EN

MQTT, SNMP, NTP

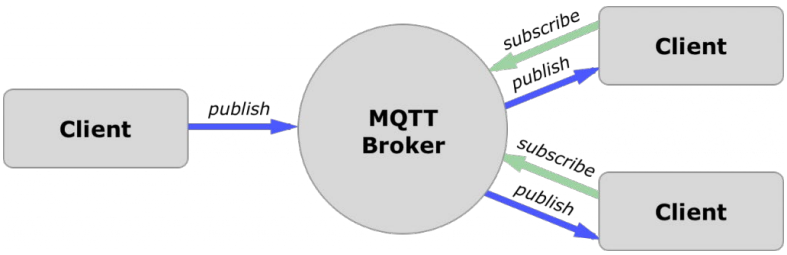
Getting Started

Contents

1. MQTT protocol

1.1 MQTT basics

The MQTT protocol (Message Queuing Telemetry Transport) is an event-driven protocol based on the TCP / IP stack that uses the Publisher / Subscriber network model. Currently, MQTT is the de facto standard for data exchange in Industrial Internet of Things (IIoT) applications.



Advantages of MQTT:

- low network traffic due to asynchronous data exchange
- compactness of message
- the ability to work in an unstable data transmission channel
- different levels of quality of service (QoS)

The MQTT architecture defines three types of devices on the network:

- publishers – clients that are data sources for subscribers
- subscribers – clients that needs data from publishers
- broker – a device (usually a PC with server software) that receives messages from publishers and sends them to subscribers

A device can be a publisher and a subscriber at the same time.

Published messages are organized in a hierarchy of ***topics***. When a publisher has a new data to distribute, it sends a message with the data under the particular topic to the connected broker. The broker distributes the message to any clients that have subscribed to that topic.

1.2 Topic names

The name of the topic is a UTF-8 encoded character string that the broker uses to filter messages for each connected client. Topic names are case sensitive.

The full topic name, which must be entered in the query, consists of one or more topic ***levels***, which are separated by forward slashes (***topic level separators***). The ***topic name*** is understood to be the code word on the last level.

When a client subscribes to a topic, it can subscribe to the exact topic of a published message or it can use wildcards to subscribe to multiple topics simultaneously. There are two kinds of wildcards: ***single-level*** (+) and ***multi-level*** (#) (see Example).

Structure of topic name:

Device_series/Device_name/Function/Node/Topic_name

where

- ***Device_series*** – MX210
- ***Device_name*** – name specified in the ***Device name*** parameter
- ***Function*** – GET (read the input and output values) or SET (write the output values)
- ***Node*** – I/O type (DI, DO, AI, AO)
- ***Topic_name*** – see ***Topic name*** column in Table 1.

Table 1. Topic levels

Device model	Function	Node	Topic name	Description	Format
202, 204, 212, 214, 221, 301, 302, 311, 312	GET	DI	MASK	Bitmask of digital inputs	UINT
202, 204, 212, 214, 301, 302, 311, 312	GET	DIn	COUNTER	Value of a counter or of an optional function	UINT
301, 302, 311, 312, 402, 403, 410	SET	DO	MASK	Bitmask of digital outputs	UINT
301, 302, 311, 312, 401, 402, 403, 410	GET	DO	STATE	Bitmask of digital outputs	UINT
311, 312, 410	GET	DO	DIAGNOSTICS	Diagnostics bitmask of digital outputs	UINT
101	GET	AIn	VALUE	Value of an analog input	REAL
501	SET, GET	AOn	VALUE_PERCENT	Value of analog output in %	REAL
			VALUE_PHYS	Value of analog output in mV or µA	REAL

Example:

MK210-311

1. Read the bitmask of digital inputs
   MX210/Device_name/GET/DI/MASK
   Received value: 15 (HIGH on inputs 1-4)
2. Write the bitmask of digital outputs
   MX210/Device_name/SET/DO/MASK
   New value: 15 (outputs 1-4 set)
3. Single-level wildcard usage
   MX210/Device_name/GET/+/COUNTER
   Received value: counter values of all digital inputs. The topic is equivalent to the group of topics:

   MX210/Device_name/GET/DI1/COUNTER
   MX210/Device_name/GET/DI2/COUNTER
   MX210/Device_name/GET/…/COUNTER
   MX210/Device_name/GET/DIn/COUNTER

