USER MANUAL

IIOT EDGE DEVICES

SENECA S.r.l.
Via Austria 26 – 35127 – Z.I. - PADOVA (PD) - ITALY
Tel. +39.049.8705355 – 8705355 Fax +39 049.8706287
www.seneca.it

ORIGINAL INSTRUCTIONS

CAUTION

SENECA does not guarantee that all specifications and/or aspects of the product and firmware, included in them, will meet the requirements of the actual final application even if the product referred to in this documentation is in compliance with the technological state of the art.
The user assumes full responsibility and/or risk with regard to the configuration of the product to achieve the intended results in relation to the specific installation and/or end application.
SENECA may, with prior agreement, provide consultancy services for the successful completion of the final application, but under no circumstances can it be held responsible for its proper functioning.
The SENECA product is an advanced product, the operation of which is specified in the technical documentation supplied with the product itself and/or can be downloaded, if desired prior to purchase, from the www.seneca.it website.
SENECA has a policy of continuous development and accordingly reserves the right to make and/or introduce - without prior notice - changes and/or improvements to any product described in this documentation.
The product described in this documentation may solely and exclusively be used by personnel qualified for the specific activity and in accordance with the relevant technical documentation, with particular attention being paid to the safety instructions.
Qualified personnel means personnel who, on the basis of their training, competence and experience, are able to identify risks and avoid potential hazards that could occur during the use of this product.
SENECA products may only be used for the applications and in the manner described in the technical documentation relating to the products themselves.
To ensure proper operation and prevent the occurrence of malfunctions, the transport, storage, installation, assembly, maintenance of SENECA products must comply with the safety instructions and environmental conditions specified in this documentation.
SENECA's liability in relation to its products is governed by the general conditions of sale, which can be downloaded from www.seneca.it.
Neither SENECA nor its employees, within the limits of applicable law, will in any case be liable for any lost profits and/or sales, loss of data and/or information, higher costs incurred for goods and/or replacement services, damage to property and/or persons, interruption of activities and/or provision of services, any direct, indirect, incidental, pecuniary and non-pecuniary, consequential damages in any way caused and/or caused, due to negligence, carelessness, incompetence and/or other liabilities arising from the installation, use and/or inability to use the product.

CONTACT US	
Technical support	supporto@seneca.it
Product information	commerciale@seneca.it

Document revisions

DATE	REVISION	NOTES	AUTHOR
31/08/2020	0	First revision	MM
23/09/2020	1	Added new function “Serial Trace” Added new function “Factory reset” Added new function “Copy Log to USB” from display and from webserver Moved chapter MODBUS EMBEDDED I/O REGISTERS	MM
23/09/2020	2	Added new parameter “Sleep Timeout” in MQTT CONFIGURATION Aligned with firmware 104 revision	MM
26/11/2020	MI00557-3	Eliminated “optional” in the WI-FI characteristics	A. Zambolin
15/04/2021	MI00557-4	Aligned with fw 108 revision	MM
25/08/2021	MI00557-5	Aligned with fw 109 revision R-PASS product added Removed Bandwidth Limitation parameter in chapter 21.11	MM
02/05/2022	MI00557-6	Aligned with fw 109 revision Added R-PASS product with 2 Ethernet ports	MM
06/05/2022	MI00557-7	Added R-PASS-S product aligned with fw 210 revision	MM
15/12/2022	MI00557-8	Added info on SNMP, OPC-UA protocol. Added R-COMM support Aligned with fw 223 version Added function block list for -S versions	MM
20/06/2023	MI00557-9	Additions by Seneca Service	AS / MM
28/06/2023	MI00557-10	Added new models Z-PASS1-RT, Z-PASS2-RT, Z-TWS4-RT, Z-PASS2-RT-S. Replaced VPN BOX with VPNBOX2 Aligned with SSD/R-PASS fw 232 revision Aligned with -RT fw 1012 revision	MM
03/07/2023	MI00557-11	Small corrections	AZ
20/07/2023	MI00557-12	Stated corrections to the chapter 11.2 (MQTT client)	MM
21/12/2023	MI00557-13	Added chapter “SMS controls”	AZ
14/11/2024	MI00557-14	Rewritten manual for new firmware version rev 3xxx Added model SSD-S Added model SSD-E Updated to rev fw 3100	MM
26/11/2024	MI00557-15	Updated to rev fw 3120 Script execution action addition	MM
27/11/2024	MI00557-16	Added chapter on the keys creation for SSH access to the device	MM
13/12/2024	MI00557-17	New features added for release fw 3122 (on the NETWORK AND SERVICES web page)	MM
21/01/2025	MI00557-18	New features added for release fw 3140 (Audio calls) Packets options changed for SSD	MM
25/02/2025	MI00557-19	New features added for release fw 3160 (multiprotocol groups and error counters) Other tag embedded registers added New instructions added on how to add tags not enabled by default	MM
01/04/2025	MI00557-20	Added new features for firmware release 3180 (String tag, Modem tag)	MM
16/04/2025	MI00557-21	Added new features for firmware release fw 3190 (integration with Seneca Cloudbox2)	MM GS
12/06/2025	MI00557-22	Added new features for firmware release 3230 (added digital counters) Added timing of refresh of the embedded I/Os	MM GS

		Fix Modbus Register of analog I/Os	
26/06//2025	MI00557-23	Added example of MQTT publication with default parameters	MM / GS
30/07/2025	MI00557-24	Added changes for RED-DA regulations from firmware 4.0.0.0 Increased the maximum frequency of digital meters from 100 Hz to 1000 Hz	MM / GP
02/09/2025	MI00557-25	Added changes for firmware revision 4.0.3.0	MM / GP

TABLE OF CONTENTS

1. INTRODUCTION

SENECA IIoT EDGE gateways are key components of industrial automation and offer a range of features that drive efficiency and reliability. These gateways act as digital sentinels of the factory, combining supervision, diagnostics, processing and data storage in a single compact unit.

Supervision is the first line of defence, as IIoT EDGE gateways continuously monitor the health and performance of connected field devices, collect real-time data and provide insights that enable predictive maintenance, reducing downtime and operating costs.

Diagnostic capabilities are also key. These gateways use advanced analyses to detect anomalies and deviations from expected behaviour. In this way, they enable proactive problem resolution, preventing problems before they escalate. The result is higher uptime and more consistent production.

Processing power is another key feature. IIoT EDGE gateways have the processing power to perform data processing operations on-the-fly. They can pre-process data at the source, filtering, aggregating, or transforming it before sending it to the cloud or central systems. This minimizes bandwidth usage and latency, maximizing the value of the data.

Data storage is essential for buffering and storing data locally. In the event of network outages, these gateways ensure that critical data is not lost, and also facilitate historical analysis and reporting, enabling informed decisions.

Real-time management of field devices is the hallmark of these gateways, they can remotely configure, update, and control industrial equipment, allowing operators to respond quickly to changing conditions or emergency situations. This capability simplifies operations and improves overall system resilience.

Security is key, and IIoT EDGE gateways excel at this. They establish secure VPN connections to central control systems, encrypting data in transit, and also apply access controls, ensuring that only authorized personnel can interact with them, safeguarding against cyber threats. These gateways comply with the most stringent cybersecurity standards, starting with OWASP penetration testing, NIST 800 115 Risk Analysis, and IEC 62443.

IIoT EDGE gateways are indispensable in today's industrial environments. They serve as front-line intelligence, providing data supervision, diagnostics, processing, and storage. Secure VPN connections and real-time device management make them the linchpin of efficient, responsive, and secure industrial operations.

1.1. FIRMWARE WITH OPEN SOURCE LPG

Firmwares can contain Open Source software under GPL contract. According to Section 3b of the GPL, it is possible to have the source code for these parts. The source code with the Open Source software license terms can be obtained upon request from Seneca s.r.l.

Send your request to supporto@seneca.it with the subject "Open Source".

2. MODELS

The Edge IIOT Gateway series consists of the following models:

MODEL	DIGITAL I/O	ANALOG INPUTS	DIGITAL COUNTERS (WITH BACKUP)	DISPLAY	PLC STRATON	MODEM 4G	INTEGRATED UPS	SERIAL PORTS	ETHERNET PORTS	CAN PORT	WIFI	IEC61850 IEC60870 PROTOCOLS
SSD	2 DIDO	NO	MAX 2	7" TOUCH + VIRTUAL	NO	NO	NO	2	2	NO	YES	NO
SSD-S	2 DIDO	NO	MAX 2	7" TOUCH + VIRTUAL	YES	NO	NO	2	2	NO	YES	NO
SSD-E	2 DIDO	NO	MAX 2	7" TOUCH + VIRTUAL	YES	NO	NO	2	2	NO	YES	YES
R-PASS	4DI 4DO	2	MAX 4	VIRTUAL	NO	OPTIONAL	OPTIONAL	2	4 (1+3 in switch)	YES	OPTIONAL	NO
R-PASS-S	4DI 4DO	2	MAX 4	VIRTUAL	YES	OPTIONAL	OPTIONAL	2	4 (1+3 in switch)	YES	OPTIONAL	NO
R-PASS-E	4DI 4DO	2	MAX 4	VIRTUAL	YES	OPTIONAL	OPTIONAL	2	4 (1+3 in switch)	YES	OPTIONAL	YES
Z-PASS1-RT	6 DIDO	2	MAX 6	VIRTUAL	NO	NO	NO	3	2	YES	NO	NO
Z-TWS4-RT	6 DIDO	2	MAX 6	VIRTUAL	YES	NO	NO	3	2	YES	NO	NO
Z-TWS4-RT-E	6 DIDO	2	MAX 6	VIRTUAL	YES	NO	NO	3	2	YES	NO	YES
Z-PASS2-RT-4G	6 DIDO	2	MAX 6	VIRTUAL	NO	YES	NO	3	2	YES	NO	NO
Z-PASS2-RT-4G-S	6 DIDO	2	MAX 6	VIRTUAL	YES	YES	NO	3	2	YES	NO	NO
Z-PASS2-RT-4G-E	6 DIDO	2	MAX 6	VIRTUAL	YES	YES	NO	3	2	YES	NO	YES

N.B. Depending on the model, the CAN port may be available but not managed by the firmware revision.

2.1. MODEL DESCRIPTION

2.1.1.SSD / SSD-S / SSD-E

Surprise Smart Display is a 7-inch HMI touch-sensitive colour display (capacitive touch panel), with resolution 800 x 480 and LED backlight

It is an operator panel designed to control and monitor devices, plants or production lines.

Smart Display also offers extensive connectivity thanks to the features of Industrial Gateway, Serial Device Server, Bridge and WI-FI, it is also equipped with an ever-increasing number of industrial protocols.

A novelty introduced in the industrial automation world is the possibility to display variables of the Modbus RTU protocol in a completely passive mode (serial sniffer).

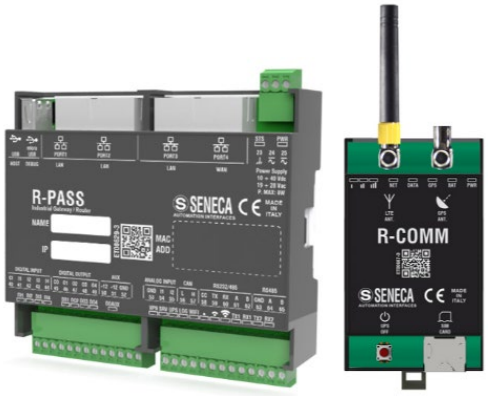
The preloaded software application allows the display of parameters, the sending of commands, the configuration of tags, communication, individual video pages and alarm management.

Includes support for the latest version of LET'S VPN for the maintenance and monitoring of remote devices.

The –S version is also available which includes the PLC Straton IEC 61131.

In addition to including the Straton PLC, the -E version has licenses for energy management protocols.IEC61850 and IEC60870

2.1.2.R-PASS / R-PASS-S / R-PASS-E



R-PASS is a device designed for the control and monitoring of the operation of devices, systems or production lines, it also offers extensive connectivity thanks to the Industrial Gateway, Serial Device Server, Bridge and WI-FI functions, it is also equipped with a continuously increasing number of industrial protocols especially in the IOT sector.
A novelty introduced in the industrial automation world is the possibility to display variables of the Modbus RTU protocol in a completely passive mode (serial sniffer).
It is also equipped with a virtual display accessible from any device via a web browser.
Includes support for the latest version of LET'S VPN for the maintenance and monitoring of remote devices.
The –S version is also available which includes the PLC Straton IEC 61131.
It is possible to connect to the device the R-COMM option which includes a 4G modem and a UPS (optional).
The model with 4 Ethernet ports is available, with and without WIFI.
For more information on the Straton PLC refer to the website: https://straton-plc.com/en/
In addition to including the Straton PLC, the -E version has licenses for energy management protocols.IEC61850 and IEC60870

2.1.3.Z-PASS1-RT / Z-TWS4-RT / Z-TWS4-RT-E

Z-PASS1-RT/Z-TWS4-RT is a device designed for the control and monitoring of the operation of devices, systems or production lines, it also offers extensive connectivity thanks to the Industrial Gateway, Serial Device Server and Bridge functions, it is also equipped with a number of continuously increasing industrial protocols especially in the IOT sector.
A novelty introduced in the industrial automation world is the possibility to display variables of the Modbus RTU protocol in a completely passive mode (serial sniffer).
It is also equipped with a virtual display accessible from any device via a web browser.
Includes support for the latest version of LET'S VPN for the maintenance and monitoring of remote devices.
The Z-TWS4-RT version is also available which includes the PLC Straton IEC 61131.
For more information on the Straton PLC refer to the website: https://straton-plc.com/en/
In addition to including the Straton PLC, the -E version has licenses for energy management protocols.IEC61850 and IEC60870

2.1.4.Z-PASS2-RT-4G / Z-PASS2-RT-4G-S / Z-PASS2-RT-4G-E

Z-PASS2-RT-4G is a device designed for the control and monitoring of the operation of devices, systems or production lines, it also offers extensive connectivity thanks to the Industrial Gateway, Serial Device Server and Bridge functions, it is also equipped with a number of continuously increasing industrial protocols especially in the IOT sector.
A novelty introduced in the industrial automation world is the possibility to display variables of the Modbus RTU protocol in a completely passive mode (serial sniffer).
It is also equipped with a virtual display accessible from any device via a web browser.
Includes support for the latest version of LET'S VPN for the maintenance and monitoring of remote devices.
It integrates the latest generation universal 4G modem.
The –S version is also available which includes the PLC Straton IEC 61131.
For more information on the Straton PLC refer to the website: https://straton-plc.com/en/
In addition to including the Straton PLC, the -E version has licenses for energy management protocols.IEC61850 and IEC60870

2.2. HARDWARE AND SOFTWARE OPTIONS

The devices are available in various hardware formats and with different software features.
All software features can be purchased at the time of ordering or at a later time. The software features are unlocked by entering a key in the appropriate page of the device's web server.

2.2.1.SSD

Smart Display has the following hardware options:

HARDWARE OPTIONS	DESCRIPTION
SMART DISPLAY	7” DISPLAY WITH CAPACITIVE TOUCH NR 2 DIGITAL INPUT NR 2 DIGITAL OUTPUT No. 2 INDEPENDENT ETHERNETs

	MAX 2 DIGITAL COUNTERS WITH BACKUP NON-VOLATILE MEMORY WI-FI / ROUTER WI-FI No. 1 USB HOST PORT
Z-MBUS	MBUS PROTOCOL CONVERTER (METERBUS)

And it is possible to purchase the licenses of the following software options (the packages can be activated even more than one at the same time), by contacting Seneca directly.

SOFTWARE OPTIONS	DESCRIPTION
PACKAGE INCLUDED	Graphic display with widgets and synoptics Virtual display with widgets and synoptics Datalogger max 2000 tags with scaling Alarms Gateway/Router/Firewall Gateway ethernet-serial Serial Sniffer Modbus TCP Client/Server protocol Modbus RTU Master/Slave protocol OPC-UA server protocol
“IOT” PACKET, LOGICS, VPN	HTTP and MQTT protocol for cloud connection” with “Easy Cloud” technology Programmable logics through "IF THEN ELSE" Remote Alarming Simplified VPN connection via "Seneca LET's VPN" environment and VPNBOX2 support Or Open VPN Standard
STRATON (-S) PLC PACKAGE	Allows you to activate the Straton PLC IEC 61131

	Additional protocols provided: Modbus RTU, Modbus TCP-IP, MQTT, OPC-UA Client, MeterBus, S7 Client, SNMP
ENERGY PROTOCOL PACKAGE (-E)	Allows you to activate the Straton PLC and the licenses for the additional protocols IEC61850, IEC60870, Modbus RTU, Modbus TCP-IP, MQTT, OPC-UA Client, MeterBus, S7 Client, SNMP

2.2.2.R-PASS

R-PASS has the following hardware options:

HARDWARE OPTIONS	DESCRIPTION
R-PASS	NR 4 DIGITAL INPUT NR 4 DIGITAL OUTPUT No. 2 ANALOG INPUT 0-10V / 0-20 mA MAX 4 DIGITAL COUNTERS WITH BACKUP NON-VOLATILE MEMORY No. 4 TOTAL: Ni. 1 INDEPENDENT ETHERNET + No. 3 IN SWITCH BETWEEN THEM No. 1 USB HOST PORT
R-PASS-W	NR 4 DIGITAL INPUT NR 4 DIGITAL OUTPUT MAX 4 DIGITAL COUNTERS WITH BACKUP NON-VOLATILE MEMORY No. 4 TOTAL: No. 1 INDEPENDENT ETHERNET + 3 IN SWITCH BETWEEN THEM WIFI
R-COMM-0-4GWW	4G GLOBAL MODEM
R-COMM-B-4GWW	4G GLOBAL MODEM + BATTERY POWERED UPS
Z-MBUS	MBUS PROTOCOL CONVERTER (METERBUS)

And it is possible to purchase the licenses of the following software options (the packages can be activated even more than one at the same time), by contacting Seneca directly.

SOFTWARE OPTIONS	DESCRIPTION
PACKAGE INCLUDED	Virtual display with widgets and synoptics Datalogger max 2000 tags with scaling Alarms

	Gateway/Router/Firewall Gateway ethernet-serial Serial Sniffer Modbus TCP Client/Server protocol Modbus RTU Master/Slave protocol OPC-UA server protocol HTTP and MQTT protocol for cloud connection" with "Easy Cloud" technology Programmable logics through "IF THEN ELSE" Simplified VPN connection via "Seneca LET's VPN" environment and VPNBOX2 support Or Open VPN Standard
STRATON (-S) PLC PACKAGE	Allows you to activate the Straton PLC IEC 61131 Additional protocols provided: Modbus RTU, Modbus TCP-IP, MQTT, OPC-UA Client, MeterBus, S7 Client, SNMP
ENERGY PROTOCOL PACKAGE (-E)	Allows you to activate the Straton PLC and the licenses for the additional protocols IEC61850, IEC60870, Modbus RTU, Modbus TCP-IP, MQTT, OPC-UA Client, MeterBus, S7 Client, SNMP

2.2.3.Z-PASS1-RT / Z-TWS4-RT

Z-PASS1-RT / Z-TWS4-RT has the following hardware options:

HARDWARE OPTIONS	DESCRIPTION
Z-PASS1-RT / Z-TWS4-RT	No. 6 CONFIGURABLE DIGITAL INPUT/OUTPUT MAX 6 DIGITAL COUNTERS WITH BACKUP NON-VOLATILE MEMORY No. 2 ANALOG INPUT 0-10V / 0-20 mA No. 2 INDEPENDENT ETHERNETs No. 1 USB HOST PORT No. 1 SLOT SD CARD
Z-MBUS	MBUS PROTOCOL CONVERTER (METERBUS)

SOFTWARE OPTIONS	DESCRIPTION
PACKAGE INCLUDED	Virtual display with widgets and synoptics Datalogger max 2000 tags with scaling Alarms Gateway/Router/Firewall Gateway ethernet-serial Serial Sniffer Modbus TCP Client/Server protocol Modbus RTU Master/Slave protocol OPC-UA server protocol HTTP and MQTT protocol for cloud connection" with "Easy Cloud" technology Programmable logics through "IF THEN ELSE"

	Simplified VPN connection via "Seneca LET's VPN" environment and VPNBOX2 support Or Open VPN Standard
STRATON (-S) PLC PACKAGE (ALREADY INCLUDED IN THE Z-TWS4-RT MODEL ONLY)	Allows you to activate the Straton PLC IEC 61131 Additional protocols provided: Modbus RTU, Modbus TCP-IP, MQTT, OPC-UA Client, MeterBus, S7 Client, SNMP
ENERGY PROTOCOL PACKAGE (-E)	Allows you to activate the Straton PLC and the licenses for the additional protocols IEC61850, IEC60870, Modbus RTU, Modbus TCP-IP, MQTT, OPC-UA Client, MeterBus, S7 Client, SNMP

2.2.4.Z-PASS2-RT-4G

Z-PASS2-RT-4G has the following hardware options:

HARDWARE OPTIONS	DESCRIPTION
Z-PASS2-RT-4G	No.1 4G GLOBAL MODEM + GNSS No. 6 CONFIGURABLE DIGITAL INPUT/OUTPUT No. 2 ANALOG INPUT 0-10V / 0-20 mA No. 2 INDEPENDENT ETHERNETs No. 1 USB HOST PORT No. 1 SLOT SD CARD
Z-MBUS	MBUS PROTOCOL CONVERTER (METERBUS)

SOFTWARE OPTIONS	DESCRIPTION
PACKAGE INCLUDED	Virtual display with widgets and synoptics Datalogger max 2000 tags with scaling Alarms Gateway/Router/Firewall Serial Sniffer

	Gateway ethernet-serial Modbus TCP Client/Server protocol Modbus RTU Master/Slave protocol OPC-UA server protocol HTTP and MQTT protocol for cloud connection" with "Easy Cloud" technology Programmable logics through "IF THEN ELSE" Simplified VPN connection via "Seneca LET's VPN" environment and VPNBOX2 support Or Open VPN Standard
STRATON (-S) PLC PACKAGE	Allows you to activate the Straton PLC IEC 61131 Additional protocols provided: Modbus RTU, Modbus TCP-IP, MQTT, OPC-UA Client, MeterBus, S7 Client, SNMP
ENERGY PROTOCOL PACKAGE (-E)	Allows you to activate the Straton PLC and the licenses for the additional protocols IEC61850, IEC60870, Modbus RTU, Modbus TCP-IP, MQTT, OPC-UA Client, MeterBus, S7 Client, SNMP

3. IP ADDRESSES

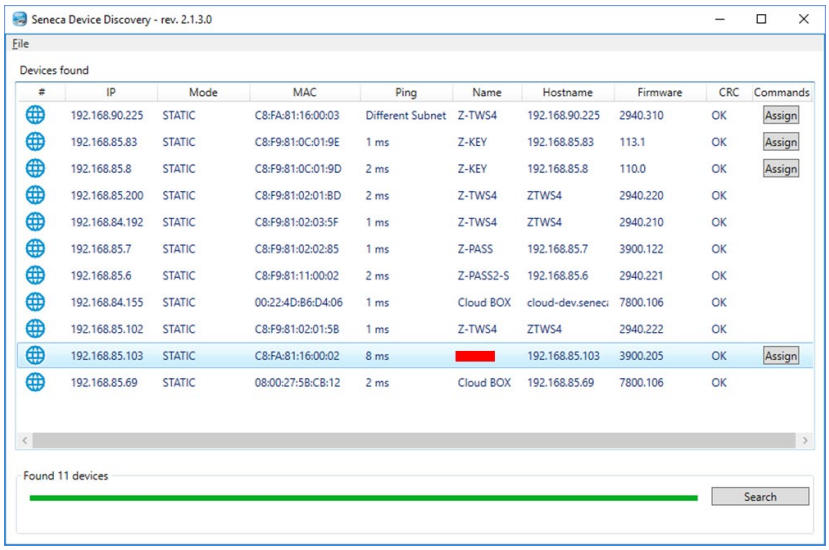
3.1. FACTORY IP ADDRESSES

The devices leave the factory with the following configuration:

ETHERNET “LAN” PORT	static IP: 192.168.90.101
ETHERNET “WAN” PORT	DHCP active
WI-FI	Not active (where present)

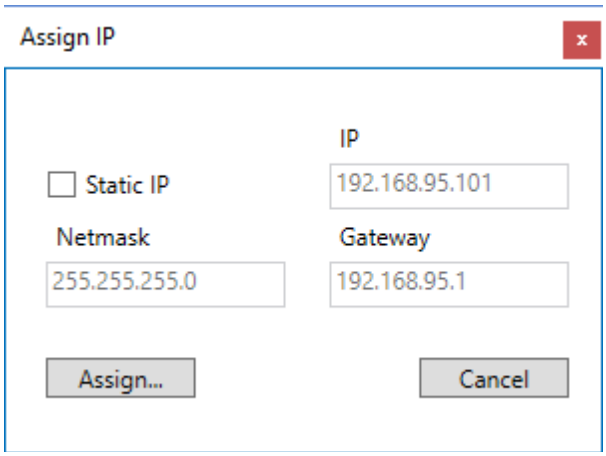
3.2. IP ADDRESS SEARCH

The devices leave the factory with the default IP address 192.168.90.101, on Ethernet (LAN),
If this address is changed or forgotten, it can be recovered using the "Seneca Device Discovery" software.

