a part of a system, the other elements of the system shall comply with the EMC requirements and shall be intended for use in the same ambient conditions.

13. Safety

This device must not be used for medical, life-saving purposes or for any purpose where its failure could cause serious injury or the loss of life.

To reduce the risk of fire, only flexible stranded wire, with cross section 0.5mm² or larger for wiring of digital and analog inputs and relay output of the device should be used.

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

There is a risk of overheating (damage) of the controller if recommended free spaces to adjacent devices are not ensured. The joint part with external component shall have space for attachment/removal of the cable after installation.

Teracom does not guarantee the successful operation of the product if the product was used under conditions deviating from the product specifications.

To ensure that the device works correctly follow the steps below:

- ensure that the device is installed correctly, refer this user manual;
- log into the devices via a browser program;
- make proper setup;
- go to SETUP->INPUTS and set up Digital input 1 as “Discrete OPEN/CLOSED”;
- short the “S01+ (Digital in 1)” and “S01- (SGND)”;
- in the field “Value for Digital input 1 proper value should be displayed;
- at the same time flashing “STS” led should indicate the 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.

14. 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 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.
Fig. 1

Fig. 2