4. Multi-level wildcard usage
   MX210/Device_name/GET/#
   Received value: all module parameters available for reading. The topic is equivalent to the group of topics:

   MX210/Device_name/GET/DI/MASK
   MX210/Device_name/GET/DI1/COUNTER
   MX210/Device_name/GET/DI2/COUNTER
   MX210/Device_name/GET/…/COUNTER
   MX210/Device_name/GET/DIn/COUNTER

1.3 Configuration in akYtecToolPro

The module of MX210 series supports the MQTT v3.1.1 protocol and can be used as a client. It can publish information about the status of its inputs and outputs and can be subscribed to topics which control its outputs.

To configure the MQTT parameters, open the node ***MQTT*** in the parameter tree.

NOTE
When using the MQTT protocol, it is recommended to set the parameter "Safe state timeout" (Modbus Slave group) to 0, since writing is usually event-driven and not cyclic in this case.

Table 2. MQTT parameters

Parameter	Description	Range	Default value	Access
Presence detection. Enable	If ***On***, the module publishes the message "Online" to the topic specified in the parameter ***Topic name*** after powering on. If no messages are received from the module, the broker publishes an “Offline” message in this topic.	***On*** / ***Off***	***Off***	RW
Presence detection. Topic name	Topic name used for presence detection.	-	***MQTT-status***	RW
Connect to broker	Set to ***On*** to establish connection	***On*** / ***Off***	***Off***	RW
User name	Used for device authentication on the broker side. If the values are not specified, the authentication is disabled.	-	-	RW
Password		-	-	RW
Device name	Device name used in the topic name (see 1.2 / Example)	-	-	RW
Broker address	Broker IP or URL. If the broker is located in an external network, check the correct values for the parameters ***Gateway*** and ***DNS*** (***Network*** group)	-	-	RW
Port	Port for broker	0…65535	1883	RW
Store last message	If ***On***, other clients subscribed to the module's topics will receive the latest messages from these topics.	***On*** / ***Off***	***Off***	RW
Publishing interval	Publishing interval in seconds	5…600	10	RW
Quality of service	***QoS0*** - at most once (without guarantee of delivery) ***QoS1*** - at least once (with guarantee of delivery) ***QoS2*** - exactly once (with guarantee of delivery and of no duplicate messages)	***QoS0*** / ***QoS1*** / ***QoS2***	***QoS0***	RW
Keep Alive	Keep Alive interval in seconds	0…600	0	RW
Status	Status of connection to broker	-	-	R

1.4 MQTT connection test

There are many ways to test the MQTT connection. We will show one of them. For test purposes we will use:

– ***Public MQTT Broker*** (online-tool)

  Link: https://www.hivemq.com/public-mqtt-broker/

– MQTT client ***MQTT.fx*** (will receive messages from the module)

  Download: *https://softblade.de/download/*

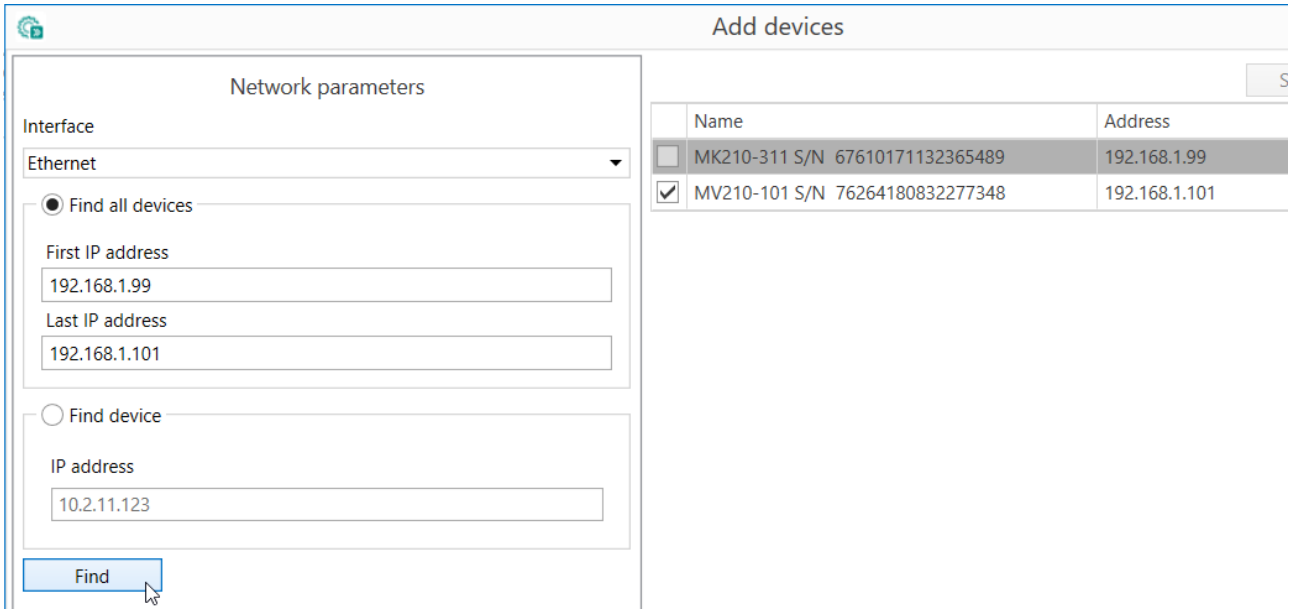
  Download ***MQTT.fx*** and install it on the PC.