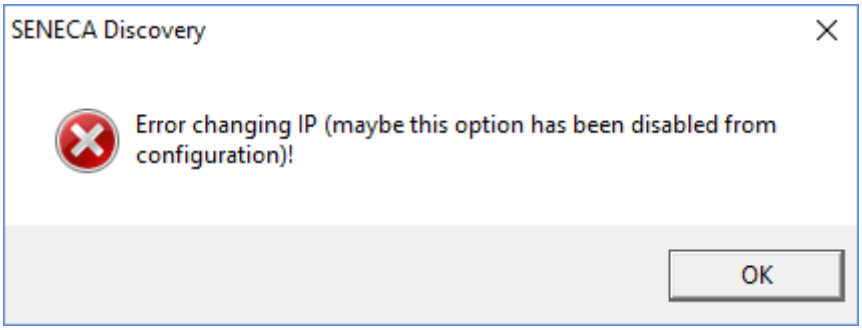


This application shows the IP address, MAC address, FW version and some other useful information, for each SENECA device connected to PC.

Moreover, by clicking on the "Assign" button, it is possible to modify the device network parameters, as shown in the following figure:



For security reasons, this function can be disabled, in this case, after clicking on the "Assign" button the following error message will be visualized



The software can be easily installed by running the installation program available at the following link: http://www.seneca.it/products/sdd

NOTE:

The IP address shown by the Seneca Discovery Device software is the IP address of the LAN peripheral when the PC is connected to the LAN port, the WAN IP address when the PC is connected to the WAN port and of the WI-FI if it is connected to the latter; moreover, all the changes of network configuration parameters are applied to the relative peripheral.

4. ACCESS TO THE WEBSERVERS OF THE DEVICES

IIOT devices are equipped with two webservers:

- The webserver with the virtual display
- The configuration webserver

4.1. CONFIGURATION WEBSERVER ACCOUNT

In addition to the “ADMIN” account, there are also the “guest” and “operator” accounts:

4.1.1. CONFIGURATION WEBSERVER WITH “GUEST” ACCOUNT

You can access the configuration site of the device with the “guest” account; this account is not allowed to access all the pages but it is possible to view all the configuration parameters and the status information, without being able to modify them; therefore, in all the pages, the “APPLY” buttons (and any other button used to make changes) are disabled.
To log in with a "guest" account, connect your browser to the IP address of the device, for example:
https://192.168.90.101
and, when required, provide the following credentials (default values):
Username: guest
Password: guest
You will then be asked to change your password.

4.1.2. CONFIGURATION WEBSERVER WITH “OPERATOR” ACCOUNT

You can access the device configuration site with “operator” account; this account can only configure IP addresses.
To log in with a "operator" account, connect your browser to the IP address of the device, for example:
https://192.168.90.101
and, when required, provide the following credentials (default values):
Username: operator
Password: operator
You will then be asked to change your password.

4.2. FIRST ACCESS TO THE WEB SERVER

The devices are accessible by default from the “LAN” Ethernet port with the static IP address 192.168.90.101 Web servers are available via https.

On the https protocol the webserver with the virtual display is located on port 443, so type:

https://192.168.90.101/gui

The web server for configuration is always on port 443 but at the following URL:

https://192.168.90.101

Username: admin
Password: admin
The first time you log in, you will be asked to change the default password.

WebFactory webserver (only available in legacy mode) can be accessed at:

https://192.168.90.101/user

4.3. WEBSERVER WITH VIRTUAL DISPLAY

For more information on this Webserver, refer to the relevant chapter of this manual.

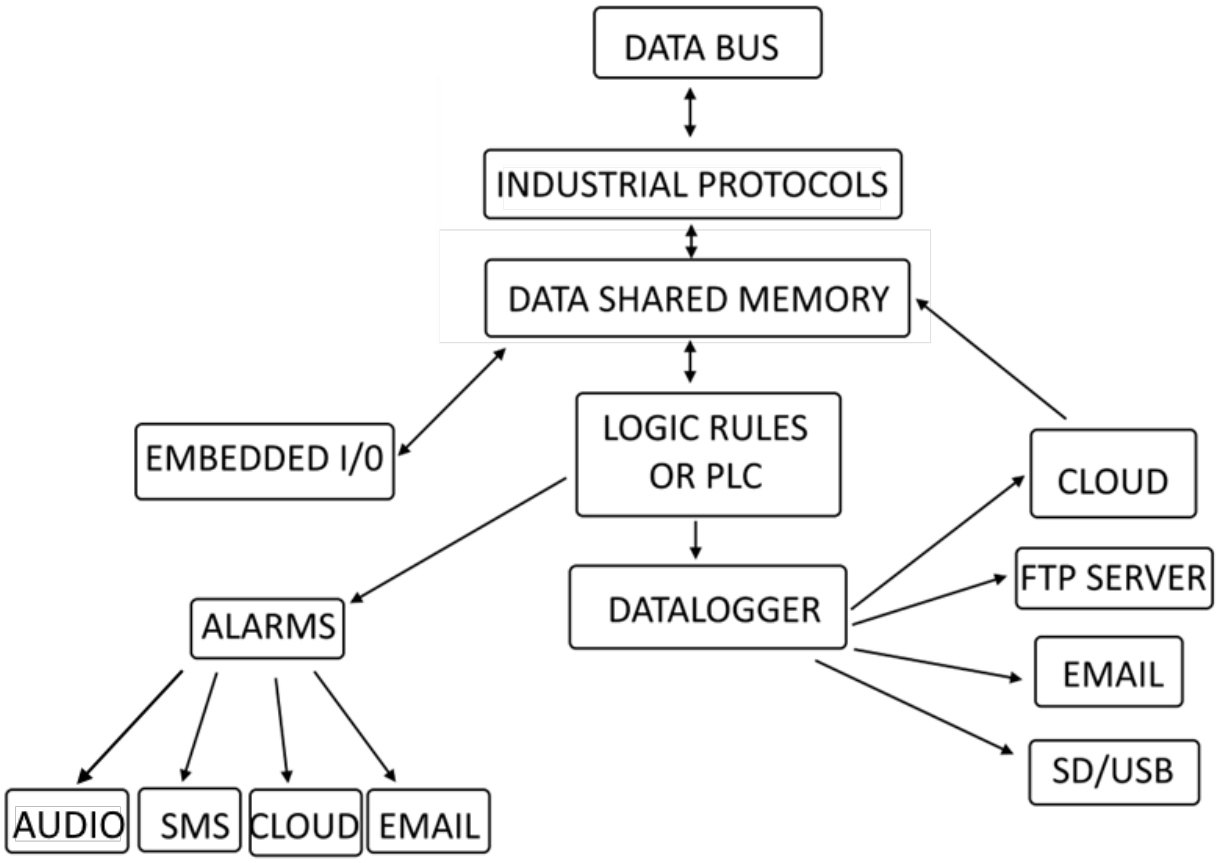
4.4. CONFIGURATION WEBSERVER

For more information on this Webserver, refer to the relevant chapter of this manual.

5. DATA ACQUISITION AND PROCESSING, ALARMS GENERATION AND SENDING, DATA SENDING

Edge IIOT devices allow you to acquire data from the embedded IOs of the devices or from the buses (via industrial communication protocols), this data is saved in a shared memory and can be processed via scaling or via logical rules or via the Straton PLC. Once the data has been processed, it is possible to save it in an external storage device (USB or SD card) or send it to the clouds or FTP/Email servers etc.
The alarms are generated by the logical rules and can also be sent to the clouds or via Email/SMS or via audio call.

Please refer to the following block diagram:



The acquisition of data (Tags) in the buses (Data Bus) takes place via industrial protocols (Industrial Protocol) or via direct acquisition of the integrated I/Os (Embedded I/O).
This data flows into the shared memory (Data Shared Memory), in this memory the logic rules or PLC perform the data processing (Logic Rules or PLC).
The data logger acquires the processed data and stores it via client protocols (on Cloud, FTP server, Email, SD card, Usb storage).
The logical rules or the PLC generate alarms that can be sent via EMAIL, Cloud, SMS or audio calls.

Audio calls can be configured to patrol as long as there is confirmation of alarm receipt via a DTMF tone combination (#99*).
The Cloud can access and then write the data already processed in the shared memory (Shared Memory).
Below we will analyse the main components of the block diagram.

5.1. THE DATA BUS AND INDUSTRIAL PROTOCOLS

Typically, the data resides in external devices and must be connected via industrial protocols.
The device includes a series of industrial protocols so that it can connect with the most varied third-party manufacturers.
Among the most important protocols we mention the Modbus protocols and the OPC-UA protocol

5.1.1. MODBUS PROTOCOLS

Modbus was born as a serial communication protocol by Modicon (a company now part of the Schneider Electric group) to connect their programmable logic controllers (PLCs). It has become a de facto standard in industrial communication, and is currently one of the most widespread connection protocols in the world among industrial electronic devices. In addition to the serial version, Seneca devices also support the Ethernet-based version.
The supported Modbus protocols are:

Modbus RTU Master protocol
Modbus RTU Slave protocol
Modbus TCP-IP Client protocol
Modbus TCP-IP Server protocol

For further information, see website:

https://modbus.org/

Thanks to these protocols it is possible to acquire variables in the memory directly from external Modbus RTU slave or Modbus TCP-IP server devices.

5.1.2. OPC- UA PROTOCOL

OPC Unified Architecture (OPC-UA) is a standardized machine-to-machine communication protocol for industry 4.0 developed by the OPC Foundation.
OPC-UA is a vendor-independent communication protocol and is based on the client-server principle.
Seneca devices support the OPC-UA server protocol also with security policy.

For further information, see website:

https://opcfoundation.org/

In particular, the OPC-UA server "exports" the internal memory tags then, using an OPC-UA client or other protocol it will be possible to read and write directly all tags.

5.2. SHARED MEMORY AND TAGS

The data acquired from the buses or I/O integrated into the devices flow into the shared memory, this memory is accessible from outside the device with various protocols (for example OPC-UA or Modbus TCP-IP or RTU).
Each piece of data is identified by a mnemonic name and a type (integer, floating point etc.), thus characterized it takes the name of "Tag".
On these Tags it is possible to perform various types of processing as we will see later in the manual.

5.3. DATALOGGER

The Seneca IIOT Edge Gateways include a powerful data logger that allows you to manage up to 2000 variables at the same time (TAG). It is also possible to scale each variable and perform further processing with the PLC or with logical rules. The data acquired by the data logger can then be sent to the various clouds/FTP/EMAILs or to USB/SD memories.

For the function, when the gateway function is set to “Modbus Gateway with Shared Memory” in the device it is also possible to activate the “Data Logger” mode:

Tag values are periodically stored in files (called "log files"), which can then be transferred.

Tags can be associated with up to four groups of Data Loggers, which can have different sampling and transfer periods and different transfer methods.

The following "transfer" methods are currently supported;

- copied to USB stick / SD Card
- transferred to an FTP server
- sent to one or more e-mail addresses, as an attachment
- Sent to a server via http post
- Sent to an MQTT broker

More than one of the above methods can also be enabled at the same time.

Log files are stored in flash memory, so if one of the transfer methods temporarily fails, it can be successfully transferred later.

For each group of data loggers, the "cache" is filled if at least one of the following cases is reached:

- 1000 log files
- 500000/(number of groups enabled) samples (i.e. number of lines of a single log file)

When the limit is reached, the cache is "rotated", i.e. the oldest files are overwritten by the new ones.
The file protocols (copy to USB/SD card, EMAIL or FTP) use standard "csv" type log files, they can therefore be processed by Excel™ or PC software.

Here is a portion of an example log file

```
INDEX;TYPE;TIMESTAMP;ZPASS_DI;ZPASS_DO;ZPASS_DI_1;ZPASS_DI_2;ZPASS_DI_3;ZPASS_DI_4;ZPASS_DO_1;ZPASS_
DO_2;ZPASS_DO_3;ZPASS_DO_4;GPS_ERROR;GPS_HOUR;GPS_MINUTE;GPS_SECOND;GPS_DAY;GPS_MONTH;GPS_YEAR;GPS_L
ATITUDE;GPS_LONGITUDE;GPS_HDOP;GPS_ALTITUDE;GPS_COG;GPS_SPEED_KM;GPS_SPEED_KN;GPS_FIX;GPS_NUM_SAT;SH
M_TAG1;ZPASS2_105_TAG1;ZPASS2_106_TAG1;ZPASS2_106_TAG2
1;LOG;29/05/2018 09:49:45;0;0;0;0;0;0;0;0;0;0;0;7;49;31;29;5;18;45.37417;11.94554;1.5;12.7;249.56;0;0;2;4;0;32767;14;11.5
2;LOG;29/05/2018 09:49:50;0;0;0;0;0;0;0;0;0;0;0;7;49;31;29;5;18;45.37417;11.94554;1.5;12.7;249.56;0;0;2;4;0;32767;14;11.5
3;LOG;29/05/2018 09:49:55;0;0;0;0;0;0;0;0;0;0;0;7;49;31;29;5;18;45.37417;11.94554;1.5;12.7;249.56;0;0;2;4;0;32767;14;11.5
4;LOG;29/05/2018 09:50:00;0;0;0;0;0;0;0;0;0;0;0;7;49;31;29;5;18;45.37417;11.94554;1.5;12.7;249.56;0;0;2;4;0;32767;14;11.5
```

If for a tag the actual value is not available (for example, if the tag corresponds to a log that does not respond to Modbus requests), the value written in the corresponding field of the log file can be set to "ERR!"
The "ERROR MODE" parameter can also be set to LAST VALUE or to a user-defined FAIL value.

Please note that each time a configuration change is made that affects the functionality of the Data Logger (from a page in the "Datalogger" section) the following procedure is performed:

- Data Logger processes are interrupted
- The log file cache is cleared

5.4. TAG PROCESSING: LOGICAL RULES AND STRATON PLC

Two main forms of Tag processing can be used in the device.
The first is through logical rules, the second is through a PLC (optional).
For more information, refer to the respective chapters of this manual.

5.5. CONNECTION TO CLOUDS VIA "EASY CLOUD" TECHNOLOGY

The "Easy Cloud" technology is based on the MQTT protocol and allows bidirectional connection with the main available clouds.

Some of the clouds to which devices can connect are:

5.6. ALARMS

A complete set of parameters are available for TAG alarms, as indicated in the “Alarm Configuration” page of the webserver.
The entire alarm status can be viewed in the “Alarm Summary” page and the alarm history can be retrieved in the “Alarm History” page.
Moreover, in the "Tag View" page, the columns "ALARM" and "ANALOG DANGER ALARM" show the current alarm status for each tag.
Alarm generation is managed through logical rules or directly from the Straton PLC (optional).

6. GRAPHICAL DISPLAY OF DATA ON THE DISPLAY / VIRTUAL DISPLAY

The Seneca IIOT Edge Gateways include a powerful graphical interface, depending on the model there is a 7" touch physical display and/or a virtual display accessible via a web browser. Everything that can be done in the real display is also available in the virtual one, the finger touch is replaced by the pointer and the mouse button.

The display consists of 3 sections:

"A" Represents the bar with the device information
"B" Represents the Smart Display menu
"C" Represents the Widget page

6.1. INFORMATION BAR

Represents the information related to the device status, in particular:

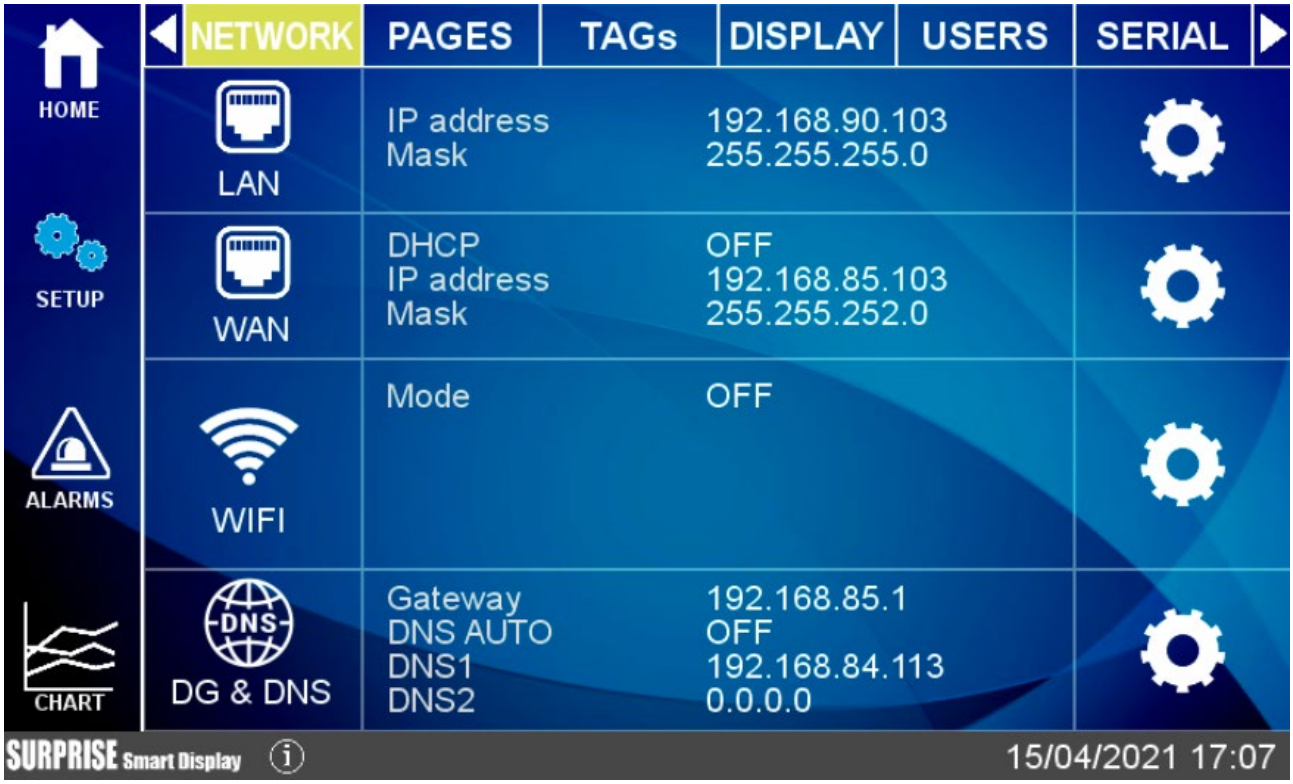
Icon "A" provides information about the device (such as firmware revision) and manufacturer
Icon "B" provides user account information; in case you are not logged in, the icon is replaced by a padlock. The icon on the left, if pressed, allows to logout, the icon on the right indicates the type of user account (A stands for administrator). In the case of guest accounts the icon is shown as follows :
"C" icon shows the status of the serial port COM1
Icon "D" shows the status of the serial port COM2
Icon "E" shows the status of the VPN connection: "Seneca Let's VPN" or "OpenVPN standard".
Icon "F" Provides the strength of the WI-FI signal (if present, depending on the model)
"G" icon shows the status of the datalogger
"H" shows the date/time of the device

6.2. MENU

Shows the menu:
HOME leads to the main page
SETUP leads to device configuration of the device
ALARMS leads to the alarms section
CHART leads to the section related to the graphic analysis of the datalogger data
It is also possible to hide the menu pressing the side bar:

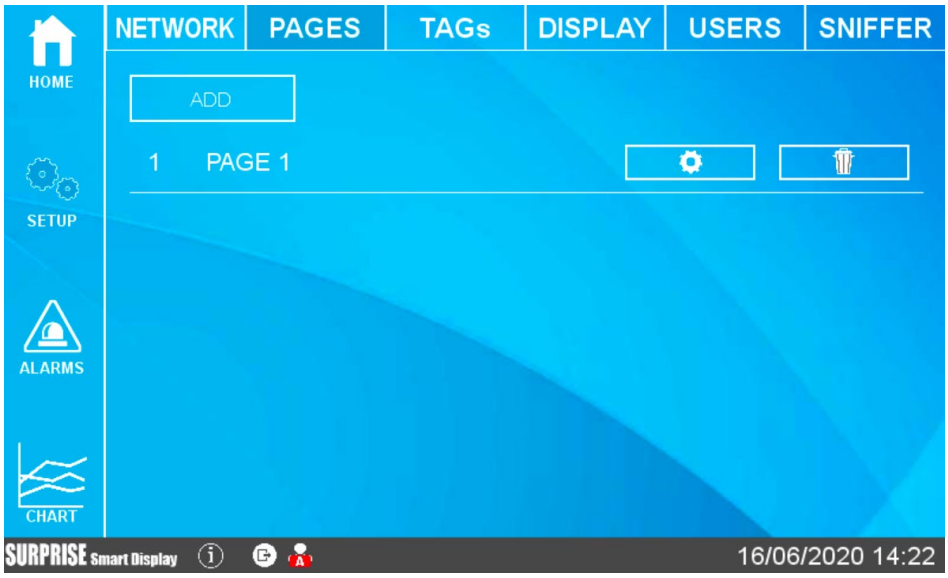
6.2.1.SETUP

6.2.1.1. NETWORK

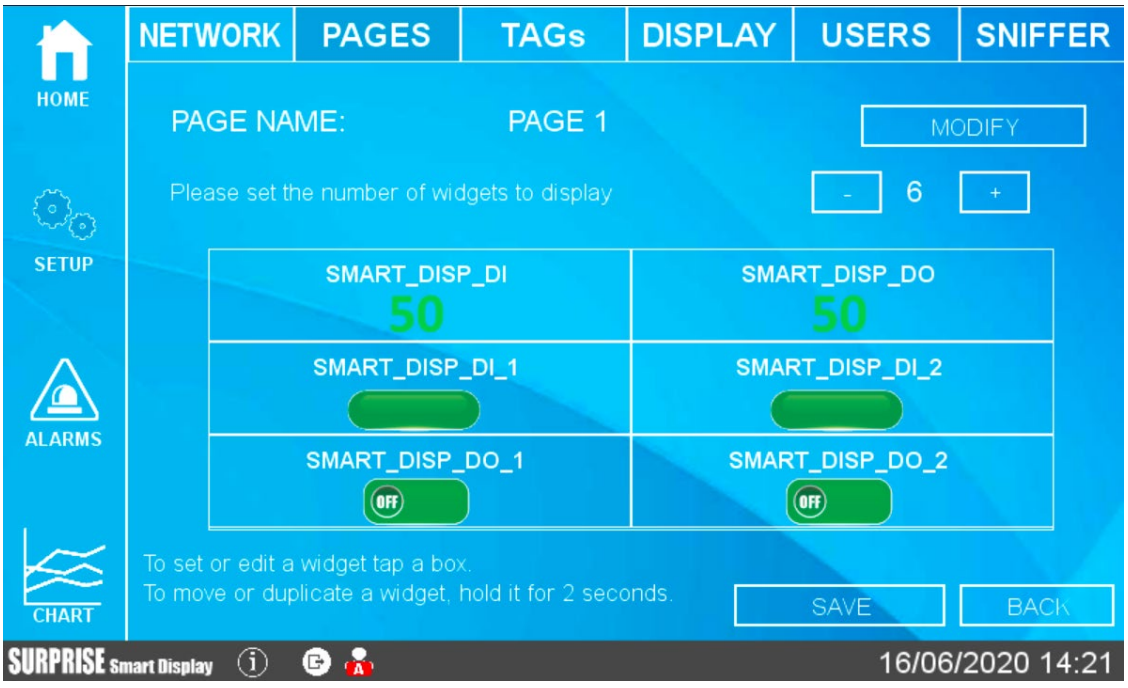


In this section it is possible to configure the settings for the LAN and WAN Ethernet ports and WI-FI network port.
The WIFI port section allows to you choose WI-FI Station or Access Point mode.
The Station mode allows the device to connect to an existing Wi-Fi network, instead, the Access Point mode allows the device to create a new Wi-Fi network to which other devices can connect.

6.2.1.2. PAGES



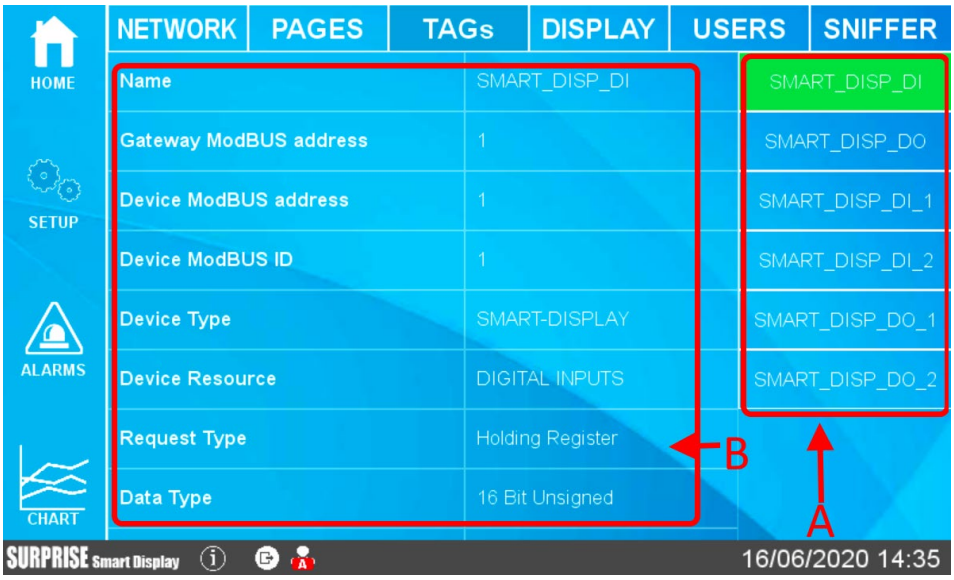
On the first screen it is possible to add as many pages as user desires and once pages are created, he is able to edit configuration of each one.