Connection test:

1. Note the access information of ***Public MQTT Broker***.

You can access the broker at: Broker: broker.hivemq.com
TCP Port: 1883
Websocket Port: 8000

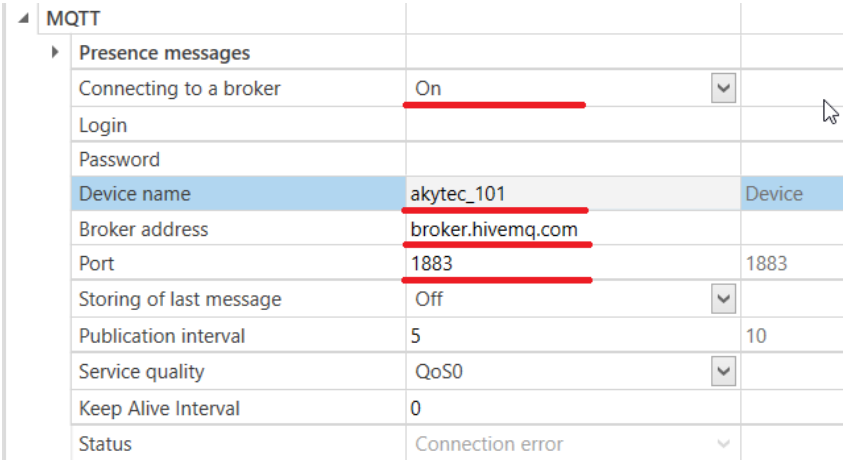
2. Connect the module to Ethernet and power it on. We will take MV210-101 (8 AI).
3. Start the configurator akYtecToolPro and add the device to the project.



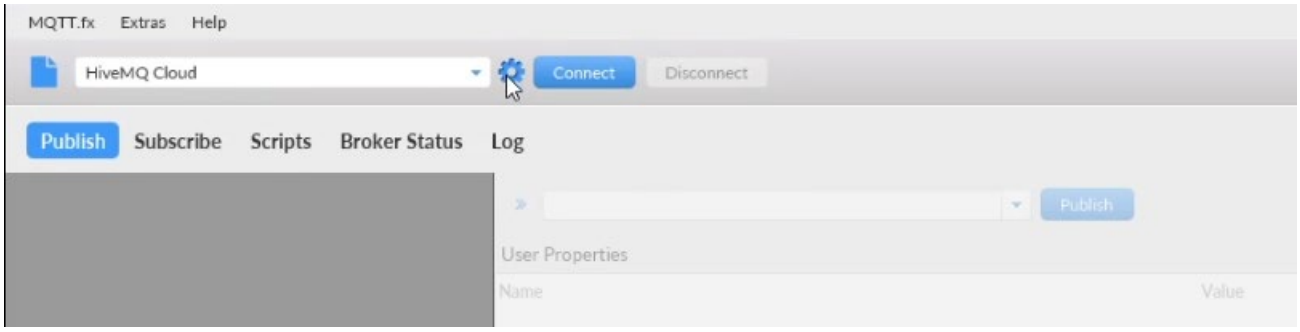
4. MV210-101: select the signal for one input and connect the respective sensor to it.