It is possible to change the page name and the number of widgets to show.
In the central part there is a preview of the page visualization.
Pressing on a widget icon it is possible to modify the widget parameters: type, colour, etc...

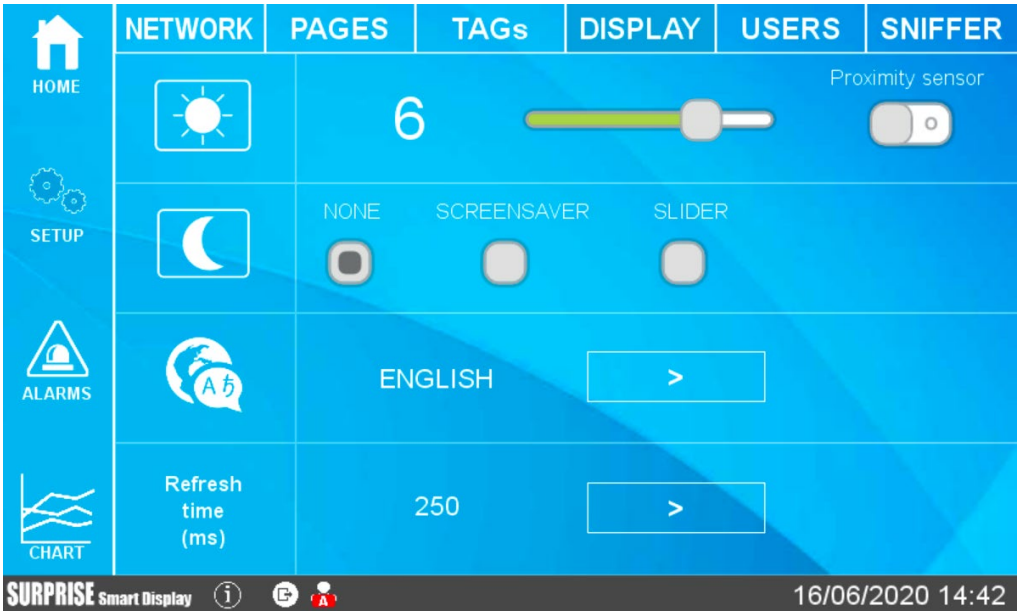
In addition to a widget page it is possible to add a Synoptic page. In a synoptic page it is possible to freely position the widgets and upload graphics from a PC or from a graphics library inside the device to create synoptic pages without the aid of external software.

TAGS



In this section the configured tags are visualized.
The device tags are located on the right side (A), it is possible to browse the list.
The parameters of each tag appear in the central part (B), you can also scroll through the list.
From firmware version 109 it is possible to add, edit and delete tags also from the display.

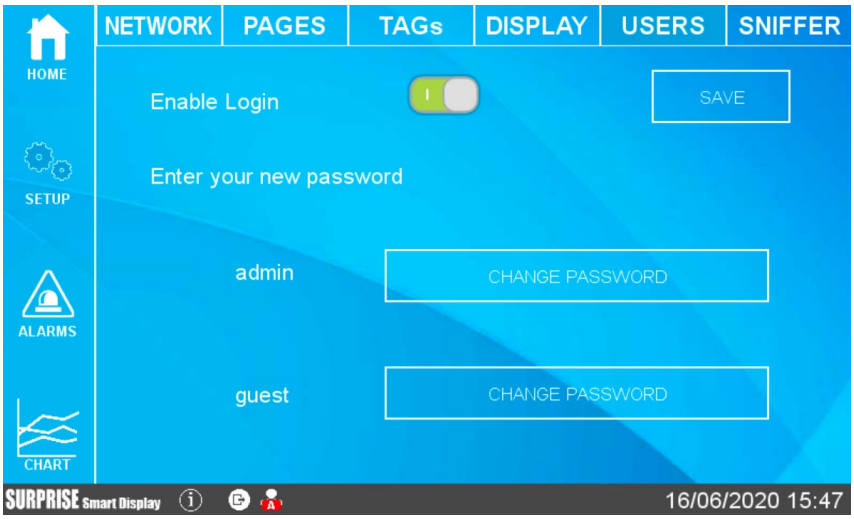
6.2.1.3. DISPLAY



This section allows to configure the screen brightness, language and screen refresh time.
In order to safeguard the consumption and the duration of the screen, it is possible to activate the screensaver (the backlighting of the screen is lowered after the set idle time).

If the screensaver mode is enabled it is possible to exit by pressing anywhere on the screen (or making a movement in front of the screen if the proximity sensor is activated).
Slider mode, instead, allows to cycle the widget pages automatically after a preset time.

6.2.1.4. USERS



This section allows to configure the users who can access to the display.
It is possible to disable the login to access the display (free access) or activate an administrator account and/or guest account.

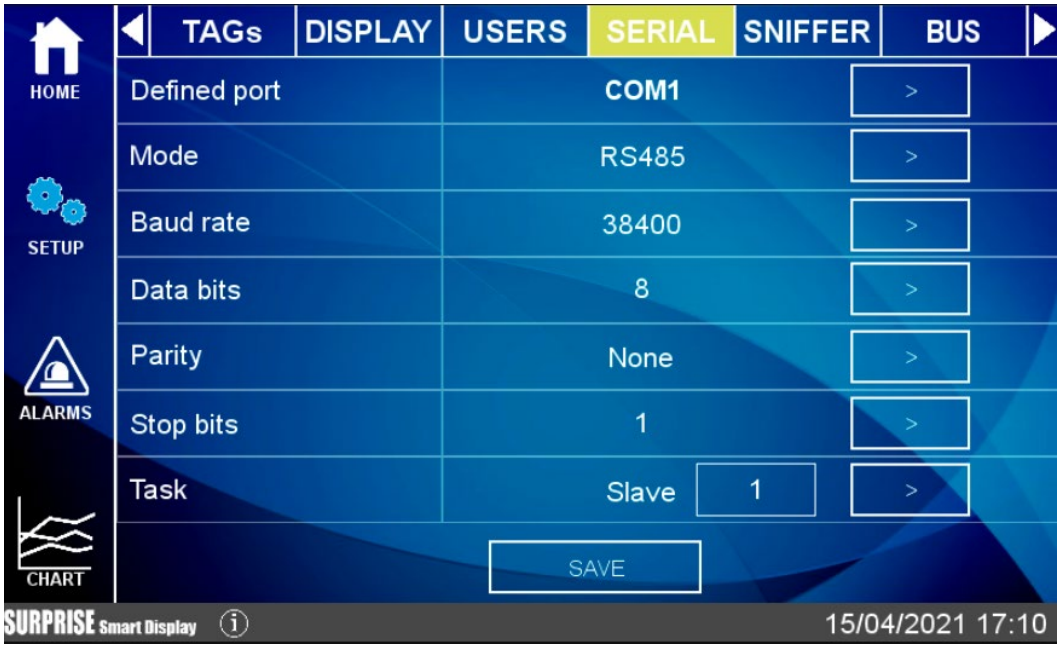
According to the following table

ACCOUNT TYPE	*CHANGING THE VALUE OF A TAG*	*SETUP MENU DISPLAY*	*SETUP MODIFICATION*
ADMIN	Yes	FULL	Yes
GUEST	Yes	ONLY "NETWORK" AND "TAGS"	NO
NO ACCOUNT	No	NO	NO

If the screen saver is switched off and none touch the screen for 2 minutes the system will automatically logout.
If the screen saver is activated and none touch the screen for a time equal to the screen saver time, the system will automatically logout.

6.2.1.5. SERIAL

Allows you to configure the parameters of serials and define whether the Modbus protocol must be Master or slave.

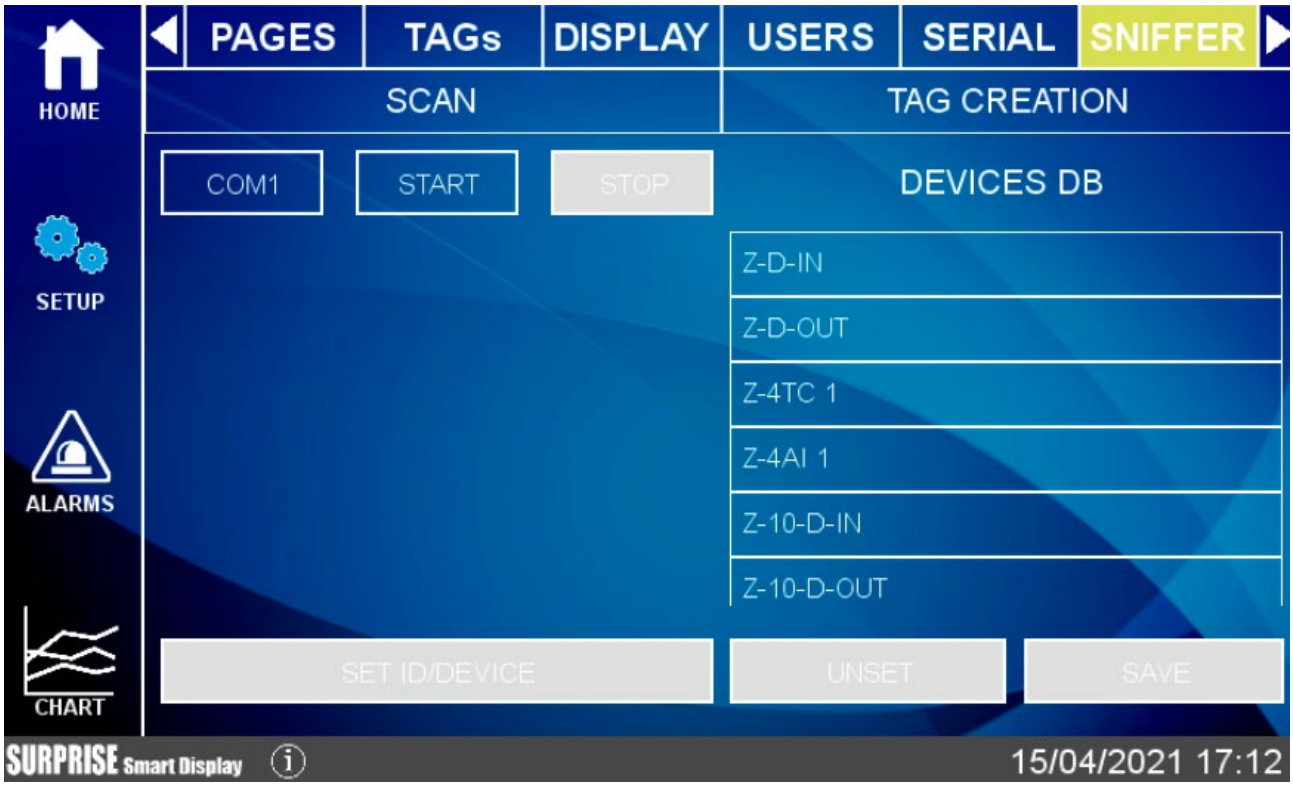


6.2.1.6. SNIFFER

The serial sniffer function allows you to insert one or more sniffer devices into an existing system with Modbus RTU protocol in an RS485 bus.
For Modbus RTU protocol there is always a single master and a series of slave devices. The master requires registers to read/write to each slave, who answers sending requested data.
In order to insert a device that displays data without changing the existing configuration, it is necessary to insert one or more devices in passive mode (sniffer).
At this point the devices will receive all the serial packets transmitted between the master and the slaves and it is necessary to associate these packets to tags that will be valued.

ATTENTION!
As the SNIFFER mode is purely passive all defined tags will be read-only

6.2.1.7. SNIFFER MODE CONFIGURATION STEPS



The sniffer mode is configured through the following steps (the three buttons at the top of the page):

1) BUS COMMUNICATION SCAN

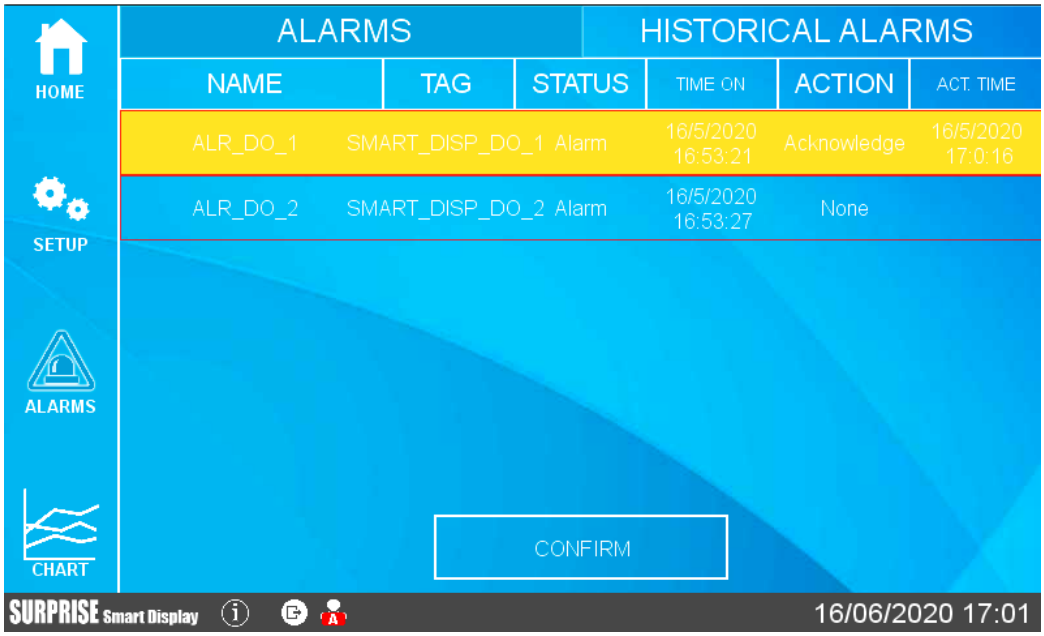
In this learning mode the device will start to scan the flow of information passing through the bus. Typically, a Master interrogates all the devices in a continuous cycle, so when you are sure that the cycle has ended you can stop the scan. Attention: the operation to stop the scan is always manual.

2) TAG CREATION

In this phase the device has identified the registers that the devices are exchanging, now it is necessary to associate the name of the tag and the type of data it contains. In the case of a system with Seneca products, it will be necessary to introduce the type of Seneca device and the system will automatically associate the correct tags, in the case of third party devices, the information relating to each register identified will be requested.

6.2.2. ALARMS

This section shows the active alarms and alarm history.
If the alarm requires manual acknowledgement, it is possible to use the appropriate button:



In the Historical section are represented all the alarms that have occurred so far:

ATTENTION!
ALARMS ARE CONFIGURED IN THE APPROPRIATE SECTION OF THE WEBSERVER

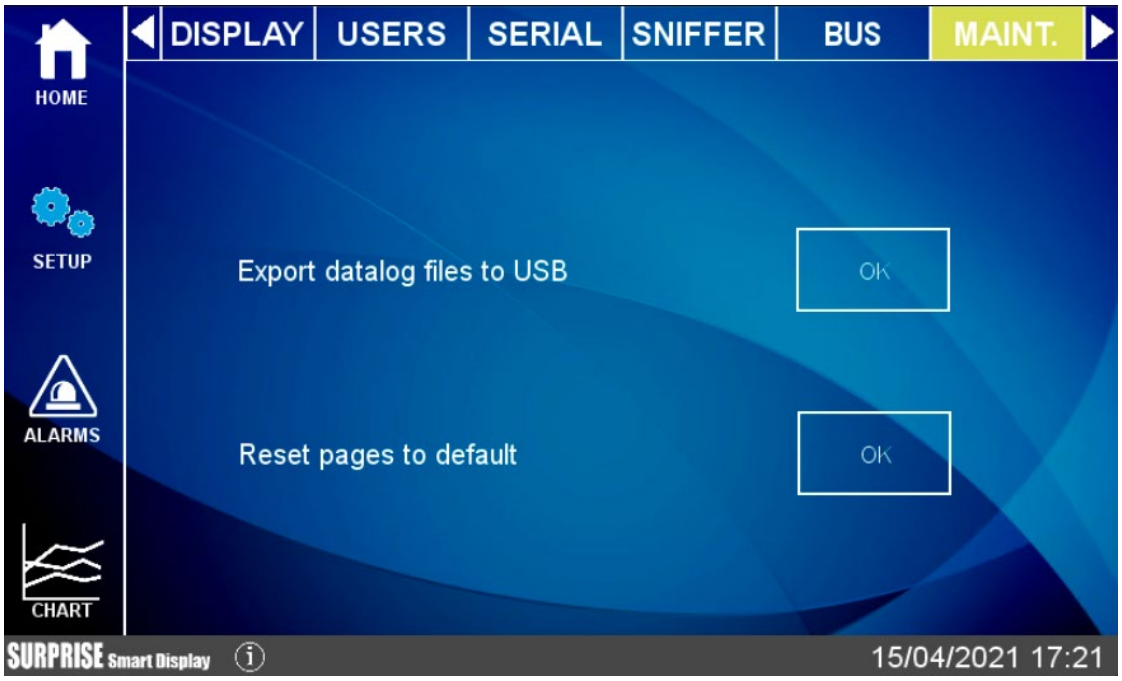
6.2.3. BUS

This section allows external devices to be added via serial and/or Ethernet and their tags to be inserted:

The device uses a database that includes records of all Seneca devices.
Adding a device can be done in manual mode (by entering the device among those in the database or from a manufacturer other than Seneca) or by automatically searching for the device on serial or Ethernet.
The automatic search also automatically creates tags but only works with Seneca devices.

6.2.4. MAINTENANCE

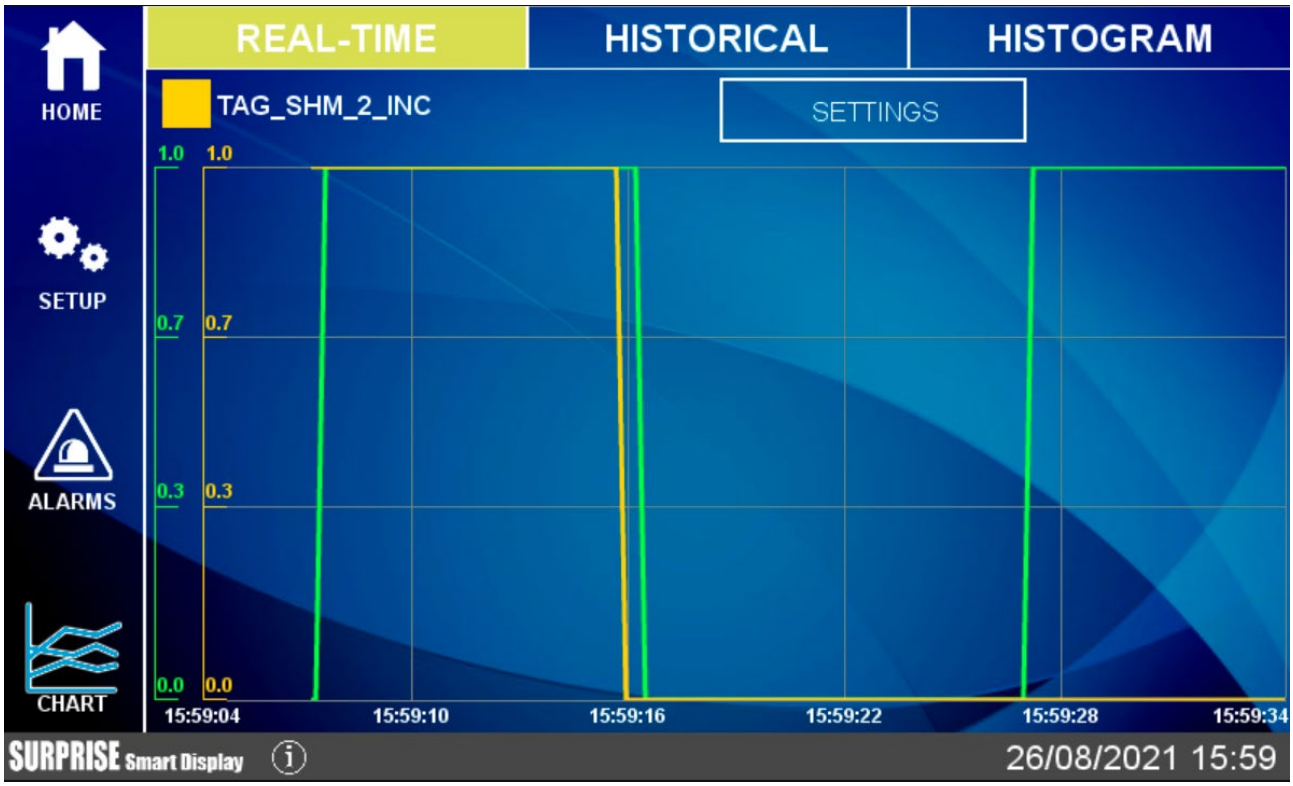
The Maintenance menu allows maintenance operations to be carried out on the device:



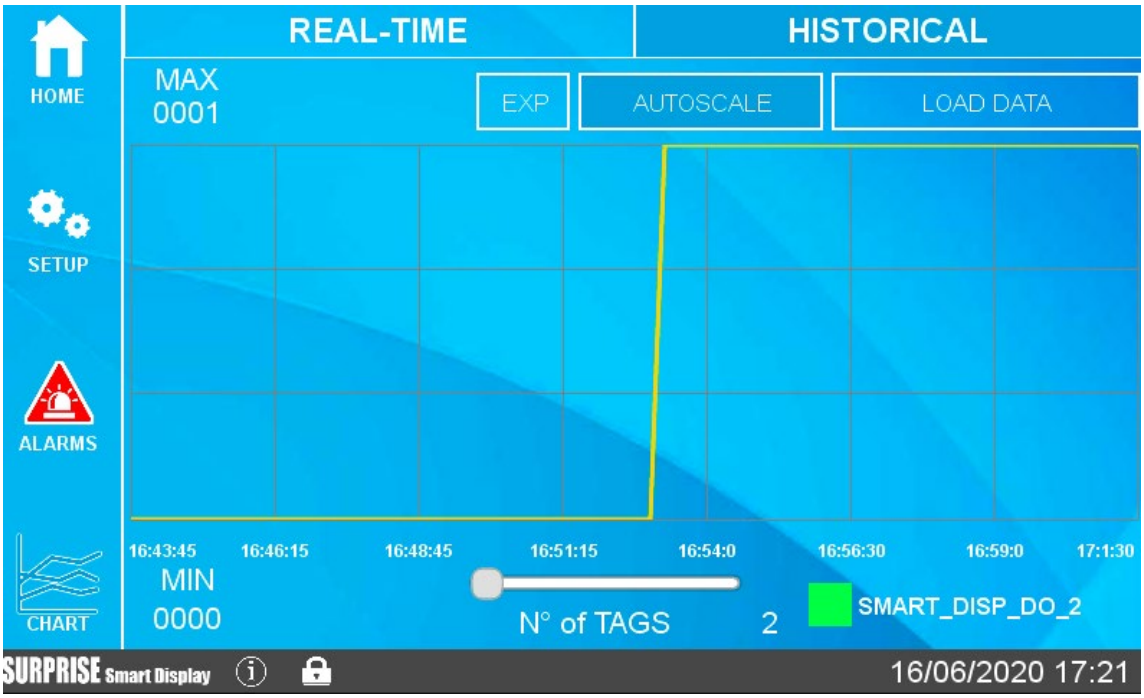
6.2.5. CHART

There are 3 types of graph available: Real Time, Historical and Histogram.

In the Chart Real Time section the tag values are displayed in real time (maximum 10 tags):



The configuration of the real time graph will be recalled also from the relative widget.
In the Historical section, on the other hand, you can load data in the desired range and move back and forth in the graph, using the touch screen.

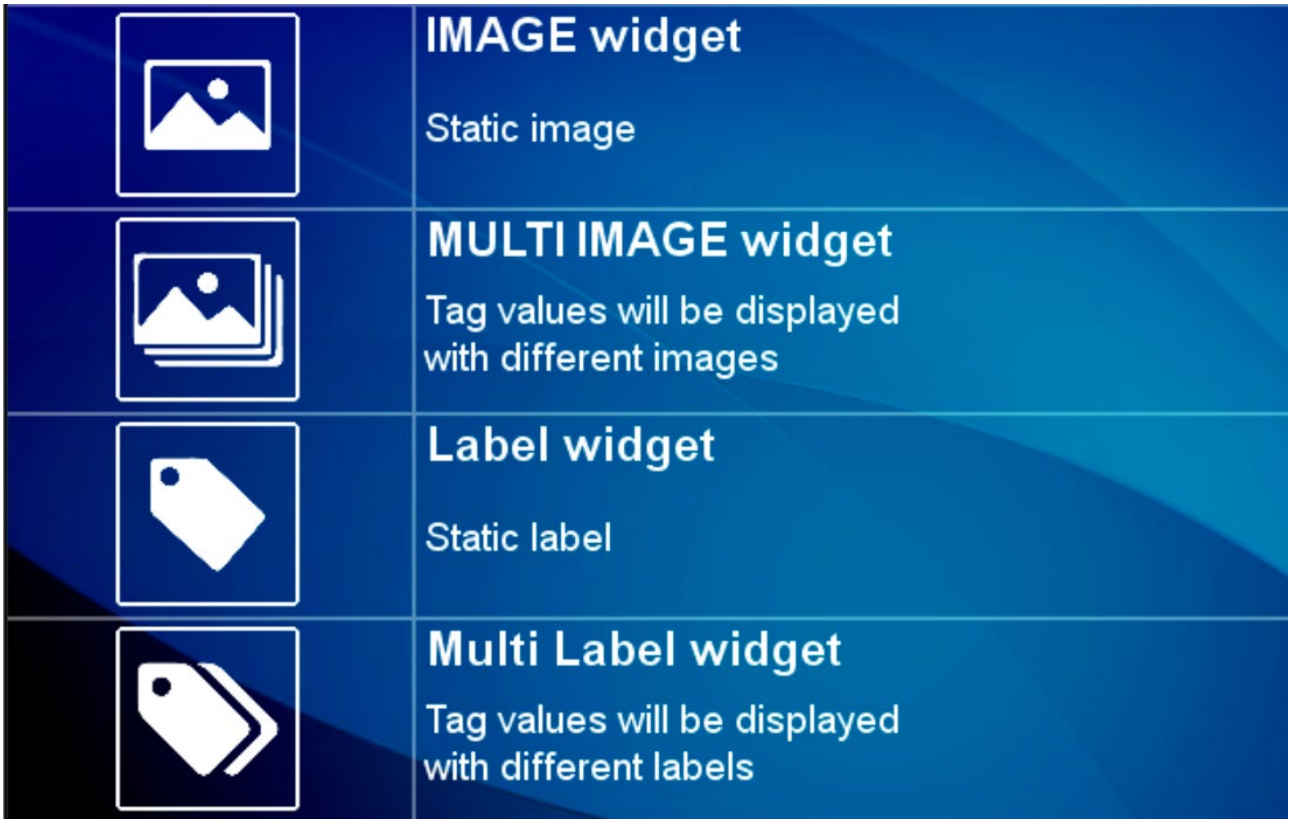


In case a USB disk is connected, it is possible to export to a file the chart values displayed, by pressing the "EXP" button.
If user is connected via web to the remote display, pressing the "EXP" button the browser will download the file directly to the PC.

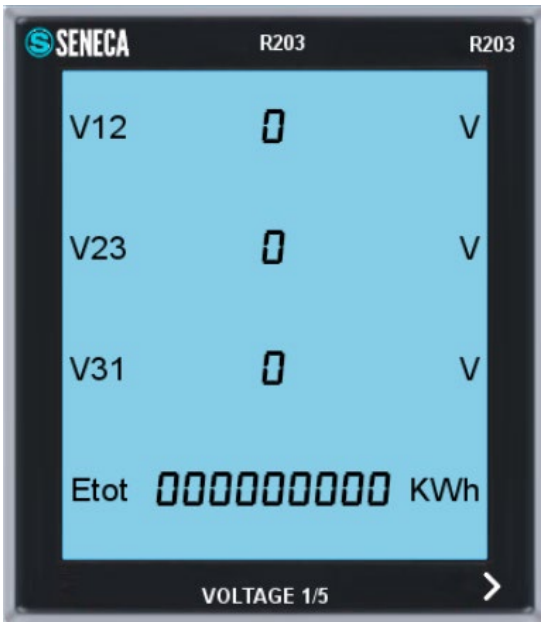
The Histogram chart is essentially the same as the Historical chart but with a histogram representation.

6.3. TYPE OF WIDGETS

Widgets are graphic elements that can be linked to one or more TAGs.
These can be used in both widget pages and synoptic pages.
There are various widgets available, here are some examples:



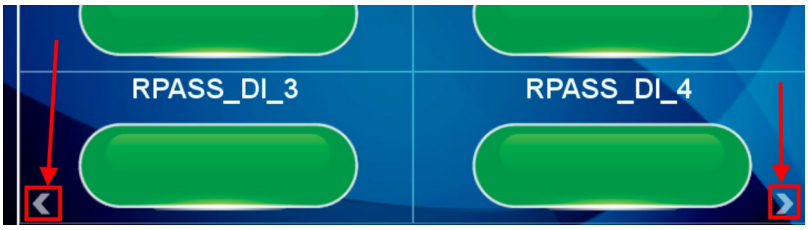
Widget macro graph (virtual display):



This is a virtual display, scroll through the pages of the virtual display by pressing the ">" arrow at the bottom right.
It is possible to place up to 2 virtual displays for each widget page.

6.3.1. PAGE CHANGE

To scroll from a page to the next, simply slide the finger to the left (this operation is called "swipe") as along the pages of a book;
Similarly, to return to the previous page, simply slide the finger to the right.
To change the page it is also possible to press a "forward" arrow and a "back" arrow:



6.4. TYPE OF WIDGET PAGE

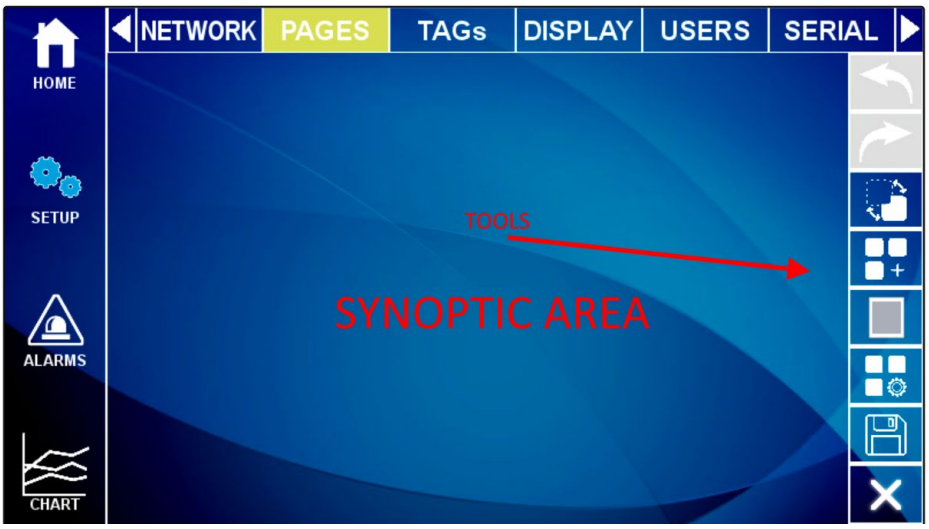
Represents the widget page, in this section it is possible to visualize the widgets related to the configured tags.
It is possible to choose among the various available grids, the widgets will be automatically positioned within the grid.
Each widget graphically represents the value of one or more TAGs.

6.5. SYNOPTIC PAGE TYPE

In a synoptic type page it is possible to freely move the widgets by adding graphics and also create animated synoptic ones.

Synoptic type pages can be freely mixed with widget type pages.

To create a synoptic page Select Pages and press the "Add Synoptic Page" button.
At this point a new page will open with tools on the left:



Here is the meaning of the tool icons:

 UNDO
Cancels the last operation performed

 REDO
Repeats the last operation cancelled by the UNDO

 BACKGROUND
Allows you to choose a graphic file to use as the background of the page

 ADD WIDGET
Adds a widget to the page

 ADD VIRTUAL DISPLAY WIDGET
Adds a virtual display widget

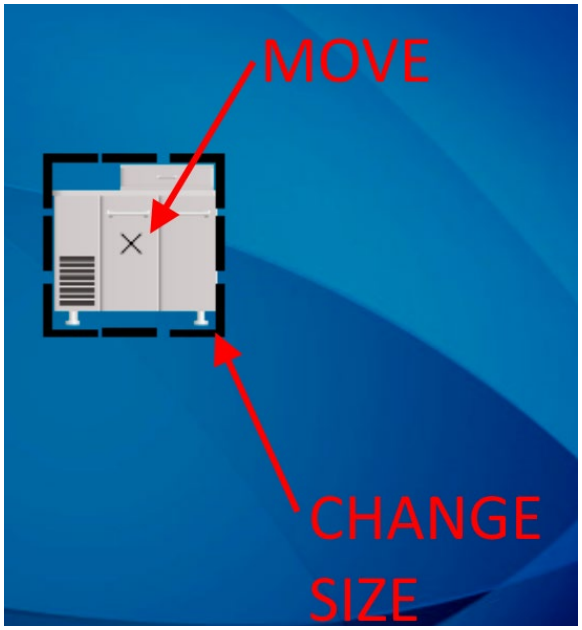
 WIDGET CONFIGURATOR
Allows the widget configuration

 SAVE PAGE
Saves the page changes

 EXIT
Exits the page

6.5.1. "ADD WIDGET" TOOL

The "ADD WIDGET"  button allows the addition of a widget on the page, once the widget has been inserted it is possible to move it by touching the widget in the central cross. To change the size of the widget, move the sides of the rectangle containing the widget:



When a widget is selected, a new series of tools appears on the right, the meaning of which is as follows:

 GRID

A grid is activated, moving the widgets they will follow the set grid.

 ALIGN

The widget is aligned

 CONFIG

Modification of the configuration parameters of the selected widget is allowed and viewed

 DELETE

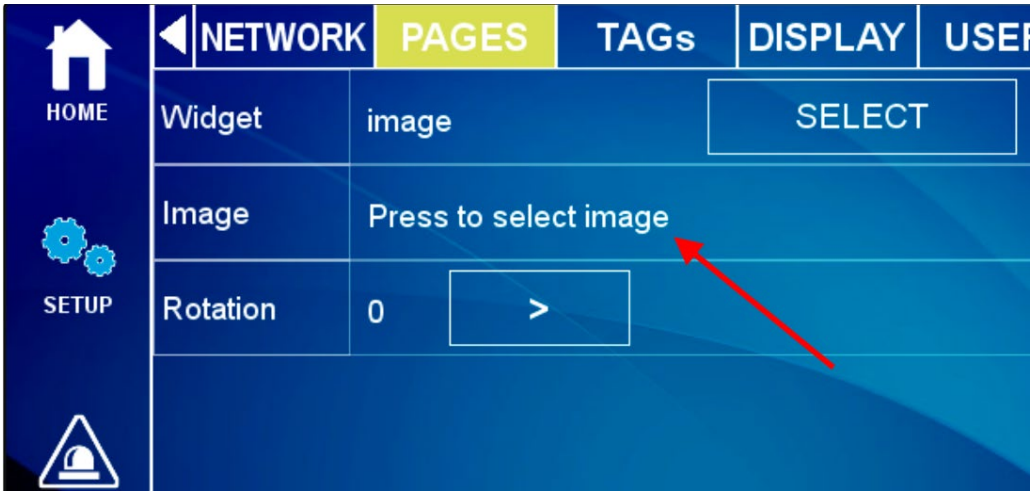
The Widget is removed from the page

 EXIT

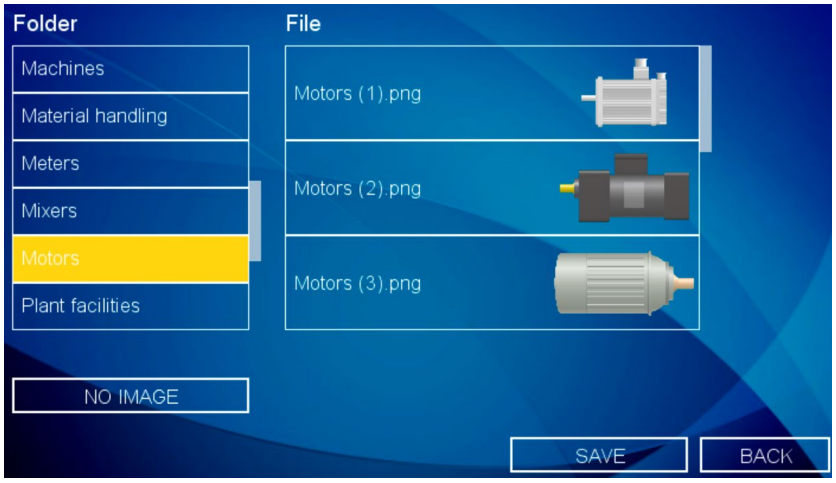
You return to the initial page of the synoptic

6.5.2. DATABASE OF SYMBOLS FOR THE SYNOPTIC PAGES

Inside the device is a database of graphic symbols that can be used in widgets.
The symbols are divided into categories. To access the symbols, select, for example, the "Image" widget:

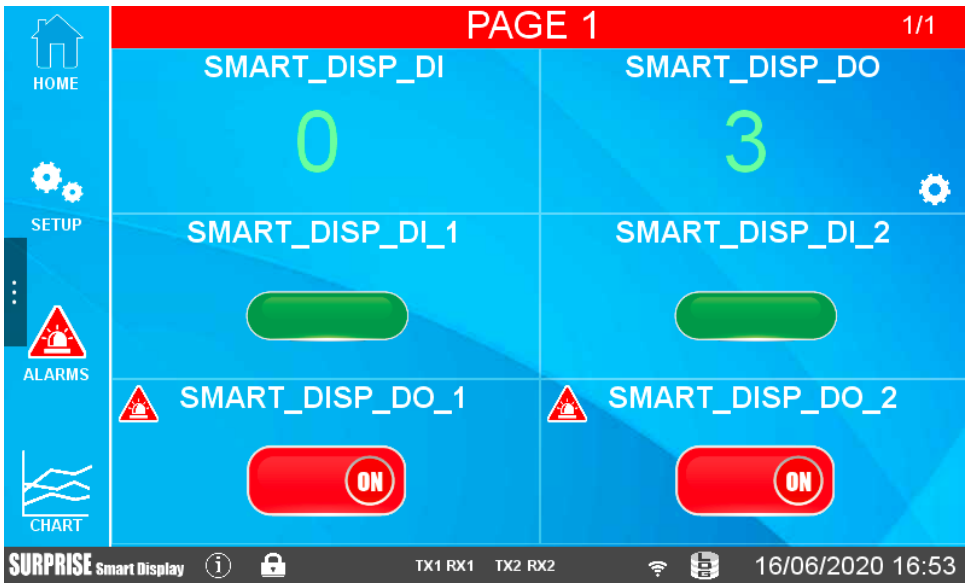


For example, selecting the "Motors" category displays the graphic files relating to engines:



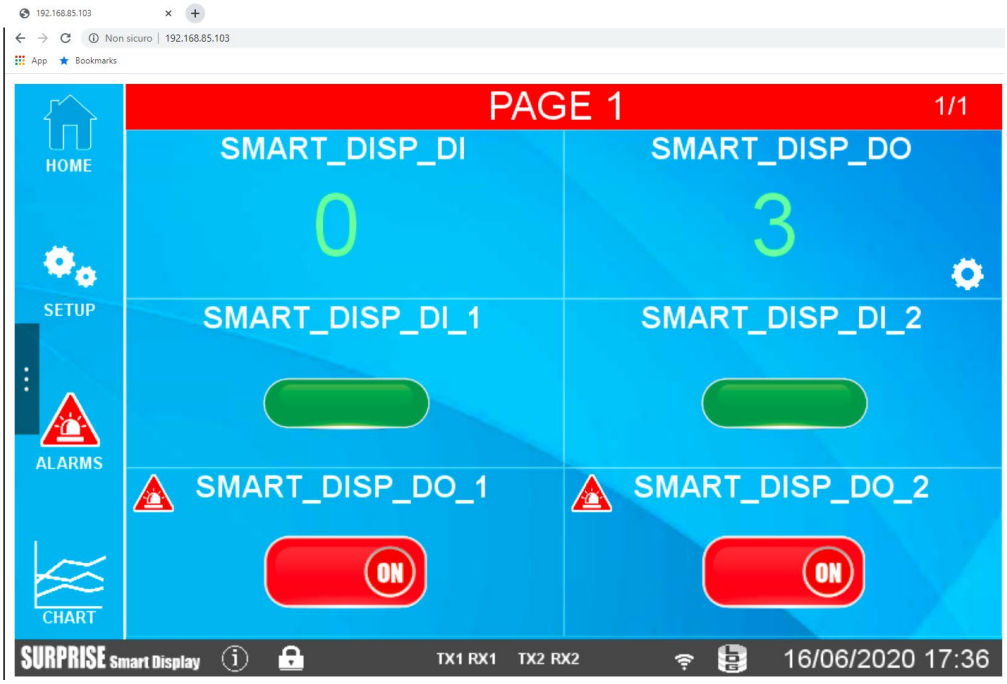
6.6. ALARMS

When an alarm occurs on at least one TAG, the title of the page is outlined in red and the faulty tags display the alarm icon, see the figure:



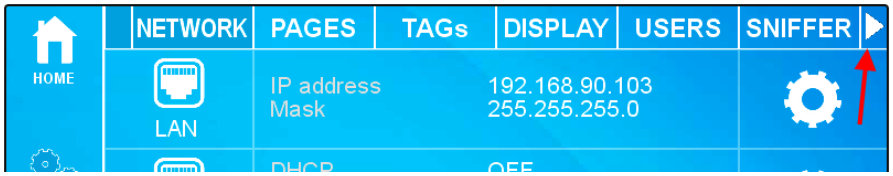
6.7. VIRTUAL DISPLAY

All the operations that can be done on the physical display can also be done connecting to the device web page via a web browser via port 80 (default).
To connect to the virtual display, enter the device's IP address into a browser on a PC or smart device:

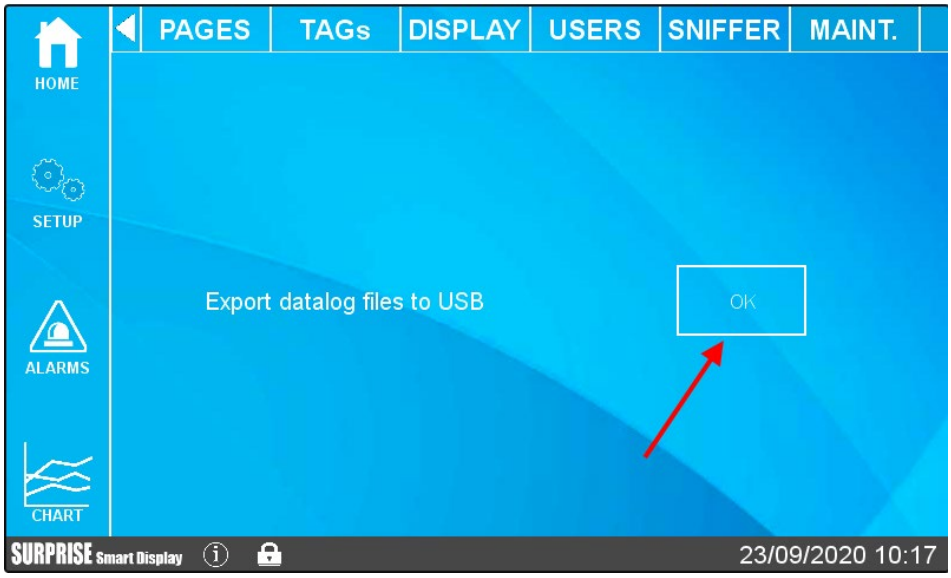


6.8. DOWNLOADING LOG FILES TO USB FLASH DRIVE

By inserting a USB stick in the HOST port it is possible to carry out a complete download of the files acquired by the datalogger.
To carry out this operation it is necessary to reach the "Maintenance" menu by tapping "SETUP" and then the arrow that extends the menu:



Now select "MAINT." and then press the relevant button to perform the operation:



At this point the system will download all the files acquired by the datalogger.
In the root of the USB stick there will then be many folders (one per day of recording) with the files related to that day (divided in turn into folders representing the active log groups). This functionality is also active via Webserver in the "TAG VIEW" section.

7. *INDUSTRIAL GATEWAY / ROUTER / FIREWALL*

The devices allow you to set the firewall, port mapping and other advanced features such as 1:1 NAT.
In addition to these features, you can also activate the industrial gateway function.

7.1. SERIAL ETHERNET GATEWAY

You can activate the available protocols to create gateways for industrial protocols (for example from/to Modbus RTU to/from Modbus TCP-IP). Or you can activate the transparent mode.

7.2. MODBUS ETHERNET TO SERIAL GATEWAY

The device can be configured to operate as a Gateway from Modbus Ethernet to Modbus Serial.
In this working mode, Modbus TCP Requests received from Ethernet interfaces are converted into Modbus RTU requests and sent to the serial interface; in the same way, Modbus RTU replies received from the serial interface are converted into Modbus TCP replies and sent back to the source network interface.

A Modbus Ethernet to Serial Gateway request can be activated for each of the available serial ports.
In this mode Modbus Gateway can support up to 50 simultaneous Modbus TCP connections. These connections can also be established through a VPN tunnel.

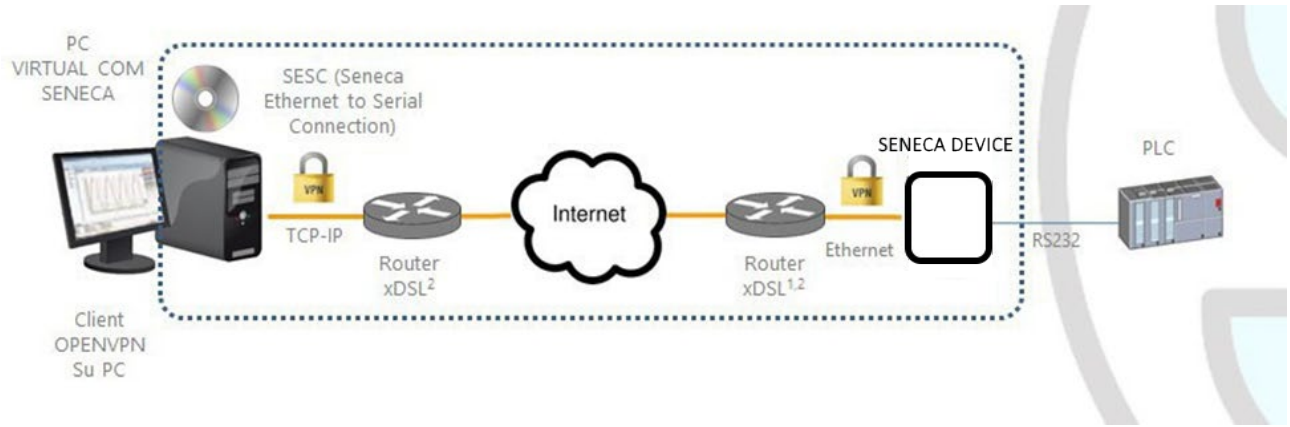
7.3. TRANSPARENT ETHERNET TO SERIAL GATEWAY

As an alternative to Modbus Ethernet to Serial Gateway, the device can be configured to operate as a "Transparent Gateway". The big difference between these two working modes is that while Ethernet to Serial works only with the Modbus protocol, Transparent Gateway can be applied to extend (transport) any serial communication (regardless of RS232/RS85 protocol) through the TCP/IP stack.
You can choose the following transparent gateway modes:

- Virtual COM (with RFC 2217 support)
- Point-to-point serial tunnel on TCP
- Point-to-point serial tunnel on UDP

Each mode will be fully described in the following paragraphs.

7.3.1. VIRTUAL COM WITH RFC 2217 SUPPORT



The Virtual COM with RFC 2217 support feature allows a PC application, which transmits data only on a serial line, to communicate with a remote serial device, using Ethernet/Internet; in other words, through the Seneca device, a PC and a serial device, located in distant sites, can communicate because they are directly connected.

In this mode, data sent over the LAN or WAN are received by the Seneca device and sent to the serial port; the response packets follow the reverse path.

The RFC 2217 support defines some features that allow the PC to set the properties (baud rate, data bits, stop bits and parity) of the serial port of the Seneca device remotely; so, when Virtual COM operating mode is selected for a port, the port is reconfigured independently from the previous settings and the values configured in the Seneca device are overwritten.

For the Virtual COM to work, a utility called "Seneca Ethernet to Serial Connection" must be installed on the PC. The TCP connection can be established through a VPN tunnel, as shown above.

Once the connection is established, a program using the virtual COM port will transmit the data to the serial port of the device; for example, Modbus RTU requests sent by a Modbus Master program will reach the Modbus slave devices connected to the RS485 bus of COM2.

Particular attention must be paid to the "Data Packing Interval" parameter, which can be set when the Virtual COM operating mode is selected: this parameter allows you to define the time interval, in milliseconds, used by the Seneca device as a criterion for packing the bytes of data received from the serial port before sending them to the network; in other words, when the Seneca device does not receive any more bytes from the serial port for the given time interval, it packs the received bytes and sends them over the established TCP connection; the optimal value to set for this parameter depends on the protocol that is transparently routed from the TCP/IP network to the serial line and vice versa.

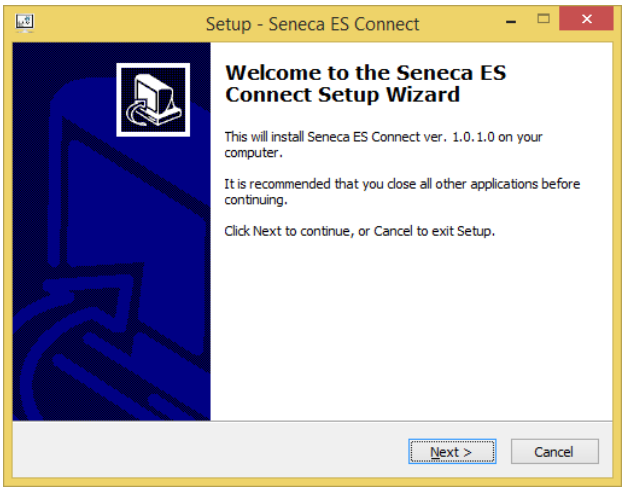
ATTENTION!