AI 1	
Input signal	Pt100 (a = 0.00385)
Curve offset	0
Curve slope	1
Upper limit	250
Lower limit	-50
Sampling time	3000
Filter bandwidth	10
Digital point offset	1
Filter time constant	3
AI 2	
Input signal	4-20 mA
Curve offset	0
Curve slope	1
Upper limit	250
Lower limit	-50
Sampling time	3000

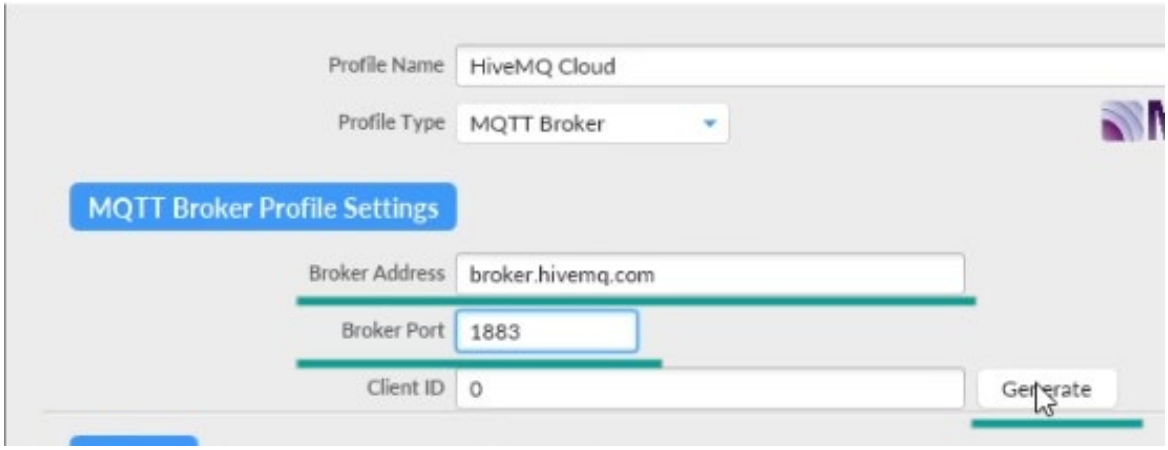
5. Enable the MQTT connection for the module and set the MQTT parameters: device name, broker address, port number. Click the item ***Write parameters*** to save the settings.



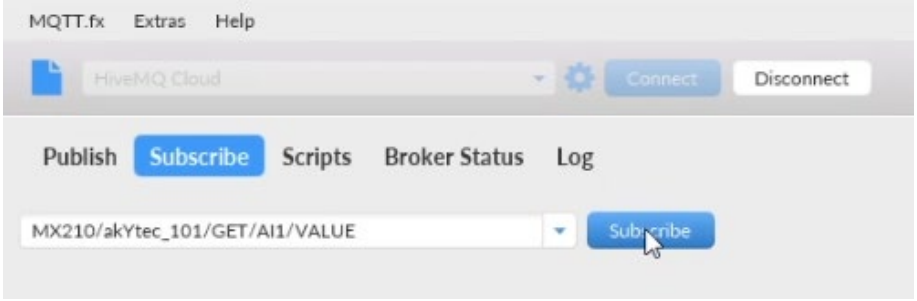
6. Start ***MQTT.fx*** and open ***Settings***.



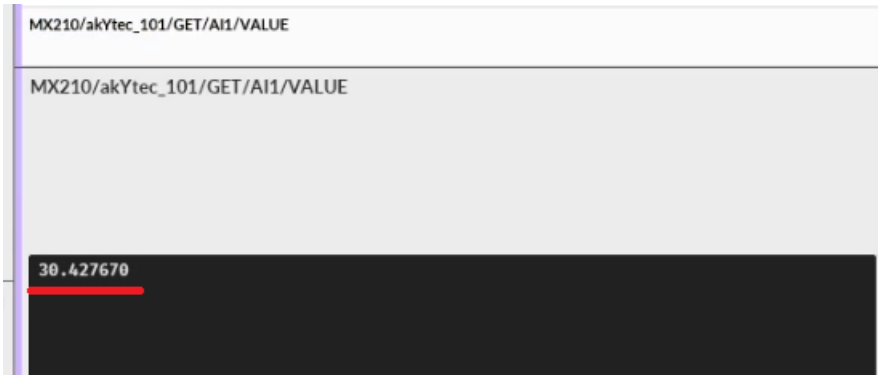
7. Enter ***Broker Address*** and ***Broker Port.***
8. Click ***Generate*** to generate Client ID and then ***OK*** to confirm.