In Virtual COM operating mode only one serial port can be used

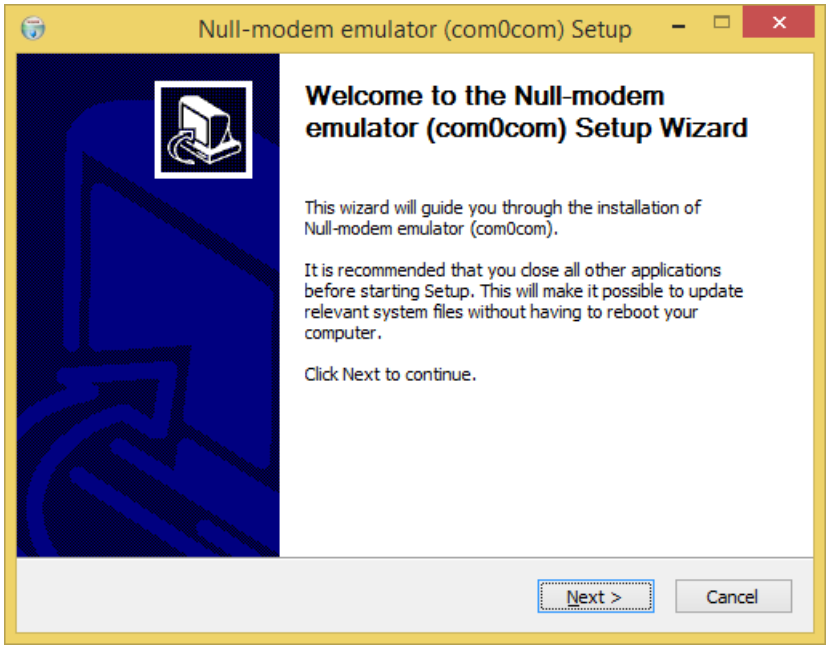
7.3.1.1. SENECA ETHERNET TO SERIAL CONNECT

7.3.1.1.1. INSTALLING THE SENECA SERIAL TO ETHERNET DRIVER

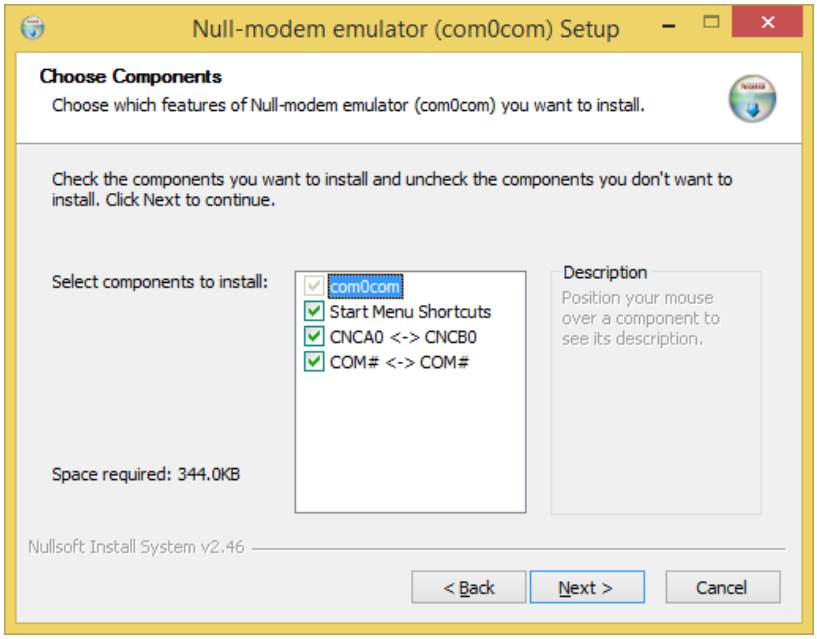
Seneca Ethernet to Serial Connect is compatible with 64 bit Windows systems.
Double-click on the installer



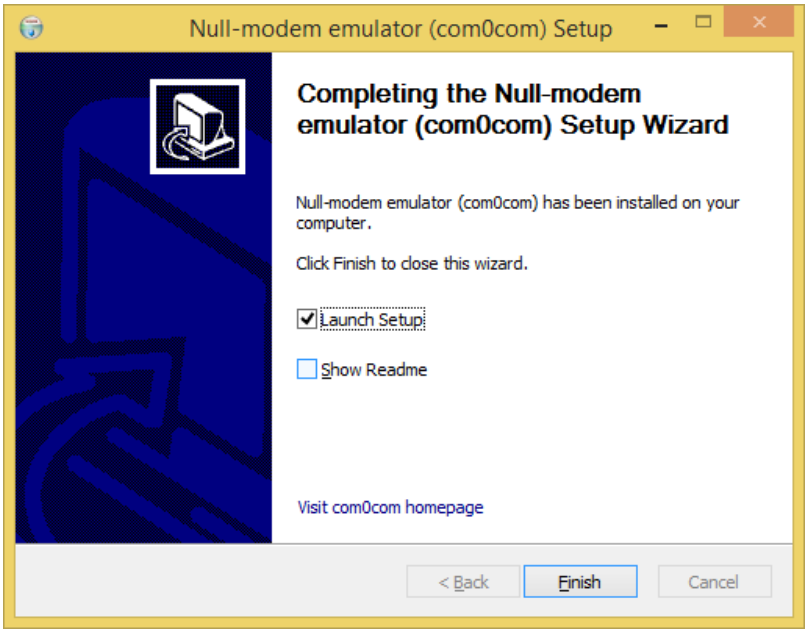
Then the com0com driver will be installed:



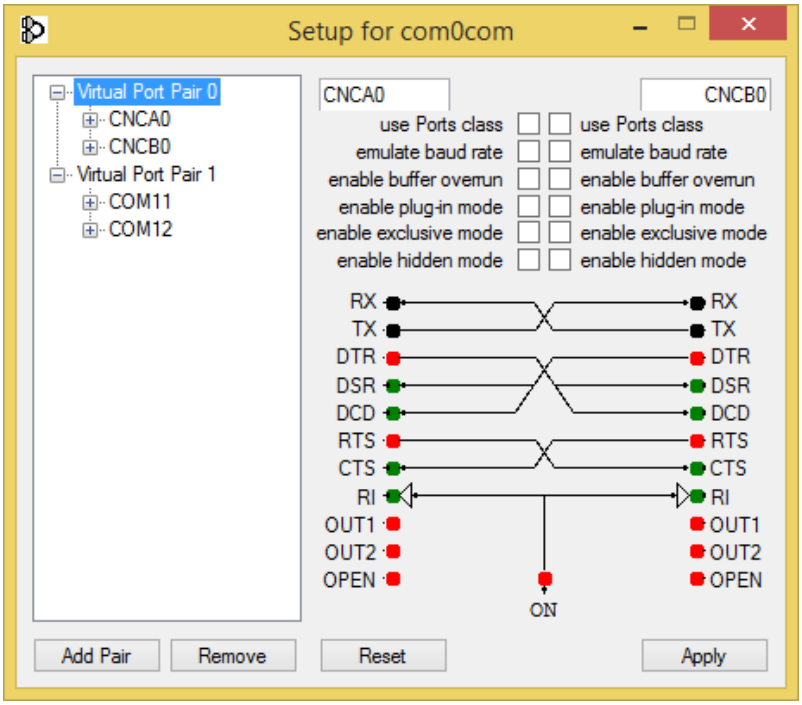
Select the virtual port names CNCA0<->CNCB0 and COM#<->COM#:



Now click on "Start Setup":



Press Finish, the com0com setup will open:



During installation two pairs of virtual COM are created:
CNCA0, CNCB0
and also:

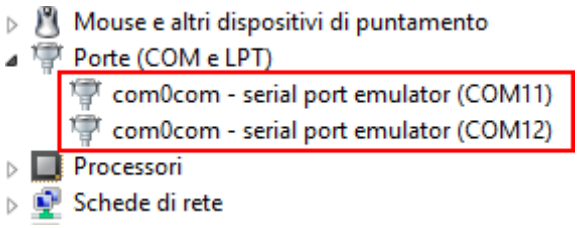
COM11, COM12 (note that com# may be different in your system).

The first pair can be used in software that supports CNCA names, the other in software that only supports Port Classes.

If you need to add more virtual ports, press the "Add Pair" button, then select whether or not you need a Class port.

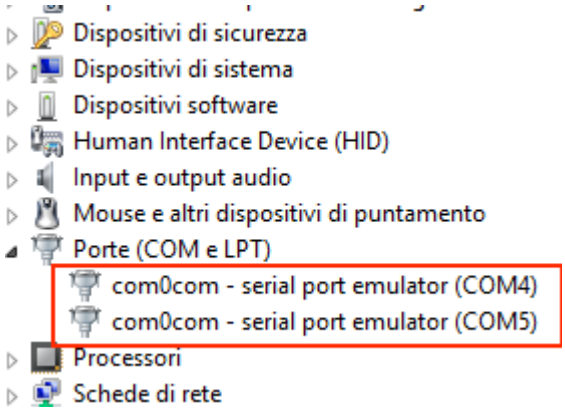
Confirm the driver installation with "Apply".

The pair of serial port emulators COM11-COM12 will be available in Device Manger



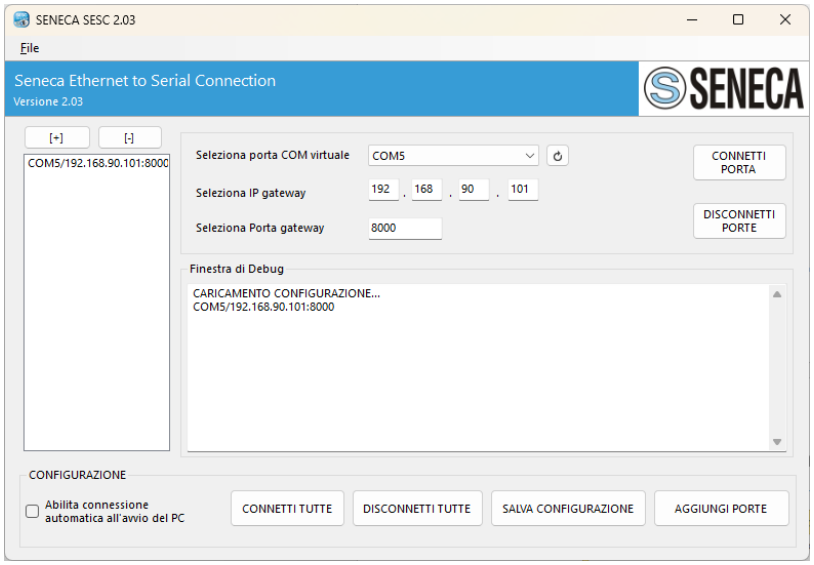
7.3.1.1.2. COM PORT SELECTION FOR SENECA ETHERNET TO SERIAL TO CONNECT

The driver installation will use the first 2 serial ports that are free (in our case the driver has created the pair COM4 and COM5):

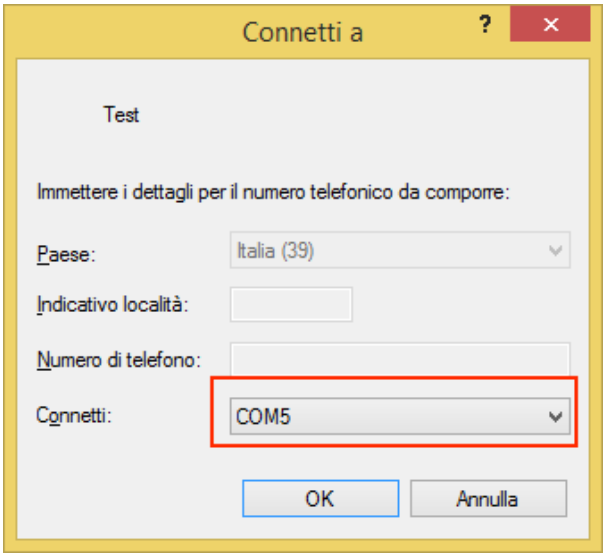


The software will use only one port (the correct port in the com0com setup), only com0com ports will be displayed.

Select COM5 in the Seneca ES connector:

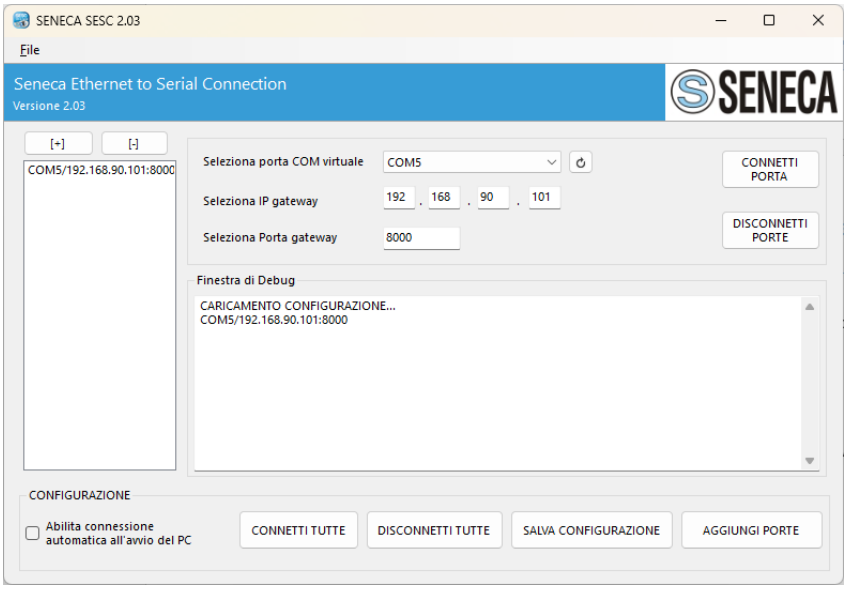


Now use the same COM port on application to use (e.g. in the terminal software)



COM5 is now connected to the Seneca device by a TCP connection to port 8000.

7.3.1.1.3. SENECA SERIAL TO ETHERNET CONFIGURATION



- Select the virtual COM port
- Select the IP address of the Seneca device
- Select TCP-IP port

Click on "CONNECT PORT".

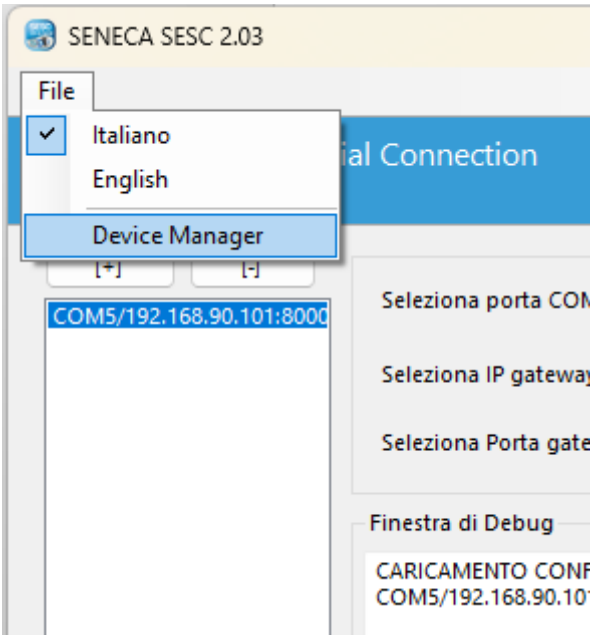
If you need to connect another serial com to another Seneca device, just press the “ADD PORT” button and then the [+] button to configure the new com port and, selecting it, enter the new IP address, then always press the “CONNECT PORT” button.

To disconnect all ports, click on “DISCONNECT PORTS”

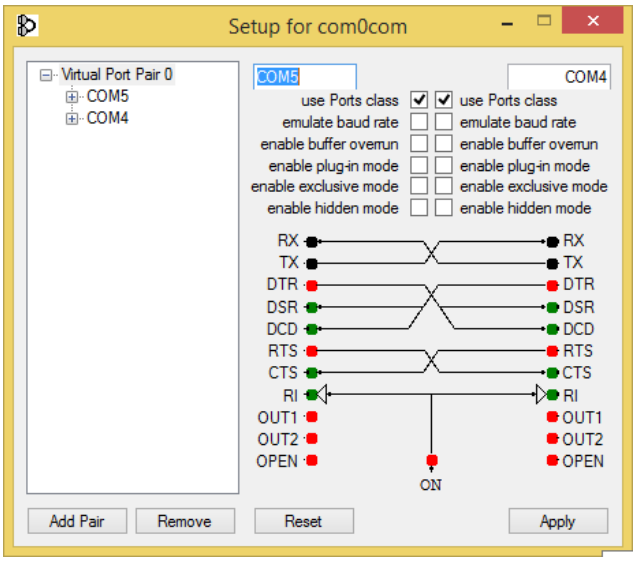
7.3.1.1.4. CHANGING THE PORT NUMBER

Older software applications can only use a small range of COM ports, so you may need to change the virtual COM port number.

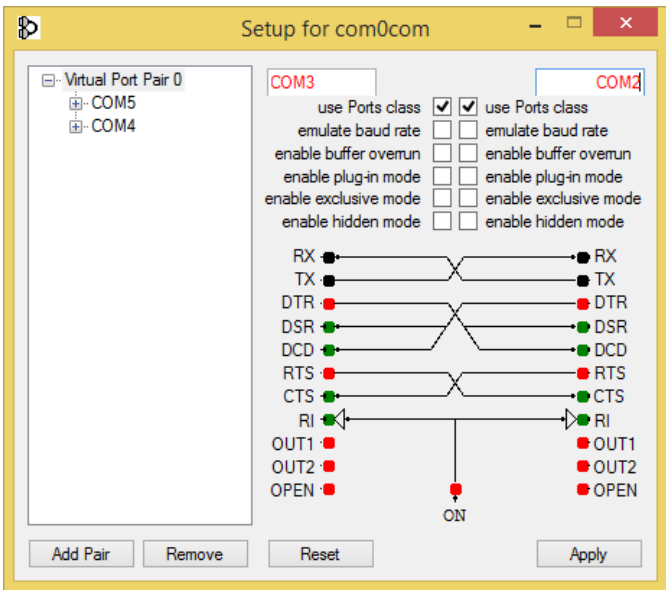
In our case the COM pair created is COM4/COM5, let's see the procedure to change it to COM2/COM3
Click on DEVICE MANAGER - File menu



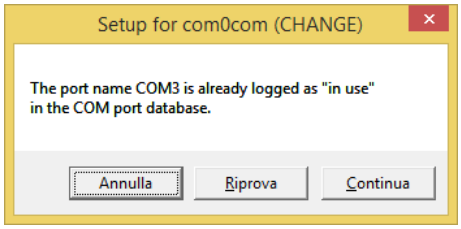
The com0com configuration window appears:



Now change COM5 to COM3 and COM4 to COM2, then click on "Apply":

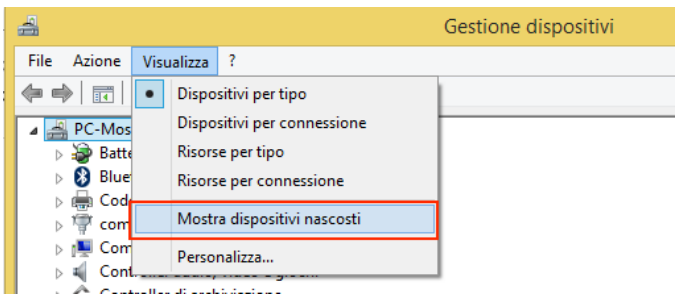


Sometimes the COM may be marked "in use":

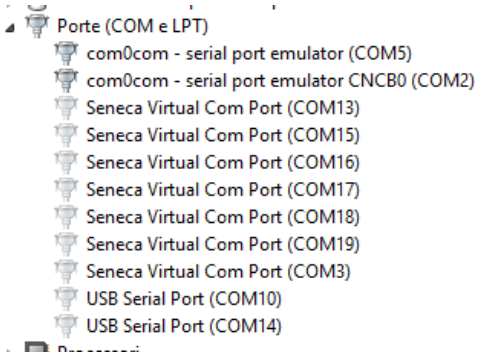


If you need to use this COM number, click "Continue", then go to device configuration.

Since the port is not connected, click on "Show hidden devices":

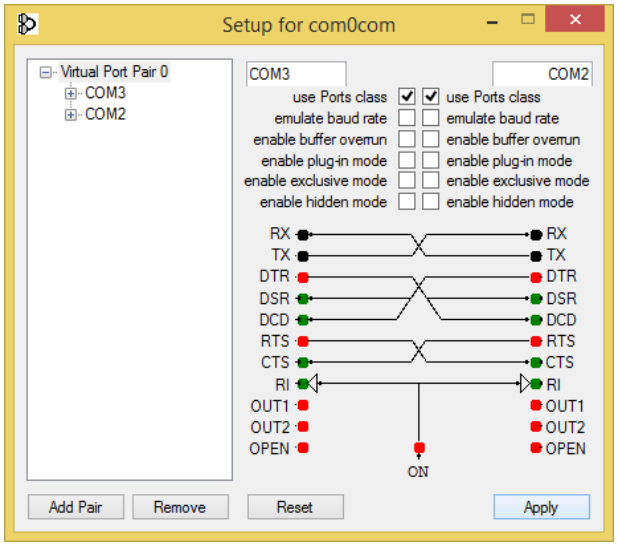


Now all unused ports are displayed in transparency (even our COM3):

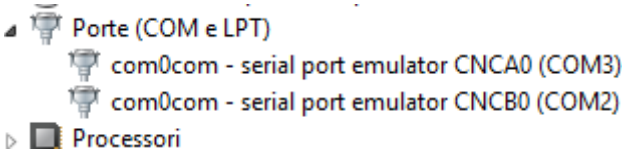


Now select the COM3 port and click on "Uninstall":

Now COM3 is free, and we can use it on the com0com setup:

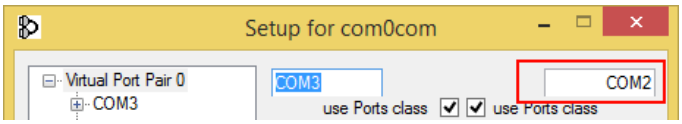


Finally click on "Apply", now the pair COM3/COM2 is created:



ATTENTION!

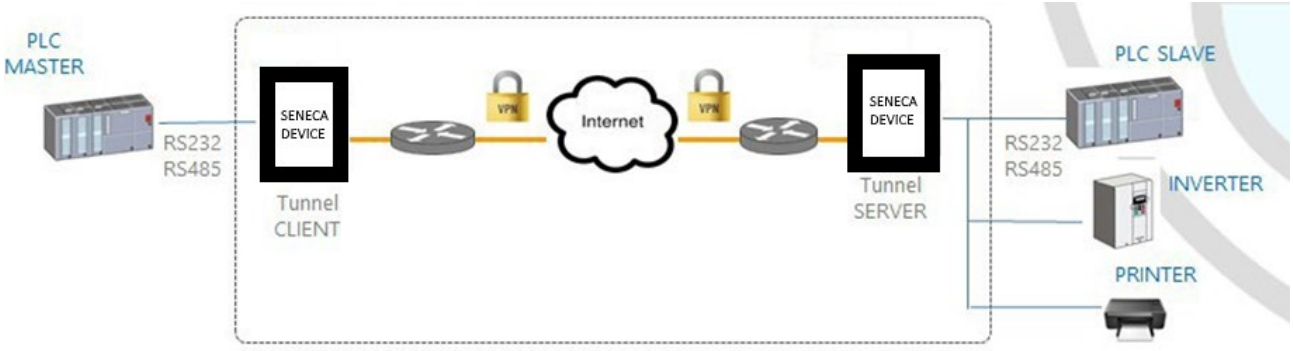
The Seneca Ethernet to Serial Connect Software always uses the correct port of the pair created in the com0com configuration (in our case COM2).



7.3.1.1.5. AUTOMATIC CONNECTION AT PC STARTUP

Once you have configured the necessary ports, you can automatically start the software at PC startup so that the connections are always active.
To do this, save the configuration with the appropriate button and check the enabling of automatic connection at PC startup:

7.3.2. SERIAL TUNNEL POINT ON TCP



The point-to-point serial tunnel allows to extend a serial connection between two serial devices (that support the same protocol) via a TCP/UDP connection.

In TCP operating mode, one of the two Seneca devices is defined as "Master" and the other is the "Slave": the first is a Tunnel Client, which receives data from the serial line and sends them to an outgoing TCP connection,

while the second is a Tunnel Server, which receives data from an incoming TCP connection and sends them to the serial line; in this mode a "tunnel" is established between the two serial ports.

During configuration, on the Master, you must set the destination IP address and the destination Port that defines the outgoing TCP connection; on the Slave, you must set the Listening Port on which the incoming TCP connection is accepted.
The tunnel can also be established through a VPN connection.

ATTENTION!

In Serial Tunnel Point-to-Point the operating mode on TCP, only one connection is accepted for a given serial port.

7.3.3. POINT--TO-POINT SERIAL TUNNEL ON UDP

The Serial Tunnel Point-to-Point operating mode on UDP is very similar to that of TCP.

The only difference is that none TCP connection is established and the serial data is carried by a UDP packet.
The configuration parameters are the same as for the serial tunnel over TCP.
Again, The UDP packet can also passes through a VPN connection

CAUTION

In Serial Tunnel Point-to-Point operating mode on UDP, only one connection is accepted for a given serial port.

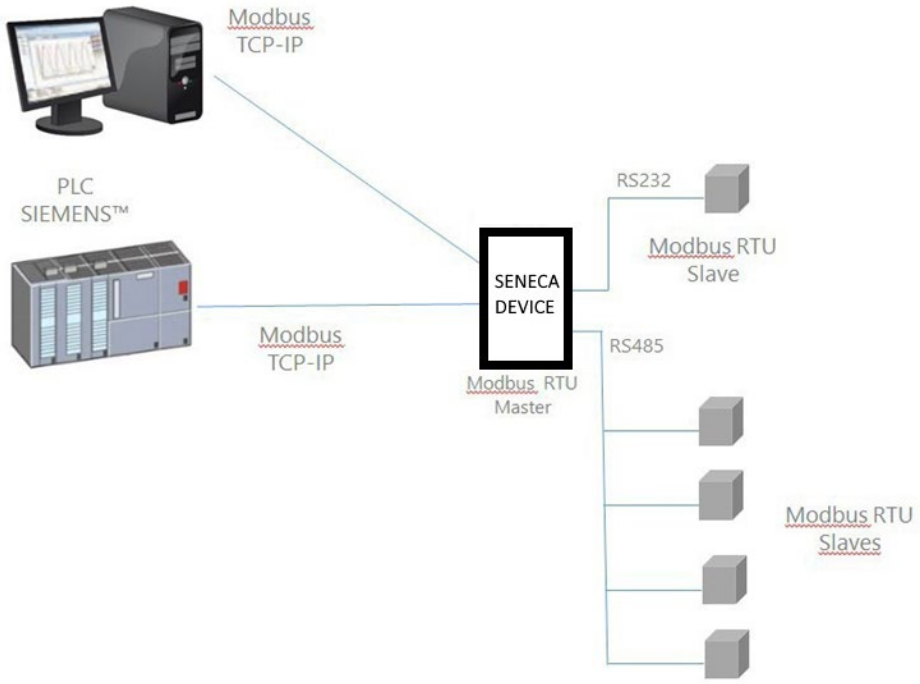
7.4. MODBUS GATEWAY WITH SHARED MEMORY

The device can be configured to work as a Modbus Gateway with Shared Memory: in this mode, a set of configured tags are periodically and continuously read by Modbus RTU Slave or Modbus TCP Server devices; these values are copied and made available in a shared memory.

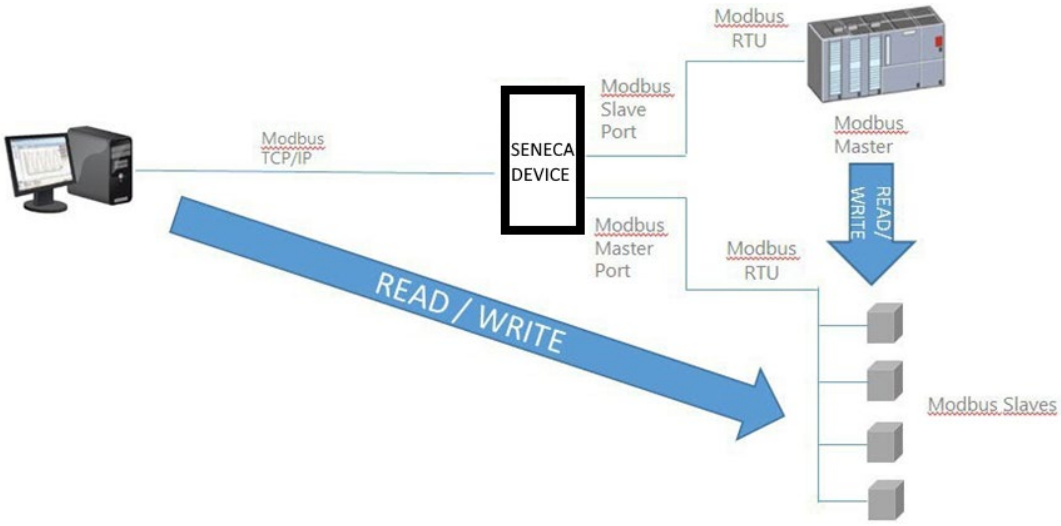
This mode supports up to 2000 tags and accepts up to 50 Modbus TCP Clients simultaneously, one Modbus TCP/IP Server (or slave) is always running on a configured TCP port.

For each of the available serial ports you can define the type of "Task": a serial port can be configured as Modbus RTU Master or Modbus RTU Slave or disabled.
In this way different combinations are possible.

In addition, tags can be read to/from up to 25 Modbus TCP Server.
Finally, you can define some tags that are related to the "embedded" digital I/O present in the device.
The following pictures show some typical scenarios.

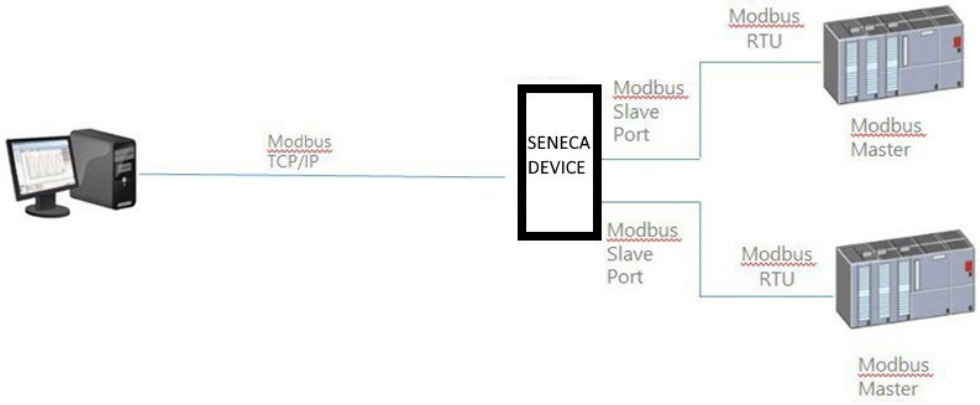


In the figure above, two serial ports are configured as Modbus RTU Master.



In this case, one serial port is configured as Modbus Slave and another is configured as Modbus Master.

When some registers acquired by the Modbus Slaves must be available for a PLC, which only supports the Modbus Master protocol, the device can be configured with one serial port defined as Modbus Slave (connected to the PLC) and another in Modbus Master (connected to the Modbus Slaves). The PLC Modbus RTU Master and the Modbus TCP client(s) will write/read the shared memory registers of the Seneca device, while the Modbus Gateway keeps the shared memory aligned with the Modbus Slaves registers.



In the figure above, two serial ports are configured as Modbus Slave and connected to a Modbus Master PLC port; in this way, the two PLCs and the Modbus TCP Client can write/read the shared memory to share data.

The Modbus Gateway Shared Memory mode provides some interesting features, as explained below.

In addition to the "classic" behaviour of the gateway, the tags can be configured to operate in "Bridge" mode; this mode allows you to "refresh" the tag values from the serial side only when the gateway receives Modbus TCP/RTU requests for those tags; this can be very useful when using RTU devices with "Fail safe" outputs, where it is necessary to cyclically write the outputs otherwise a fail would occur.
Modbus Gateway Shared Memory also performs request optimization, placing as many registers as possible in a single read/write request; it is possible to set the maximum number of registers in a request independently for each serial port/TCP Server and for read and write operations; this option can be useful for connecting RTU devices that support a maximum number of different registers on different serial ports.
Tag configuration can be created using a Microsoft Excel™ Template provided by Seneca; this template can considerably reduce configuration time, particularly when a large number of tags need to be configured.

8. DEVICE CONFIGURATION VIA CONFIGURATION WEBSERVER

The devices can be fully configured via a series of web pages.

8.1. "SUMMARY" PAGE

This page shows the main information about the status of the device and the user currently logged in.
It is also possible to view the installed firmware version and the activated options.

8.2. NETWORK AND SERVICES PAGE

Below are all the configuration parameters available on this page, with a short explanation and the default value of the parameter for each.

8.2.1. NETWORK SECTION

DHCP ON WAN
Allows you to activate or not DHCP on the "WAN" Ethernet port

LAN IP Address
Allows you to set the IP address of the "LAN" Ethernet port

LAN Network Mask
Allows you to set the mask of the "LAN" Ethernet port

WAN IP Address
Allows you to set the IP address of the "WAN" Ethernet port

WAN Network Mask
Allows you to set the mask of the "WAN" Ethernet port

Default Gateway
Allows you to set the default gateway for the "WAN" Ethernet port

DNS Mode
Allows you to set whether the DNS should be defined as static or taken from DHCP

DNS Server
Allows you to set the IP address of the DNS server

IP Configuration from Discovery

Allows you to select whether or not it is possible to change the IP configuration from the Seneca Discovery Device software (Attention: from the Seneca Discovery Device it is possible to change only the settings of the Ethernet/Wifi port to which you are connected). Please note that for the LAN port it is not possible to activate DHCP.
When the parameter is enabled, it is valid only until the first change is made by the Seneca Discovery Device software, as the firmware sets it to "Disable" immediately afterwards.
The factory default setting is "Enable."

8.2.2. WEB SERVER SECTION

Protocol
Allows you to select the protocol for the webserver, you can choose between https or none.

If you select https you can access the two web servers with the default addresses:

https://192.168.90.101 e https://192.168.90.101/gui

HTTPS Port
Allows you to set the port of the configuration webserver and the GUI

8.2.3. SFTP/SSH SERVER SECTION

Enable
Allows you to configure whether or not to enable the SFTP, SCP e SSH server protocol for accessing the device.

Port
Allows you to configure the port for the SFTP, SCP and SSH servers.

8.2.4. DATA FOLDER SHARING SECTION

Enable
Allows you to enable or disable the sharing of the data/folder from Windows devices via Samba protocol.

8.2.5. NETWORK REDUNDANCY SECTION

Enable
Allows you to enable and select the communication redundancy strategy.
You can choose between the following configurations:

OFF -> Redundancy is disabled
WAN/MOBILE -> If communication to the server set via the WAN Ethernet port is interrupted, it enables communication via Mobile modem (if available).

MOBILE/WAN-> If communication to the server set via the Mobile modem is interrupted, it enables communication via the WAN Ethernet port.
WAN/WIFI-> If communication to the server set via the WAN Ethernet port is interrupted, it enables communication via WIFI.
WIFI/WAN-> If communication to the server set via WIFI is interrupted, it enables communication via the WAN Ethernet port.

Ping Address
Allows you to set the address of the server to reach to use as a test for redundancy (attention: for redundancy to work, the server must respond to the ping request)

8.2.6. R-COMM SECTION (for R-PASS model only)

R-COMMM Available
If enabled, it activates control of the optional R-COMM module

R-COMM UPS Mode
It configures the UPS operating mode present in the R-COMM module.
Important: Check that the R-COMM model purchased has the "UPS" function before configuring these parameters.
If the R-COMM purchased does not include the UPS, this parameter must be set to "OFF".
OFF-> does not use R-COMM UPS to power R-PASS
Shutdown immediately-> in case of mains power failure closes the log files and performs a clean shutdown of R-PASS
Shutdown on low power-> in case of a mains power failure R-PASS continues to work as long as the battery is charged, when it is discharging it closes the log files and performs a clean shutdown of R-PASS

8.2.7. WATCHDOG SECTION

Enable
If enabled, it allows a reboot if the device remains blocked for a time equal to the watchdog timeout.

Timeout
It represents the time in seconds that the device can remain blocked before performing a reboot.

8.2.8. DEBUG LOGS SECTION

Enable
If enabled, it creates log files to be analysed by Seneca technicians.
The log files can be downloaded from the “Conf. Management“ page of the webserver

8.2.9. HARDWARE PORTS SECTION

SD
Enables or disables mounting of the inserted microSD memory.

USB
Enables or disables mounting of the inserted USB memory.

SERIAL CONSOLE
Enables or disables the console on the microUSB port.

8.3. PLC CONFIGURATION PAGE

8.3.1. STRATON PLC SECTION

Enable
Allows you to activate or deactivate the Straton PLC

TCP Port
Allows you to set the port for the connection with the Straton environment (IDE)

Redundancy Enable
Allows you to enable or disable the redundancy of the Straton PLC, 2 identical devices are created, one of which is automatically set as master and one as slave. The devices continuously exchange information between each other. If one becomes unavailable, the other is activated virtually without loss of continuity.
For more information, refer to the Straton PLC manual.

Redundancy IP Address
Allows you to set the IP address of the second PLC that is part of the redundancy.

License Key
Allows you to activate the Energy protocols (IEC61850, IEC60870-5-104 or IEC60870-5-101).
The key to be entered is sent by Seneca support in case of purchase of the respective licenses.

Retain Variables Enable
Allows you to configure how retain-type TAGs are to be managed (only if the Straton PLC is set to “shared” mode).

A Retain-type Tag is cyclically saved in a non-volatile memory so that, in the event of a device shutdown, it does not lose the acquired value.
A classic case is the value of an energy meter.

If set to OFF: the retain variables are managed by the firmware, if set to ON the management of the retain variables is done by the PLC.

8.3.2. Real-Time Behaviour SECTION

ENABLE
Enables Real Time mode in the PLC
By enabling this function, the operating system scheduler switches to Real Time mode and allows you to manage the PLC by reducing the Jitter of the PLC cycle times.
If a real time protocol is used in the PLC, it is recommended to enable this function.

8.3.3. ZNET4 COMPATIBILITY SECTION

ENABLE
This parameter allows you to enable or disable the features that allow configuration via the Z-NET4 software.

8.4. PLC MODBUS CONF. PAGE

8.4.1. Modbus TCP Client SECTION

These parameters allow you to set the IP address and the port of the Modbus TCP-IP server to which the Modbus TCP-IP client of the Straton PLC must connect without statically entering them in the IDE configuration.
This is very useful in case you need to create multiple PLCs that point to different Modbus TCP-IP servers without recompiling the Straton project each time.
In order for Straton to use these parameters, you must use the following text instead of the IP and Port value of the Modbus TCP-IP server:

mbtcpcli_param

at this point the IP address and the port will be replaced with the values set here.

IP Address
Allows you to set the IP address of the Modbus TCP-IP server to connect to via the Straton Modbus TCP-IP client.
Attention: in the Straton IDE you must enter the text:

mbtcpcli_param

instead of the IP address.

TCP Port
Allows you to set the Modbus TCP-IP server port to connect to via the Straton Modbus TCP-IP client.
Attention: in the Straton IDE you must enter the text:

mbtcpcli_param

instead of the TCP Port.

8.4.2. Modbus Pass-through SECTION

This function is only available if the Straton PLC is active

Enable

If enabled, it allows you to activate the modbus passthrough when the Straton PLC is running. Any modbus TCP-IP request arriving at the set port will be forwarded to the COM2 serial port.
Only if you use the Z-NET software to configure the device, you can change the COM2 port with another one.

TCP Port

This is the port used for Modbus passthrough.

8.5. SERIAL PORTS PAGE

The Mode parameter affects both the firmware Gateway and the Straton PLC, while the other properties of the serial ports refer to the Gateway features of the firmware of the devices. If the Straton PLC uses the same serial port, the parameters configured here (baud, bits no., etc.) will be overwritten and will therefore have no effect (those defined in the Straton PLC have priority).

8.5.1. COM1 SECTION (RS485/RS232/MBUS)

Mode

Selects the type of serial to use for COM1 (both for the PLC and for the firmware):
RS232, RS485 or RS232-MeterBus (via optional Z-MBUS device).

Baud Rate

This is the baud rate at which the serial port must operate.

Data Bits

This is the number of bits at which the serial port must operate.

Parity

This defines whether parity should be used and what type.

Stop Bits

This defines whether or not to use stop bits.

8.5.2. COM2 SECTION (RS485)

Mode

Selects the serial type to use for COM2 (both for the PLC and for the firmware): for COM2 you can only choose RS485.

Baud Rate
This is the baud rate at which the serial port must operate.

Data Bits
This is the number of bits at which the serial port must operate.

Parity
This defines whether parity should be used and what type.

Stop Bits
This defines whether or not to use stop bits.

8.5.3. COM4 SECTION (RS485)

This port is only available in the Z-PASS1/2-RT, Z-TWS4-RT models.

Mode
Selects the serial type to use for COM2 (both for the PLC and for the firmware): for COM4 you can only choose RS485.

Baud Rate
This is the baud rate at which the serial port must operate.

Data Bits
This is the number of bits at which the serial port must operate.

Parity
This defines whether parity should be used and what type.

Stop Bits
This defines whether or not to use stop bits.

8.6. WI-FI CONFIGURATION PAGE

This page is only available on models with a Wi-Fi port.

Mode

You can select from:
OFF: The WI-FI port is off
Station: The WI-FI is connected to an existing network
Access Point: The device creates a new WI-FI network to which devices can connect

SSID
If Mode is "Access Point" you can define the name of the new WI-FI network that the device will create.
If Mode is valid "Station" displays the SSID of the network you are connected to.

KEY MODE
Represents the encryption protocol to be used.

SCAN/APPLY
Allows, in Station mode, to select the WI-FI to connect to

8.7. I/O CONFIGURATION PAGE

In this page you can configure the IOs on board the device.

8.7.1. Digital I/O Configuration SECTION

This section allows you to configure the digital IOs.
Each device model has a different digital IO configuration:

SSD MODEL

Input/Output 1 Mode
It is possible to choose between:

Remote Connection Disable
The channel is set as INPUT and if set LOW it enables the possibility of opening a remote VPN connection with the device, if set HIGH every VPN connection is blocked.

General Input
The channel is set as general digital Input

General Output
The channel is set as general digital Output

Input/Output 2 Mode
It is possible to choose between:

Remote Connection Active
The channel is set as OUTPUT, if OPEN it means that no VPN connection is active. If CLOSED it means that a VPN connection is in progress.

Local alarm
The channel is set as an input that is typically connected to an external control PLC, when it is HIGH it indicates a general error that is visible remotely via the Seneca VPN BOX1 status interface, currently this parameter is not used by VPN BOX2.

Remote toggle
Currently not used

General Input
The channel is set as general digital Input

General Output
The channel is set as general digital Output

R-PASS MODEL

Input 1 Mode

It is possible to choose between:

Remote Connection Disable

The channel is set as INPUT and if set to LOW it enables the possibility of opening a remote VPN connection with the device, if set to HIGH every VPN connection is blocked

General Input

The channel is set as general digital Input

Input 2 Mode

It is possible to choose between:

Local alarm

The input is typically connected to an external control PLC, when HIGH it indicates a general error that is visible remotely via the Seneca VPN BOX1 status interface, currently this parameter is not used by VPN BOX2.

General Input

The channel is set as general digital Input

Input 3 Mode

General Input

The channel is set as general digital Input

Input 4 Mode

General Input

The channel is set as general digital Input

Output 1 Mode

It is possible to choose between:

Remote Connection Active

If OPEN it means that no VPN connection is active. If CLOSED it means that a VPN connection is in progress.

Remote toggle

Currently not used

General Output

The channel is set as general digital Output

Output 2 Mode

General Output

The channel is set as general digital Output

Output 3 Mode

General Output

The channel is set as general digital Output

Output 4 Mode

General Output

The channel is set as general digital Output

Z-PASS1/2 MODEL

Input/Output 1 Mode

It is possible to choose between:

Remote Connection Disable

The channel is set as INPUT and if set LOW it enables the possibility of opening a remote VPN connection with the device, if set HIGH every VPN connection is blocked.

General Input

The channel is set as general digital Input

General Output

The channel is set as general digital Output

Input/Output 2 Mode

It is possible to choose between:

Remote Connection Active

The channel is set as OUTPUT, if OPEN it means that no VPN connection is active. If CLOSED it means that a VPN connection is in progress.

General Input

The channel is set as general digital Input

General Output

The channel is set as general digital Output

Input/Output 3 Mode

General Input

The channel is set as general digital Input

General Output

The channel is set as general digital Output

Local alarm

The channel is set as an input that is typically connected to an external control PLC, when it is HIGH it indicates a general error that is visible remotely via the Seneca VPN BOX1 status interface, currently this parameter is not used by VPN BOX2.

Input/Output 4 Mode

General Input

The channel is set as general digital Input

General Output

The channel is set as general digital Output

Remote toggle

Currently not used

Input/Output 5 Mode

General Input

The channel is set as general digital Input

General Output

The channel is set as general digital Output

Input/Output 6 Mode

General Input

The channel is set as general digital Input

General Output

The channel is set as general digital Output

8.7.2. Analog I/O Configuration SECTION

Allows you to configure the analog inputs (not present in the SSD product)

Analog Input 1 Mode
You can choose whether to set the input as Voltage (0-10V) or Current (0-20mA) input.

Analog Input 2 Mode
You can choose whether to set the input as Voltage (0-10V) or Current (0-20mA) input.

8.7.3. Security Level SECTION

Service Disable
This parameter defines which access services are disabled when the "Remote Connection Disable" digital input is HIGH.

The possible values are:

VPN Connection: VPN connection block (Service VPN channel and Internet active)

VPN Service: VPN service channel block (active Internet)

Internet Connection: Blocking of internet access (both internet and VPN are blocked in the device)

SMS Service: The modem is turned off and therefore it is not possible to receive SMS messages.

8.8. REAL TIME CLOCK SETUP PAGE

This page allows you to set the device date/time parameters.
The date/time is maintained for a few days even without supplying power.

8.8.1. NTP SECTION

The Network Time Protocol, in acronym NTP, is a protocol to synchronize the clocks of the devices connected within a network. The NTP is a client-server protocol belonging to the application layer and listens on UDP port 123.

Enable
Enables or disables time acquisition from the set NTP servers. Synchronization occurs every 5 minutes.

Server primary
IP or FQDN address of the primary NTP Server

Secondary server
IP or FQDN address of the secondary NTP Server

Timezone
Time zone setting

8.8.2. RTC SECTION

In case of disabled NTP server, it is possible to manually set the date/time or acquire it directly from the connected PL.

8.9. GATEWAY CONFIGURATION PAGE

This page allows you to activate and configure the Ethernet-Serial Gateway you want to use.
For each serial port (depending on the device model, the number of serial ports is different) you can choose between:

Modbus Ethernet to Serial
This is a real-time conversion from Ethernet port to serial port from the Modbus TCP-IP protocol to serial RTU Modbus.

Transparent
This is a real-time conversion from an Ethernet port to a serial port independent of the protocol.
Modbus Shared Memory
In this mode, acquisitions are made from serial (towards a Modbus RTU slave) or from Ethernet (towards a Modbus TCP-IP server) and imported into an internal memory. This mode is essential for using the data logger, client protocols and the cloud.

> ***⚠ ATTENTION!***
>
> ***In order to use the data logger, client protocols (e.g. MQTT) and logical rules, you need to set the gateway operating mode to Modbus Shared Memory.***

None
The serial port is free or usable by the Straton PLC protocols (such as MeterBUS).

For more information on the Gateway operating modes, refer to the respective chapter of this manual.

8.9.1. Modbus Shared Memory SECTION

This section contains the configurations relating to access to the shared memory of the Modbus Shared Memory mode.

TCP Enable
This parameter enables/disables the Modbus Shared Memory Gateway service.
It is important to note that when this parameter is set to OFF, the Modbus TCP-IP server service is not running even if some serial ports are assigned to it.

TCP Port
Sets the listening port for the Shared Memory Modbus TCP server

TCP Connections Max Number [1-50]
Maximum number of TCP connections that can be accepted by the Modbus TCP server

Response Mode when Resource in Fail
This parameter defines how the response to a Modbus request (read) for a tag corresponding to a non-responding Modbus station is constructed; when mode is “Tag error value”, the value in the Modbus response is given according to the "Error Mode" / "Error Value" parameters in the tag definition; when mode is "Exception", the response contains an exception with the value 11 ("Gateway target device failed to respond").

Diagnostic Area Type
Select whether diagnostics can be accessed via Modbus Holding Registers or Modbus Input Registers.

Diagnostic Area Address

Defines the starting register of the TAG diagnostic area.
The diagnostic area reserves a bit for each configured tag (125 registers) and provides the FAIL/OK status:

Bit value on 0 -> means Tag reading error (or tag not configured)
The bit value on 1 -> means Reading tag OK

Therefore, if you need to check the error status of the first 10 tags using the default area (9001 Holding Registers), you must read the 49001 registry.
For instance, if the value of the register is:

0x3DB = 987 = 0000 0011 1101 1011
Tag 1 = OK
Tag 2 = OK
Tag 3 = FAIL
Tag 4 = OK
Tag 5 = OK
Tag 6 = FAIL
...
Note that one register before and one register after the diagnostic area will be reserved (by default registers 49000 and 49126 or 39000 and 39126).

Internal Write Functions

Allows you to choose how the TAGs are written to the Modbus registers of the slave or server devices.
This includes writings with the “SET” button of the TAG webserver page or writings of the logical rules.

8.9.2. Modbus Ethernet to Serial e Modbus Shared Memory SECTION

This section allows you to configure the Slave ID address (station modbus address) to which the device responds with its embedded IOs.
The registers representing the I/Os are accessible via Modbus TCP-IP or RTU protocol.
The addresses of the modbus registers vary depending on the model and are defined in the respective chapter of this manual

8.9.3. COM0, COM1, COM2, COM4 SECTION (DEPENDING ON THE MODEL)

Here you can set the parameters related to the gateway mode that has been chosen for each serial port.
The COM0 port is available when a USB-to-serial converter is connected

8.9.3.1. COM0 (USB)

Depending on the mode chosen for the port (in this case only Transparent mode is available) you can set the parameters:

Operating Mode
For the COM0 port, only the “Virtual COM” mode can be selected.