9. When the dialog box is closed, click the button ***Connect***. The gray circle on the right turns green. The connection is established. The module publishes data on the broker and the client on PC can subscribe them.
10. Write the correct topic and click ***Subscribe***.



11. Now you can see the measured value on the input AI1 in the right pane…



…and in the akYtecToolPro.

2. SNMP protocol

2.1 SNMP basics

Simple Network Management Protocol (SNMP) is an Internet standard protocol for collecting and organizing information about managed devices on IP networks and for modifying that information to change device behavior.

An SNMP-managed network consists of three key components:

- Managed devices – network nodes with an SNMP interface that allows unidirectional (read-only) or bidirectional (read-write) access to node-specific information
- ***Agent*** – software which runs on managed devices
- Network management station (NMS) – software which runs on the ***manager*** (administrative computer)

Managers can read (GET) and write (SET) agent parameters. Agents can send messages (***traps***) to managers about parameter changes.

Management data is exposed in the form of variables on the managed device organized in a Management Information Base (MIB) as a hierarchical tree structure. Each variable (parameter) in MIB has a unique identifier OID (object identifier), represented as a sequence of decimal numbers separated by dots. SNMP requests use OID to retrieve the desired information.

All module parameters are available via SNMP protocol. The complete list of parameters is given in the User Guide in Table D.1 “Modbus registers”.

2.2 Configuration in akYtecToolPro

To configure the SNMP parameters of the device, connect the device to the PC with the running akYtecToolPro and add this to a project.

Table 3. SNMP parameters

Parameter	Description	Range	Default value	Access
Enable	Enable SNMP connection	***On*** / ***Off***	***Off***	RW
Read community	Community name for read access level	-	***public***	RW
Write community	Community name for read/write access level	-	***private***	RW
Trap IP address	IP address to which a trap will be sent in case of changing the mask of the digital inputs (modules with digital inputs only)	-	10.2.4.78	RW
Trap port	Number of the port to which traps will be sent	0…65535	162	RW
SNMP version	Protocol version	***SNMPv1*** / ***SNMPv2***	***SNMPv1***	RW

- Set the ***Enable*** parameter to ***On*** to enable the SNMP connection
- Set the ***Trap IP address*** parameter to the IP address of the manager (PC with NMS software)
- Set the ***Trap port***
- Select the ***SNMP version***

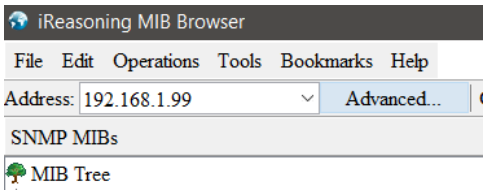
The module supports SNMPv1 and SNMPv2c protocol versions.

2.3 SNMP connection test

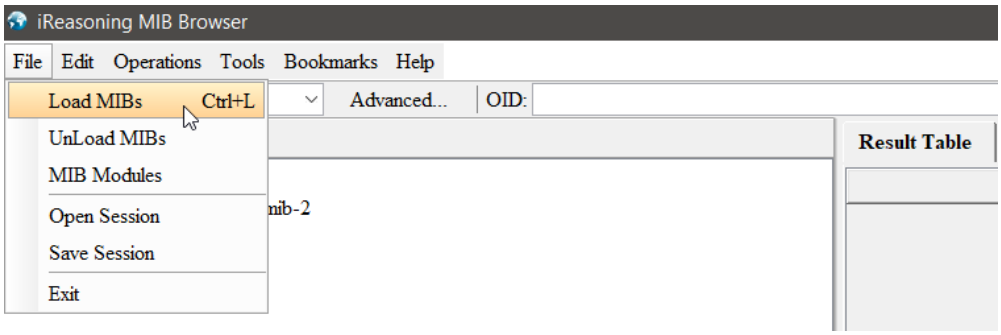
Check the availability of the Mx210 module in the SNMP network using the installed NMS software (e.g. ***OPC server for SNMP***).
Alternatively you can use one of freeware tools available in internet. In the proposed example, ***MIB Browser Free Personal Edition*** is used.