Listen Port
It is the port on which the Virtual port mode server works.

Data Packet Interval (ms)
It is the time interval that marks the end of a packet, this parameter must be set based on the type of protocol that is transiting

8.9.3.1. COM1 (RS232/RS485) COM2 (RS485) COM4 (RS485)

Depending on the mode chosen for the port, the following parameters are available

8.9.3.1.1. COM1/COM2/COM4 Modbus Ethernet to Serial

Allows you to set the parameters of the Ethernet to Serial Gateway mode

Enable
Enables or disables Ethernet to Serial mode on the serial port

Port
Sets the TCP port on which the Ethernet to Serial gateway will operate

Response wait time [ms]
Sets the serial wait time to declare a timeout

8.9.3.1.2. COM1/COM2/COM4 Transparent

Allows you to set the operation of the transparent mode.

Operating Mode
For COM1/COM4 ports you can choose between:

VIRTUAL COM

SERIAL TUNNEL POINT TO POINT ON TCP

SERIAL TUNNEL POINT TO POINT ON UDP

8.9.3.1.2.1. COM1/COM2/COM4 VIRTUAL COM

Allows you to set the parameters of the Ethernet to Serial Gateway mode

Enable
Enables or disables Ethernet to Serial mode on the serial port

Port
Sets the TCP port on which the Ethernet to Serial gateway will operate

Response wait time [ms]
Sets the serial wait time to declare a timeout

8.9.3.1.2.2. COM1/COM2/COM4 SERIAL TUNNEL POINT TO POINT ON TCP/UDP

Tunnel Role
Sets the tunnel as master or slave

Destination Address
If the Tunnel Role is master it is the IP address of the remote Tunnel Role Slave

Destination Port
If the Tunnel Role is master the Listen Port of the Tunnel Role is slave

Listen Port
If the Tunnel Role is set to slave it is the listening port of the remote master tunnel

8.9.3.1.2.1. COM1/COM2/COM4 MODBUS SHARED GATEWAY

Task
Allows you to select the type of task Modbus Shared Gateway that must be executed on the selected serial port between:
None, Master, Slave or Sniffer

None
No active task

Master
The gateway's Modbus RTU master is active to acquire data from Modbus RTU slave devices.

Slave
The gateway's modbus RTU slave is active to accept connections from a modbus RTU master

Sniffer
The serial sniffer is active, that is, it acquires the modbus RTU protocol from the serial port passively. It is used in existing systems (when there is already a modbus master and one or more modbus slaves) and you want to acquire data passively.

Slave Address
In Task = Slave mode, it is the value of the slave address (station address) that the serial port must assume.

Timeout (ms)
In Task = Master mode it is the response Timeout for Modbus RTU requests, in milliseconds

Delay between Polls (ms)
In Task = Master mode it is the interval between Modbus RTU requests, in milliseconds

Read/Write Retries
In Task = Master mode this is the maximum number of retries for Modbus RTU requests; this always applies to write requests; for read requests, it only applies to tags with "Tag mode" = "BRIDGE"

Multiple Read Max Number
In Task = Master mode, this is the maximum number of Modbus registers that can be read in a single Modbus RTU request; it is used to reduce the number of read requests sent on the serial bus (thanks to this parameter the firmware autonomously performs an optimization)

Multiple Write Max Number
In Task = Master mode, this is the maximum number of Modbus registers that can be written in a single Modbus RTU request; it is used to reduce the number of write requests sent on the serial bus (thanks to this parameter the firmware autonomously performs an optimization)

Validity Timeout
In Task = sniffer mode if a certain tag is not seen refreshed in the communication for the set time then it is set to FAIL.

8.10. VPN CONFIGURATION PAGE

This page allows you to configure a VPN, Seneca devices support two types of VPN: VPN BOX or OPEN VPN. For more information on the VPN BOX server, refer to the VPN chapter in this manual.

VPN MODE
Allows you to choose the type of VPN server to connect to, you can choose between OPEN VPN or VPN BOX.

The installed version of OPEN VPN is 2.4.7

8.10.1. VPN FILES SECTION

In the case of a VPN connection with an OPEN VPN server, this section allows you to upload the configuration file and any certificates.
The configuration file must contain all the information needed to configure the behaviour of Open VPN.

The main configuration options are:
- whether the device will function as a client or server (generally, it will be a client)
- the transport protocol (UDP or TCP)
- the IP address of the server / host name and port
- the files needed to perform authentication procedures
- etc...

This file has the extension ".ovpn" (on Windows systems) or the extension ".conf" (on Linux systems).
Regardless of the original name, it will be renamed to "ovpn.conf" on the device.
This is the only mandatory file, i.e. if this file has not been uploaded to the device the VPN cannot be enabled.
As mentioned in the Web page, in the options that require a file argument, only the file name, without path, must be provided, as in the following example:

`ca ca.crt` **OK**

`ca /home/config/vpn/ca.crt` **FAIL**

Two other important rules that must be followed are:
- the "dev" option must be: "dev tun0" or "dev tap0".
- the "log" option must be omitted (so that logs are written to syslog)

For more information about the OPEN VPN configuration file, please refer to the OPEN VPN 2.4 documentation at the link:

https://openvpn.net/community-resources/reference-manual-for-openvpn-2-4/

CA CERTIFICATE

This file must contain the certificate of the certification authority (CA) and has the extension .crt.
This is required when the configuration file contains the "ca" option.

CLIENT CERTIFICATE

This file must contain the client certificate and has the extension .crt.
This is required when the configuration file contains the "cert" option.

CLIENT KEY

This file must contain the client key and has the extension .key.
This is required when the configuration file contains the "key" option.

ADDITIONAL FILE

This file can be of any type and may be required for configuration options other than "ca", "cert" and "key".
Note that you can upload more than one additional file.

You can choose files from your PC to select files and send them to the device by pressing the “UPLOAD” button.
When loading is complete, a results page is displayed
You can check which VPN files are stored on your device by clicking the "SHOW VPN STATUS" button,

As the web page recalls, VPN files can be downloaded from the device, if necessary, via FTP / SFTP; they can be found in the /home/config/vpn directory.

You can clear all VPN files by clicking the “RESET” button; a pop-up will appear, asking for confirmation.

When you press the "SHOW VPN STATUS" button, a third section called "VPN Status" appears, which shows:

- The "Connection Status" of the VPN (i.e. "Stopped" or "Running")
- the IP address assigned to the VPN interface when "Connected", the "dummy" IP address "0.0.0.0" when "Disconnected".
• the "OpenVPN Status" (i.e.: "Stopped" or "Running")
- the number of packets / bytes received by the VPN interface when connected; "0/0" when disconnected
- the number of packets / bytes sent to the VPN interface when connected; "0/0" when disconnected
- VPN files stored on the device

Important status information is given by the "OpenVPN Status" field; if the VPN is enabled ("ON"), but this status is "Stopped", this means that the Open VPN process cannot be started correctly: probably, the configuration file contains some errors or, perhaps, some options not supported by the OpenVpn implementation of the device.
You can update the VPN status by clicking the "REFRESH" button.
Finally, you can hide the "VPN Status" section by clicking the "HIDE VPN STATUS" button.

8.10.2.OPEN VPN SECTION

Enable
Flag to enable/disable the “Open VPN” feature

Allowed Interface
Allows you to force the VPN connection through the specified interface.

Reply on WAN to packets coming from WAN
If enabled, it allows responses to packets coming from the WAN interface to be sent to the same interface and not (for example) via the VPN.

8.10.3.VPN BOX SECTION

Enable
Flag to enable / disable the "VPN Box" feature, i.e. the procedure / protocol that allows the device to configure the VPN, interacting with the "VPN Box" server (see "VPN Box User Manual")

Server
IP or FQDN address of the "VPN Box" or “VPN Box 2” server

Password
Password to access the "VPN Box" server

Tag Name
Mnemonic name used to uniquely identify the device

When you click the "SHOW VPN STATUS" button, a new section called "VPN Status" is displayed, showing:
- VPN connection status

- the VPN IP address assigned to the device this line is not displayed for the VPN "Point-to-Point (L2)" box, as no IP address is assigned to the VPN interface
- the status of OpenVPN
- the number of packets / bytes received by the VPN interface
- the number of packets / bytes sent to the VPN interface
• the Type of VPN BOX, which can be "Point-to-Point", "Point-to-Point (L2)" or "Single LAN"
• the status of the VPN BOX, if the VPN box is enabled
- the username of the connected user, if any

The following table gives a brief explanation of the possible "Result" and "Status" strings:

Result	Status	Meaning
Error (Unexpected response)		A response code has been received that is not managed by the device (should never occur)
Error (No response from VPN Box)		No response received from VPN Box (response timeout)
Error (Invalid response from VPN Box)		A response was received whose content is not valid for the device (should never occur)
Error (Wrong password)		The password set on the device is incorrect
Error (License Limit Reached)		The maximum number of devices allowed by the license is already registered on VPN Box
Error (VPN Box not configured)		The VPN Box has not yet been configured
Error (Generic error)		A generic error has occurred on VPN Box
OK		The device has just been registered on VPN Box
OK	New	The device is registered on VPN Box, but not yet configured (only "single LAN")
OK	Configuration updated	The device configuration has just been updated
OK	Configured	The device is correctly configured and available for VPN connection
OK	Ban	The device has been "banned
OK	Not found	The device is not known to VPN Box; this happens when the device registration is deleted on VPN Box
OK	Unknown	The device has an unknown status in VPN Box (should never occur)
OK	Not bound	The "tunnel" between device and VPN Box is not active; this may occur when the tunnel port is blocked (not open) in the ADSL router on the VPN Box side (only "Point-to-Point")
OK	Unexpected status	A status code has been received that is not managed by the device (should never occur)

8.11. OPC-UA SERVER CONFIGURATION PAGE

In this page, you can set parameters related to the OPC Unified Architecture (OPC-UA) server integrated into the gateway.

The device's OPC-UA server "exports" the Modbus Shared Memory Gateway tags; therefore, using an OPC-UA client software, it is possible to read / write tags using the OPC-UA protocol.

NOTE: for all variables on the OPC-UA server the namespace-id is set to "1".

8.11.1.OPC- UA Server Conf. SECTION

Enable
Enables/Disables the OPC-UA server, once enabled the server is available at the URL:

opc.tcp://IP_Address:Port/

Port
Sets the port for the OPC-UA server.

Username
Username for server access

Password
Password for server access

Security Policy
It is possible to choose between:

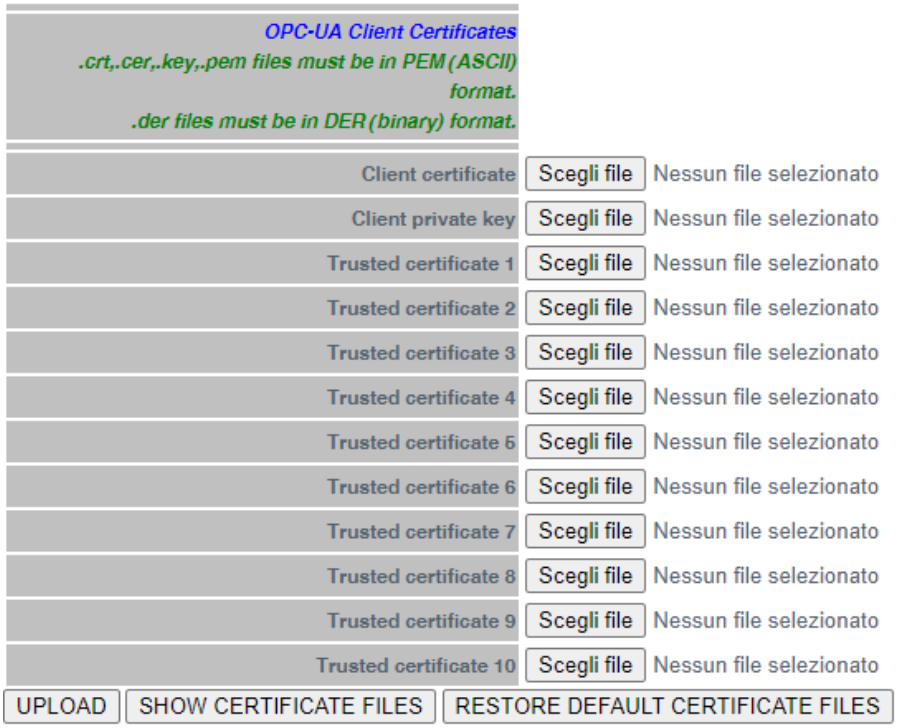
“None”
“Basic128Rsa15”
“Basic256Sha256”

8.11.1.1.OPC- UA SERVER CERTIFICATES SECTION

A default pair of certificates is already included in the product, you can also add your own certificates with the appropriate buttons.

8.12. OPC-UA CLIENT CONFIGURATION PAGE

In this page you can upload the server connection certificates for the OPC-UA client.



The "Choose File" button selects the certificate. These are only uploaded to the device after pressing the "Upload" button.
The "Show Certificate Files" button allows you to view the uploaded certificate files.
The "Restore Default Certificate Files" button allows you to restore the default certificate files.

8.13. SNMP CONFIGURATION PAGE

This page describes the configuration of the SNMP Agent.
The SNMP V2C version is supported.
The protocol can only be used if the Straton PLC is enabled.

8.13.1.GENERAL CONFIGURATION SECTION

Enable
Enables or not the SNMP protocol

Port
Port used by the SNMP protocol

Trap Type
Selects the type of Trap to use

Trap Port
Port used by Traps

Allow access from any host
When this parameter is disabled, access will be allowed only to the hosts listed below with "Access" selected.

8.13.2.COMMUNITIES SECTION

Name
Community identifier

Read
Provides Read properties to the selected Community

Write
Provides Write properties to the selected Community

8.13.3.HOSTS SECTION

IP Address
Allows you to define the Host IP

Community
Allows you to define which community the Host is associated with

Access

If Flagged, it allows the host to access the SNMP Agent

Trap
If Flagged, it allows the host to receive Traps from the SNMP Agent

8.14. USERS CONFIGURATIONS PAGE

This page shows the configuration (user/password) of all the accounts available for access to the Webserver and the Display.
You can only enter one user per type.

WEB / DISPLAY ADMINISTRATOR
This is the account that allows all operations both on the configuration webserver and on the one relating to the display (and to the display on models equipped with it).

WEB / DISPLAY OPERATOR
This is the account that allows access only to some pages of the configuration webserver, while in the display webserver and in the physical display it allows you to block access to the setup menu.

WEB / DISPLAY GUEST
This is the account that allows access to almost all pages except for the advanced maintenance pages (for example, it does not allow access to the “FW Upgrade” and “Configuration Management” pages). It can view all configuration parameters and status information, but cannot change any parameters.
Consequently, on all pages, the “APPLY” buttons (and any other buttons used to make changes) are disabled.

FTP USER
This is the account for accessing the FTP server of the device.

8.15. ROUTER CONFIGURATION PAGE

On this page you can change the parameters related to the functionality of the router.

Router Enable
Enable/Disable router functionality

DNS Enable
Flag to enable/disable the DNS forwarding service

DHCP Server Enable
Flag to enable / disable DHCP service (DHCP server)

DHCP First Address

DHCP Last Address
These parameters define the range of IP addresses assigned by the DHCP server to requesting clients

DHCP Lease Time (min)
Validity time interval for IP address assignment, in minutes.

Use Local Addresses Through VPN/Enable
Flag to enable/disable access to the device and others that are connected to the LAN, using their local IP (LAN) addresses

Mobile network firewall
Allows you to enable or disable the firewall on the mobile network (if available).

Bandwidth limitation
Allows you to set a bandwidth limitation on network interfaces:
It is possible to choose among Unlimited, 20 Mbit/s (default), 10 Mbit/s, 1 Mbit/s

UDP Packets limitation
This parameter allows you to enable/disable the limitation of the number of UDP packets entering the device.
Attention! If a UDP protocol VPN is active (default for VPN BOX2), set the parameter to OFF.

8.16. PORT MAPPING RULES PAGE

On this page you can set up port mapping rules (also known as "virtual servers").

Protocol
This parameter defines the transport protocol (or port type) affected by the rule: TCP, UDP or both

External Port
TCP or UDP port to which a packet was originally sent

Server IP Address
IP address to which the received packet is forwarded

Internal Port
TCP or UDP port to which the received packet is forwarded

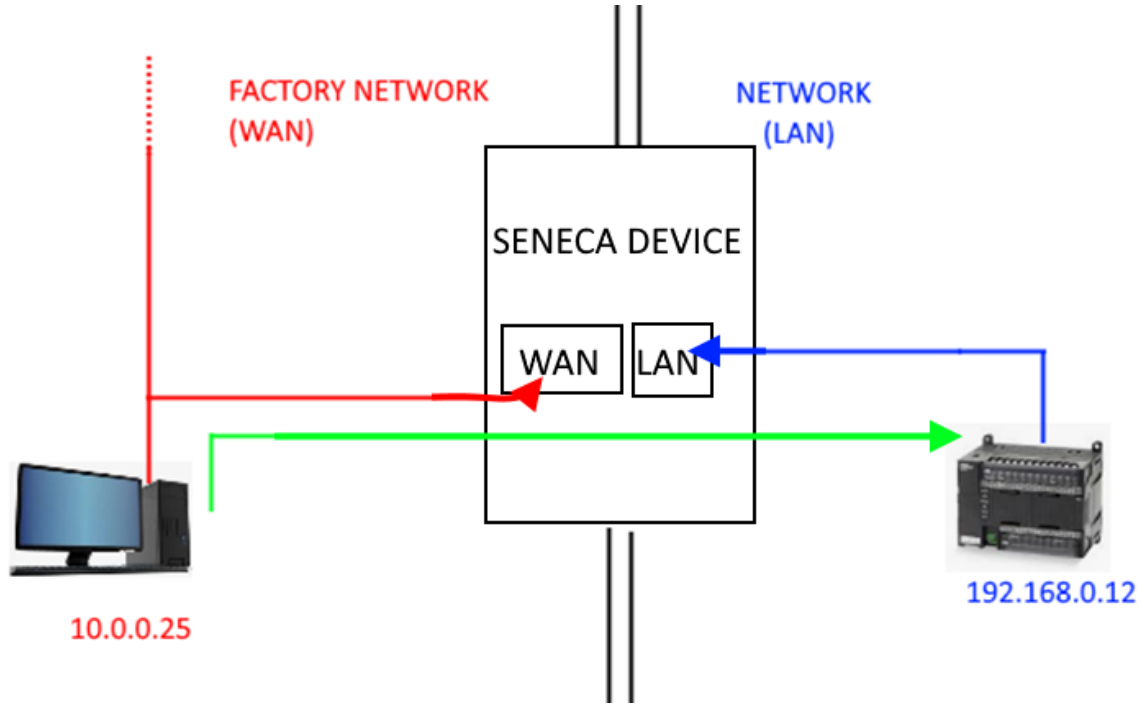
For example, if you set the values:

Protocol = TCP-IP
External Port = 502
Server IP Address = 192.168.85.103
Internal Port = 503

The rule tells the device that any TCP or UDP packet received by the gateway on port 502 (which is often used for the Modbus TCP protocol) should be forwarded to the IP address 192.168. 85.103 (which corresponds to another device) on the same destination port 503.

8.17. NAT 1:1 RULES PAGE

You can use this page to access a device (for example a PC) from the WAN to the LAN.
You then want to access a PC connected in the WAN to a PLC connected in the LAN network as shown in the figure:



It is necessary to create a new address (10.0.0.26) which is located on a PC-compatible network (10.0.0.25).

	CURRENT	UPDATED
NAT 1:1 Configuration		
Interface		WAN
Device IP Address		192.168.0.12
Mapped IP Address		10.0.0.26
Description		WAN to LAN ACCESS1

APPLY

PLC 192.168.0.12 is now accessible from the WAN using address 10.0.0.26.

Interface
Allows you to choose the interface among those available

Device IP Address
It is the address of the device that must be reached

Mapped IP Address
It is the new virtual IP address that must be compatible with the selected network (interface)
Description
It is the mnemonic description of the rule

8.18. STATIC ROUTES PAGE

This page allows you to set static routes, this function allows you to route an address or a range of addresses to different gateways.
For example, if you need to reach 2 different addresses: 192.168.85.23 and 192.168.82.56 but you need to go through 2 different gateways.

Destination Address
It is the destination address to reach

Subnet Mask
It is the subnet mask

Gateway
It is the address of the gateway that it must pass through

Interface
This is the interface used, you can choose between LAN, WAN, Mobile or VPN Layer3 or Wi-Fi (where available).

Description
It is the mnemonic text of the rule

For example, you have:

1) To access 192.168.85.23 it is necessary to pass through gateway 192.168.80.1
2) To access 192.168.82.56 it is necessary to pass through gateway 192.168.80.100
You will have to use the configuration:

Rule #1:
Destination Address = 192.168.85.23

Subnet Mask = 255.255.255.255

Gateway = 192.168.80.1

Interface = LAN

Description = Go to 85

Rule #2:
Destination Address = 192.168.82.56

Subnet Mask = 255.255.255.255

Gateway = 192.168.80.100

Interface = LAN

Description = Go to 82

8.19. MOBILE NETWORK PAGE (Mobile Configuration)

This page allows you to configure your mobile connection (if present).

8.19.1.SIM SECTION

PIN
This is the PIN number to access the SIM (if configured)

8.19.2.OPERATOR SELECTOR SECTION

Mode
You can choose the strategy to select the mobile operator:

Automatic: the operator is chosen automatically
Manual: the operator is set manually, if the operator is not available, the connection cannot take place
Manual/Automatic: allows you to set the operator in manual mode but if the operator is not available the system will switch to “automatic” mode.

Operator
Allows you to select the operator manually, to display a list of available operators in the area you need to press the “Get Operator List” button

8.19.3.DATA CONNECTION SECTION

Enable
Enables or disables the use of mobile data.

APN Mode
It allows you to manually set the APN or use the auto APN (the APN is retrieved from an internal database). Attention, the database does not contain all the possible world APNs but only the main ones.

APN
This is the APN (access point that allows mobile devices to use an Internet connection) currently used or to be used.

Authentication Type
This is the type of authentication to use for the APN

Username
This is the username for the APN

Password
This is the password for the APN

Host for connection check (ping)
This is the URL or IP that the device uses to diagnose the mobile connection.

Set Default Gateway
Allows you not to set a default gateway for the mobile network (and therefore to keep the default gateway of the WAN or WIFI network).

8.20. DDNS CONFIGURATION PAGE (Mobile Configuration)

This page allows you to configure DDNS services. Dynamic DNS (DDNS) is a technology that allows a DNS name on the Internet to always be associated with the IP address of the same host, even if the address changes over time.

TYPE
Allows you to choose the DDNS service to use from those listed.

Hostname
This is the DDNS hostname

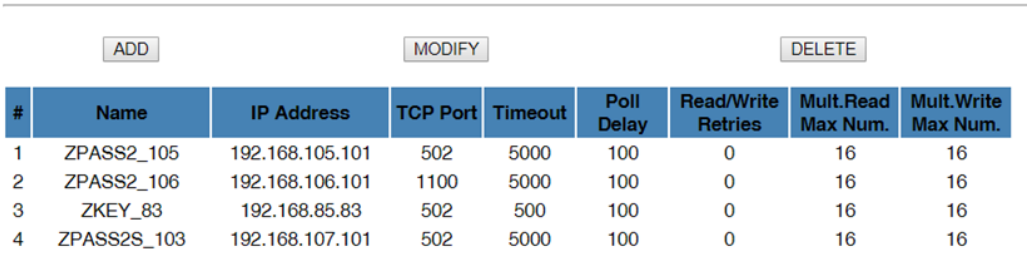
Username
This is the username for the service

Password
This is the password for the service

8.21. TCP SERVERS PAGE (Shared Memory Tag Conf.)

This page shows the list of remote Modbus TCP servers, used to acquire data in the Modbus Shared Memory Gateway function.

By clicking on the "ADD" button you can configure a new TCP server, as in the figure below:



#	Name	IP Address	TCP Port	Timeout	Poll Delay	Read/Write Retries	Mult.Read Max Num.	Mult.Write Max Num.
1	ZPASS2_105	192.168.105.101	502	5000	100	0	16	16
2	ZPASS2_106	192.168.106.101	1100	5000	100	0	16	16
3	ZKEY_83	192.168.85.83	502	500	100	0	16	16
4	ZPASS2S_103	192.168.107.101	502	5000	100	0	16	16

Name
TCP Server Mnemonic Name, this name is used to identify the TCP server in the “Tag Setup” and “Tag View” pages.

IP Address
IP address of the remote Modbus TCP-IP server

TCP Port
Server TCP port

Timeout (ms)
Connection timeout / response for Modbus TCP requests, in milliseconds

Delay between Polls (ms)
Interval between two consecutive Modbus TCP requests, in milliseconds

Read/Write Retries
Maximum number of attempts for Modbus TCP requests; this always applies to write requests; for read requests, only applies to tags with "Gateway Tag Mode" = "BRIDGE".