Connection test:

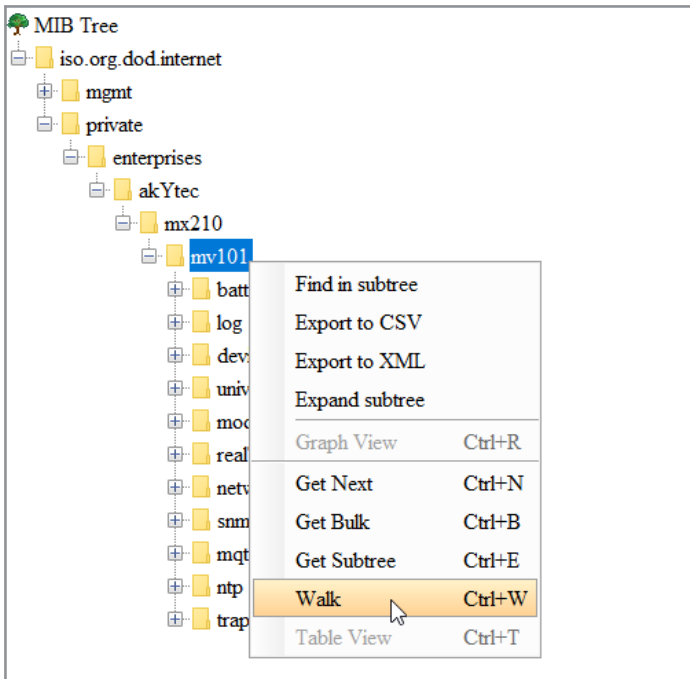
1. Start the tool.
2. Enter the IP address of the module.



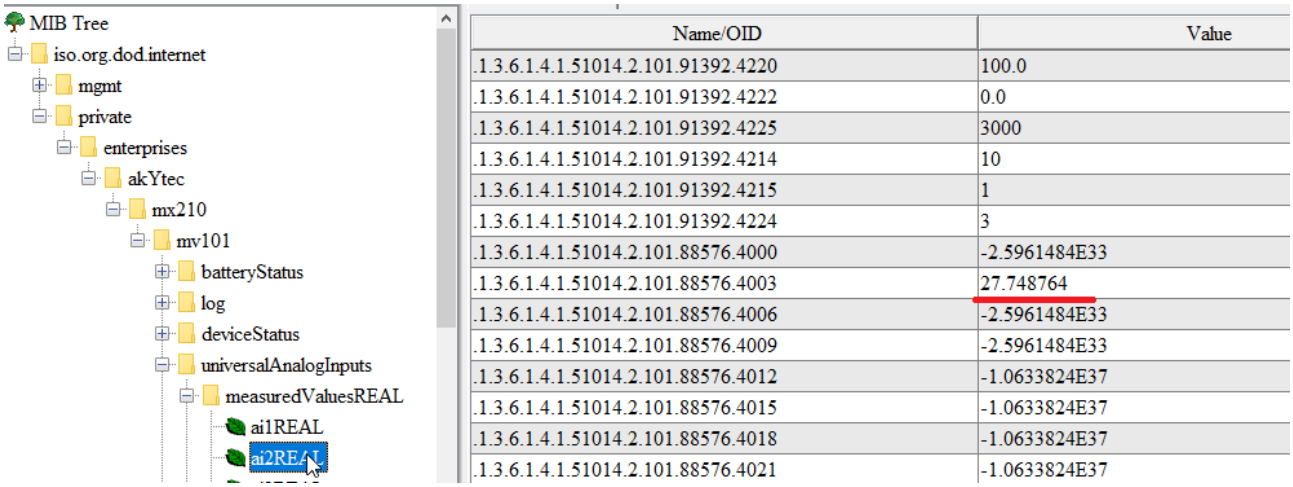
3. Select ***File > Load MIBs*** in the menu.



4. Select the MIB file in the open dialog box and click ***Open*** to confirm.
5. Unfold the hierarchical tree, select the module, right click it and select the item ***Walk*** in the context menu.

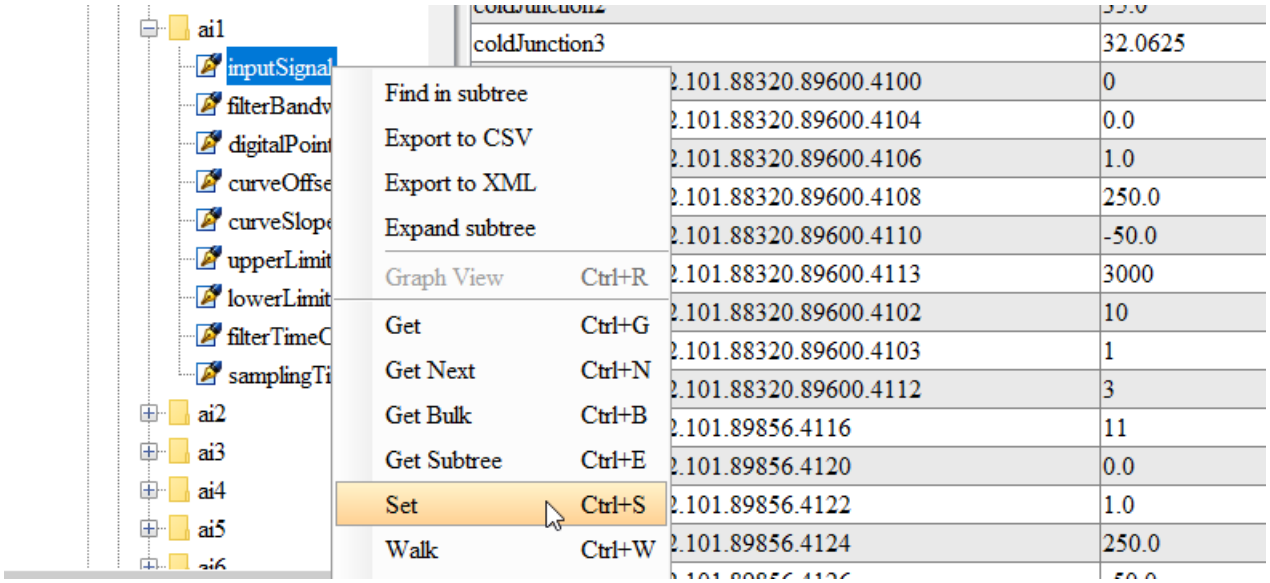


6. In the right pane, you can see all variables available in the module



Name/OID	Value
.1.3.6.1.4.1.51014.2.101.91392.4220	100.0
.1.3.6.1.4.1.51014.2.101.91392.4222	0.0
.1.3.6.1.4.1.51014.2.101.91392.4225	3000
.1.3.6.1.4.1.51014.2.101.91392.4214	10
.1.3.6.1.4.1.51014.2.101.91392.4215	1
.1.3.6.1.4.1.51014.2.101.91392.4224	3
.1.3.6.1.4.1.51014.2.101.88576.4000	-2.5961484E33
.1.3.6.1.4.1.51014.2.101.88576.4003	27.748764
.1.3.6.1.4.1.51014.2.101.88576.4006	-2.5961484E33
.1.3.6.1.4.1.51014.2.101.88576.4009	-2.5961484E33
.1.3.6.1.4.1.51014.2.101.88576.4012	-1.0633824E37
.1.3.6.1.4.1.51014.2.101.88576.4015	-1.0633824E37
.1.3.6.1.4.1.51014.2.101.88576.4018	-1.0633824E37
.1.3.6.1.4.1.51014.2.101.88576.4021	-1.0633824E37

7. The variables with the pen icon can be changed using the command ***Set*** in the context menu.



3. NTP protocol

The module supports the synchronization of the RTC with an NTP server v4.

Open the node ***NTP*** to configure NTP parameters.

Table 4. NTP parameters

Parameter	Description	Range	Default value	Access
Enable	Enable NTP connection	***On*** / ***Off***	***Off***	RW
NTP server pool	IP or URL of NTP pool. If the server is located in an external network, check the correct values for the parameters ***Gateway*** and ***DNS*** (***Network*** group)	-	pool.ntp.org	RW
NTP server 1	IP or URL of the primary NTP server	-	192.168.1.1	RW
NTP server 2	IP or URL of the secondary NTP server		192.168.1.2	RW
Synchronization period	Time synchronization period in seconds. Ensure the set value is not less than the minimum value for the selected NTP server.	5…65535 s	5	RW
Status	Server connection status	-	-	R

All specified NTP servers, including the servers from the pool, have the same polling priority.