Multiple Read Max Number
Maximum number of Modbus registers that can be read in a single Modbus TCP request; it is used to reduce the number of read requests sent via the TCP connection, thus optimising performance

Multiple Write Max Number
Maximum number of Modbus registers that can be written in a single Modbus TCP request; it is used to reduce the number of write requests sent via the TCP connection, thus optimising performance.

The maximum number of TCP-IP Modbus Servers that can be configured is 25.

8.22. TAG SETUP PAGE (Shared Memory Tag Conf.)

This page is used to configure tags in Modbus Shared Memory Gateway mode.
It is possible to import the inserted tags through an Excel template (downloadable from the Seneca website) or export the current ones.
It is also possible to insert new tags directly from the web page, all Seneca devices are available via an internal database.

The addition of a tag has the following fields (most of them pre-compiled as defined in the database included in the product)

Gateway Tag Name
Tag mnemonic name

Gateway Modbus Start Register Address
Start address of the tag on the Shared Memory Gateway

Target Device
Allows you to choose the Target device type between Custom or from Seneca database.

Target Connected To
The serial port or Ethernet resource to which the external device is connected.

Target Modbus Request Type
Indicates the type of Modbus command to use for the query between:
Coil, Discrete Input, Holding Register and Input Register

Target Register Data Type
Indicates the data type of the register between signed/unsigned up to 64bit, Real, Bool and String (max 255 bytes)

Target Data Size

Indicates the size in bytes of the chosen data type (editable only for the string data type)

Target Modbus Station Address

Device from which to read (or write to) the tag (if present in the database) or custom.

Target Resource

Represents the device resource to which the TAG is associated (e.g. Input1, Output2 etc...) only in the case other than Custom Device not present in the database.

Gateway Tag Mode

This field defines how the tag will be handled by the gateway processes; possible values are:
GATEWAY, BRIDGE, SHARED MEMORY or EMBEDDED.

The difference between Gateway and Bridge is that Bridge tags are updated only when required, in Gateway mode the tags are updated cyclically even if they are not required.

SHARED MEMORY are tags that can be written by Modbus RTU / Modbus TCP-IP or by Logical Rules and are TAGs representing local variables. This type of tag can also be used for calculated tags.

EMBEDDED
for integrated digital I/Os on board the device

Gain

This field corresponds to the value of the coefficient m in the formula
m * val + q
applied to the value "val" read by the device

Offset

This field corresponds to the value of the coefficient q in the formula
m * val + q
applied to the value "val" read by the device

Initial Value

Starting value of the tag (only for the Shared Memory case)

Error Mode

This field defines which value is provided in the answer to a Modbus (read) request, when the value from the destination device is not available.
The possible ways are:

LAST VALUE: the last available value is given.

ERROR VALUE: the value specified in the field " ERROR VALUE " is provided.

Error Value

This field defines which value is given in the reply to a Modbus request (reading), when the value from the destination device is not available and the " ERROR MODE " field is set to " ERROR VALUE".

HTTP POST VID

This field is used to create the "Variable ID" (VID) that identifies the tag in HTTP POST requests (useful only when HTTP POST protocol is enabled).
The VID string is given by the "V" character plus the number contained in the field

Read Only

If selected, the tag can only be written by an external protocol (e.g. Modbus RTU or TCP-IP) and not by a logical rule.

Retain

If selected, the tag is saved in a writable retention memory (feRAM), when you restart the device the last value is loaded from the memory.
This option is only available for SHARED MEMORY tags.

Calculated Function

Only active if Tag mode is "Shared Memory". Can be used to calculate the MIN / MAX / AVG value of a tag. Note that the calculation is only enabled if the datalogger is enabled. The averaging calculation time is given by the acquisition time.

Export to Display/PLC

If active, it allows the tag to be displayed on the display or virtual display (depending on whether the device is equipped with a display or not) and on the Straton PLC.

Alarm Enabled

This field is a read-only flag that indicates whether an alarm has been defined for the tag.

8.23. TAG VIEW PAGE (Shared Memory Tag Conf.)

This page displays the real time values of the configured tags.

The "Data Logger" buttons can be used for:
- start the Data Logger functionality, if it has been stopped (START);
- interrupt the Data Logger functionality, if running (STOP);
- clean the Data Logger's internal cache (this will also stop the Data Logger) (CLEAN CACHE).

The display is automatically updated.

The "ALARM" column shows the status of the alarm defined for the tag, if present; the ANALOG DANGER ALARM" column has a similar behaviour, but is only meaningful for analog tags when the "Alarm Low Low Value" and "Alarm High High Value" thresholds are defined in the alarm configuration.
It is also possible to export the datalogger files to a USB stick by pressing the "COPY TO USB" button.
If the TAG is writable the last column also includes a button that can be used to write a value to the selected tag.

8.24. DB DEVICE CUSTOM PAGE (Shared Memory Tag Conf.)

On this page you can manage the database of registers of external devices to connect to.

8.25. ALARM CONFIGURATION PAGE (Alarms)

This page displays the list of configured alarms.
By clicking on the "ADD" button, you can configure a new alarm.

Enabled
Flag to enable / disable an alarm

Type
This parameter indicates whether it is a digital or analog alarm; when changing the type, some parameters are enabled or disabled

Name
The name of the alarm; since this parameter is used as a key to identify the alarm, it is not possible to configure two alarms with the same name

Tag
The tag to which the alarm is connected.
The list of tags changes according to the type of alarm (digital or analog).
You can only associate one alarm to one tag

Activation Delays
This parameter defines the time interval, in seconds, during which the alarm condition must be kept true to generate the alarm

Ignore on Boot
This is a flag used to avoid generating the alarm, if the alarm condition is detected during system startup

Auto Acknowledge
This is a flag used to avoid the need for an acknowledgement (ACK) by the user to allow the alarm to be cleared when it ceases.

Boolean Alarm Value
For a digital alarm, this parameter indicates the value of the tag (LOW or HIGH) that corresponds to the alarm condition.

Alarm Low Value
For an analog alarm, this parameter defines the low alarm threshold i.e. if the tag value falls below this threshold, the alarm condition is activated

Alarm High Value

For an analog alarm, this parameter defines the high alarm threshold i.e. if the tag value exceeds this threshold, the alarm condition is activated

Alarm Low Low Value

For an analog alarm, this parameter defines the low dangerous alarm threshold, i.e. if the tag value falls below this threshold, the alarm condition is activated

Alarm High High Value

For an analogical alarm, this parameter defines the high dangerous alarm threshold, i.e. if the tag value exceeds this threshold, the alarm condition is activated.

Deadband Value

This parameter defines a range within which the alarm does not fall (hysteresis).

The possible alarm states are explained in the following table:

Status	Level	Meaning
None	-	The tag has never entered the alarm condition
Alarm	Alarm	The value of the digital has reached the value defined by the parameter "Boolean Alarm Level".
Alarm Low	Alarm	The analog tag has fallen below the value defined by the "Alarm Low Value" parameter
Alarm High	Alarm	The analog tag has exceeded the value defined by the "Alarm High Value" parameter
Alarm Low Low	Analog Danger Alarm	The analog tag has fallen below the value defined by the "Alarm Low Value" parameter
Alarm High High	Analog Danger Alarm	The analog tag has exceeded the value defined by the "Alarm High Value" parameter
Acknowledge	-	The alarm received ACK from the user (or was configured with Auto Acknowledge)
Return	-	The tag has exited the alarm condition, but the alarm has not been acknowledged and the alarm has the "Auto Acknowledge" parameter set to OFF
End	-	The tag has exited the alarm condition and the alarm has been acknowledged or the alarm has the "Auto Acknowledge" parameter set to ON

As already mentioned, when exiting the alarm condition the alarm states can follow two different paths, depending on the value of the " Auto Acknowledge" parameter:

- Alarm* → Return → <ACK> → End if “Auto Acknowledge”=OFF
- Alarm* → End if “Auto Acknowledge”=ON

8.26. ALARM SUMMARY PAGE (Alarms)

This page shows the alarms currently active in the system.

Name
Alarm name

Tag Name
Tag connected to the alarm

Level
“Hazard” level of the alarm:
Alarm" value for digital alarms
Alarm" or "Analog Danger Alarm" may apply for analog alarms

Status On
Alarm status when triggered

Timestamp On
Date Time of when the alarm was triggered

Status Action
"None" when the alarm goes off
It can evolve into:
"Acknowledged", If the alarm has been acknowledged
"Return", if the alarm has returned but the "Auto Acknowledge" setting is OFF

Timestamp Action
Date Time of action (previous field)

8.27. ALARM HISTORY PAGE (Alarms)

This page shows all alarm status transitions that have occurred in the system, up to a maximum of 1000; alarm status transitions are shown from the most recent to the oldest.

8.28. SD/USB TRANSFER CONFIGURATION PAGE (CLIENT PROTOCOLS)

This page contains parameters that indicate whether log files are copied to a USB stick (in models without a micro SD card slot) or to a micro SD card and for how long they are kept.

Enable
Enable or disable copying of logs to USB

Max Failure Counter

This parameter defines the maximum number of failed copy attempts before entering the "Wait after failure" state (see next field)

Wait After Failure (minutes)
This parameter defines the duration, in minutes, of the "Wait after failure" status.
In this state, no further attempt is made to copy a log file to the USB

Clean Period (days)
This parameter defines for how many days the log files must be kept on the USB; that is, after the specified number of days, the log files are deleted.

Files are saved in folders according to the following convention:

yyyymmdd (yyyy=year, mm=month, dd=day)

example:

20180612

Each of these folders includes a subfolder:

logX X=[1..4], number of the group

The log file name has the following convention:

Lmmmmmmm.csv

where *mmmmmmm* is the number of minutes from [1/1/2000 00:00], corresponds to the date of the first log line example:

L9701690.csv

SD cards and USB sticks must be formatted with the FAT32 filesystem.

ATTENTION!

USB STICKS OR SD CARDS ARE OFTEN FORMATTED WITH THE "EXFAT" FILESYSTEM (DEPENDING ON THE SIZE) AND MUST THEREFORE BE REFORMATTED WITH THE "FAT32" FILESYSTEM

8.29. FTP CONFIGURATION PAGE (CLIENT PROTOCOLS)

This page contains parameters related to the transfer of log files to a remote FTP server.

Enable
Enable or not the transfer of logs via FTP

Max Failure Counter
This parameter defines the maximum number of failed copy attempts before entering the "Wait after failure" state (see next field)

Wait After Failure
This parameter defines the duration, in minutes, of the "Wait after failure" status.
In this state, no further attempt is made to copy a log file to the USB

Crypto Mode
Defines which encryption to use for the FTP connection between:
- None
- TLS/SSL Implicit
- TLS/SSL Explicit

Host
Hostname (FQDN) or FTP server IP address

Port
TCP port of the FTP server

Username
Server Username

Password
Server password

Path
Directory path, on the FTP server, where the log files will be saved. It must start with the character "/".

Log files transferred via FTP will have the following format:
<*RTU_Name*>_X_log<*date_time*>.csv
Where:
- <*RTU_Name*> is the value of the "RTU Name" field in the "General Settings" page
- *X=[1..4]* is the number of the group

- <*date_time*> has the format *yyyymmdd* (yyyy=year, mm=month, dd=day); corresponds to the log first line date

Example:

SENECA_1_log20180507101507.csv

8.30. EMAIL CONFIGURATION PAGE (CLIENT PROTOCOLS)

Emails can be used to transfer log files or to send alarms; some parameters on this page are only used when transferring log files, not when sending alarms;
these parameters are marked with the caption "Data Logger Only".

Enable
Flag indicating whether log files are transferred via EMAIL or not
Note that it is possible to send alarms via EMAIL even if this parameter is set to OFF.

Max Failure Counter
This parameter defines the maximum number of failures before entering the "Wait after failure" state (see next field).

Wait After Failure (minutes)
This parameter defines the duration, in minutes, of the "Wait after failure" status.
In this state, no further attempt is made to send a log file or alarm via EMAIL

Crypto Mode
This parameter defines the encryption type of the EMAIL connection.
The possible ways are:
None
TLS/SSL
STARTTLS

Host
Hostname (FQDN) or IP address of the MAIL server

Port
EMAIL server port (TCP)

Username
EMAIL server username

Password
EMAIL server password

From
Sender's email address

To
List of one or more e-mail recipient addresses, separated by commas.
This parameter is only used for the transfer of log files
Subject
Subject of the email.
This parameter is only used for the transfer of log files

Text
Email text: If left blank a standard text is added.
This parameter is only used for the transfer of log files

Line Terminator
Type of line terminator to use

Log files sent as EMAIL attachments have names with the following format:
<RTU_Name> _X_log <date_time> .csv
where:
- <RTU_Name> is the value of the “RTU Name" parameter in the "General Settings" page
- X = [1..4] is the number of the group
- <date_time> has the format yyyymmdd (yyyy = year, mm = month, dd = day); this is the timestamp of the first sample (line) in the log file

for example:
SENECA_1_log20180507101507.csv

Emails containing alerts have the following text format:
MESSAGE: <timestamp>
<nome rtu> <testo messaggio>

with the following object:
<nome rtu>: ALARM

Sending alarm messages is managed by the “Rule Management” section.

8.31. HTTP CONFIGURATION (CLIENT PROTOCOLS)

The http post protocol can be used to send log samples or alarms (events) to an HTTP server.

Enable
Enable or not the sending of logs via http

Max Failure Counter
This parameter defines the maximum number of failures before entering the "Wait after failure" state (see next field).

Wait After Failure (minutes)
This parameter defines the duration, in minutes, of the "Wait after failure" status.
In this state, no further attempt is made to send a log file or alarm via http POST.

SSL/TLS
This parameter defines whether or not to enable HTTP connection encryption.

Host
Hostname (FQDN) or HTTP server IP address

Port
TCP port of the HTTP server

Seneca Protocol
If enabled, it allows HTTP sending with the typical parameters of the Seneca protocol (used on Cloud Box)

Authentication
Allows you to enable or disable user/password authentication

Username
HTTP server username

Password
HTTP server password

Path
Adds a PATH string

Url
Allows you to view the publication string

You can also refer to the specific document of the http protocol used

8.32. MQTT CONFIGURATION (CLIENT PROTOCOLS)

The MQTT protocol can be used to send (and receive) data or events to a cloud server (called a broker).

Enable
Enable or not the MQTT protocol.

Max Failure Counter
This parameter defines the maximum number of failures before entering the "Wait after failure" state (see next field).

Wait After Failure (minutes)
This parameter defines the duration, in minutes, of the "Wait after failure" status.
In this state, no further attempts are made to send or receive data via MQTT.

Client ID
Defines the Client ID used in the MQTT protocol

Broker Host
Defines the host name of the MQTT broker

Broker Port
Defines the MQTT broker port

Use WebSockets
Allows you to activate MQTT communication via Websockets

Keep Alive Interval (seconds)
This parameter defines Keep alive which ensures that the connection between the broker and client is still open and that the broker and client are aware that they are connected. When the client establishes a connection to the broker, it tells the broker a time interval in seconds. This interval defines the maximum period of time during which the broker and client may not communicate with each other.

Clean Session
This parameter defines the "clean session".
When the clean session flag is set to true, the client does not want a persistent session. If the client disconnects for any reason, all information and messages queued from a previous session are lost.

Message Retain
Usually if a publisher publishes a message on a topic to which no one is subscribed, the message is simply discarded by the broker. However, the publisher can tell the broker to keep the last message of that topic.

Quality of service
This parameter defines the QOS of the MQTT protocol.
Can be selected from
QOS 0 (once only, without ack)
QOS 1 (at least once, with ack)
QOS 2 (once only, with ack and resend)

Authentication

This parameter defines whether user/password authentication should be used to access the broker

Username
Broker Username

Password
Broker password

SSL/TLS
Defines if the crypto is SSL/TLS

Log on Change
This parameter defines whether topics should only be sent in case of change (based on minimum time) or not.

Publish with multiple tags
This parameter defines whether the publish contains multiple tags or whether the device should send a publish for each tag

Publish Topic for Logs
Selects the topic name for the logs using the following table:

%c	Device Client ID
%m	Device MAC Address
%M	Device MAC Address (without ':')
%e	Device IMEI
%d	Date/Time
%t	timestamp (number of seconds from 01/01/1970)
%x	text (only for "Publish Payload for Alarms")
%b	bulk (format specified in "Publish Bulk Format")
%n	Tag name (only for "Publish Bulk Format")
%v	Tag value (only in "Publish Bulk Format")
%i	Tag validity flag (only in "Publish Bulk Format")
%f	Tag id with progressive number (only in "Publish Bulk Format")
%j[field]	Adds double quotes " to [field]. The double quotes represent a string in JSON
%tag_name	Value of the "tag_name" tag
%#tag_name#	Validity of the "tag_name" tag
%u	Timestamp in [ms] (only in "Publish Fast Log Sample" and "Publish Bulk Format")
%p	Sampling period (only in "Publish Fast Log Sample")
%w	Format (only in "Publish Fast Log")

Publish Payload for Logs

Selects the format to be used for the payload in Json format using the following table:

%c	Device Client ID
%m	Device MAC Address
%M	Device MAC Address (without ':')
%e	Device IMEI
%d	Date/Time
%t	timestamp (number of seconds from 01/01/1970)
%x	text (only for "Publish Payload for Alarms")
%b	bulk (format specified in "Publish Bulk Format")
%n	Tag name (only for "Publish Bulk Format")
%v	Tag value (only in "Publish Bulk Format")
%i	Tag validity flag (only in "Publish Bulk Format")
%f	Tag id with progressive number (only in "Publish Bulk Format")
%j[field]	Adds double quotes " to [field]. The double quotes represent a string in JSON
%tag_name	Value of the "tag_name" tag
%#tag_name#	Validity of the "tag_name" tag
%u	Timestamp in [ms] (only in "Publish Fast Log Sample" and "Publish Bulk Format")
%p	Sampling period (only in "Publish Fast Log Sample")
%w	Format (only in "Publish Fast Log")

Publish Bulk Format

Selects the format for "bulk mode" according to the following table:

%c	Device Client ID
%m	Device MAC Address
%M	Device MAC Address (without ':')
%e	Device IMEI
%d	Date/Time
%t	timestamp (number of seconds from 01/01/1970)
%x	text (only for "Publish Payload for Alarms")
%b	bulk (format specified in "Publish Bulk Format")

%n	Tag name (only for "Publish Bulk Format")
%v	Tag value (only in "Publish Bulk Format")
%i	Tag validity flag (only in "Publish Bulk Format")
%f	Tag id with progressive number (only in "Publish Bulk Format")
%j[field]	Adds double quotes " to [field]. The double quotes represent a string in JSON
%tag_name	Value of the "tag_name" tag
%#tag_name#	Validity of the "tag_name" tag
%u	Timestamp in [ms] (only in "Publish Fast Log Sample" and "Publish Bulk Format")
%p	Sampling period (only in "Publish Fast Log Sample")
%w	Format (only in "Publish Fast Log")

Publish Bulk Format for Fast Logging

Selects the format for the "bulk mode" for the fast logging data according to the following table:

%c	Device Client ID
%m	Device MAC Address
%M	Device MAC Address (without ':')
%e	Device IMEI
%d	Date/Time
%t	timestamp (number of seconds from 01/01/1970)
%x	text (only for "Publish Payload for Alarms")
%b	bulk (format specified in "Publish Bulk Format")
%n	Tag name (only for "Publish Bulk Format")
%v	Tag value (only in "Publish Bulk Format")
%i	Tag validity flag (only in "Publish Bulk Format")
%f	Tag id with progressive number (only in "Publish Bulk Format")
%j[field]	Adds double quotes " to [field]. The double quotes represent a string in JSON
%tag_name	Value of the "tag_name" tag
%#tag_name#	Validity of the "tag_name" tag
%u	Timestamp in [ms] (only in "Publish Fast Log Sample" and "Publish Bulk Format")
%p	Sampling period (only in "Publish Fast Log Sample")
%w	Format (only in "Publish Fast Log")

Publish Topic for Alarms

Selects the format for topic names in alarms according to the following table:

%c	Device Client ID
%m	Device MAC Address
%M	Device MAC Address (without ':')
%e	Device IMEI
%d	Date/Time
%t	timestamp (number of seconds from 01/01/1970)
%x	text (only for "Publish Payload for Alarms")
%b	bulk (format specified in "Publish Bulk Format")
%n	Tag name (only for "Publish Bulk Format")
%v	Tag value (only in "Publish Bulk Format")
%i	Tag validity flag (only in "Publish Bulk Format")
%f	Tag id with progressive number (only in "Publish Bulk Format")
%j[field]	Adds double quotes " to [field]. The double quotes represent a string in JSON
%tag_name	Value of the "tag_name" tag
%#tag_name#	Validity of the "tag_name" tag
%u	Timestamp in [ms] (only in "Publish Fast Log Sample" and "Publish Bulk Format")
%p	Sampling period (only in "Publish Fast Log Sample")
%w	Format (only in "Publish Fast Log")

Subscribe Topic

Selects the Topic Subscribe according to the following table:

%c	Device Client ID
%m	Device MAC Address
%M	Device MAC Address (without ':')
%e	Device IMEI
%d	Date/Time
%t	timestamp (number of seconds from 01/01/1970)
%x	text (only for "Publish Payload for Alarms")

%b	bulk (format specified in "Publish Bulk Format")
%n	Tag name (only for "Publish Bulk Format")
%v	Tag value (only in "Publish Bulk Format")
%i	Tag validity flag (only in "Publish Bulk Format")
%f	Tag id with progressive number (only in "Publish Bulk Format")
%j[field]	Adds double quotes " to [field]. The double quotes represent a string in JSON
%tag_name	Value of the "tag_name" tag
%#tag_name#	Validity of the "tag_name" tag
%u	Timestamp in [ms] (only in "Publish Fast Log Sample" and "Publish Bulk Format")
%p	Sampling period (only in "Publish Fast Log Sample")
%w	Format (only in "Publish Fast Log")

LWT Topic

Selects the "Last Weel and Testament" topic according to the following table:

%c	Device Client ID
%m	Device MAC Address
%M	Device MAC Address (without ':')
%e	Device IMEI
%d	Date/Time
%t	timestamp (number of seconds from 01/01/1970)
%x	text (only for "Publish Payload for Alarms")
%b	bulk (format specified in "Publish Bulk Format")
%n	Tag name (only for "Publish Bulk Format")
%v	Tag value (only in "Publish Bulk Format")
%i	Tag validity flag (only in "Publish Bulk Format")
%f	Tag id with progressive number (only in "Publish Bulk Format")
%j[field]	Adds double quotes " to [field]. The double quotes represent a string in JSON
%tag_name	Value of the "tag_name" tag
%#tag_name#	Validity of the "tag_name" tag
%u	Timestamp in [ms] (only in "Publish Fast Log Sample" and "Publish Bulk Format")
%p	Sampling period (only in "Publish Fast Log Sample")

%w	Format (only in "Publish Fast Log")

LWT Payload
Selects the Payload text of "Last Weel and Testament"

Save Configuration URL
This is the URL for the "Save Configuration" command received from mqtt (see the chapter on sending commands from the cloud in this manual)

Load Configuration URL
This is the URL for the "Load Configuration" command received from mqtt (see the chapter on sending commands from the cloud in this manual)

FW Update URL
This is the URL for the "FW Update" command received from mqtt (see the chapter on sending commands from the cloud in this manual)

Sleep Timeout
MQTT task wake-up time, the shorter it is, the more responsive MQTT is (at the expense of higher CPU load).

MQTT Certificates
It is used to manage the certificates necessary for the TLS connection.

8.32.1.EXAMPLE

Suppose you want to send two TAGs lods: tag1 and tag2, with the following configuration:

Client ID = "Test"
Publish Topic for Logs = seneca/%c/data
Publish Payload for Logs = {"type": "data", "message": {"device": %jc, "date": %t, "signals": [%b]}}
Publish Bulk Format = {"name": %jn, "value": %v, "valid" : %i}

You will get: on "Seneca/Test/data" topic the following Payload:
{"type": "data", "message": {"device": "Test", "date": 1750942723, "signals": [{"name": "tag1", "value": 1234, "valid" : 1}, {"name": "tag2", "value": 5678, "valid" : 1}]}}

8.33. PHONEBOOK PAGE (LOGIC CONFIGURATION)

This page is used to configure the address book for sending text messages by the device via email and/or (on models equipped with a modem) SMS or audio calls.

User Type
It is possible to define three different account profiles:

Admin
This account receives alarms via SMS or EMAIL or AUDIO from any group.
This account can send SMS commands to the device, It also receives all rejected or unrecognised SMS commands (if the "SMS Relay to Admin" parameter is set to ON and all "Startup SMS" messages if the "Startup SMS" parameter is set to ON).
Manager
This account receives alarms via SMS or EMAIL or AUDIO from the group to which it belongs.
This account can send SMS commands to the device.

User
This account receives alarms via SMS or EMAIL or AUDIO from the group to which it belongs.

At the time of compilation, the group(s) to which the account belongs is required, so you can divide the alerts between the various accounts.
Note how "Admin" accounts receive alarms from any group.

8.34. MESSAGE CONFIGURATION PAGE (LOGIC CONFIGURATION)

In this section it is possible to define the text messages related to the alarms that the device must manage.
The message text can only contain ASCII characters.
It is possible to use the {TAG_NAME} syntax to include the current value of a tag in the text.

For example the message text:

“WATER LEVEL ={LEVEL} m”

Will provide a text with the tag value as text, if the tag “LEVEL” is 1,232 you will have:

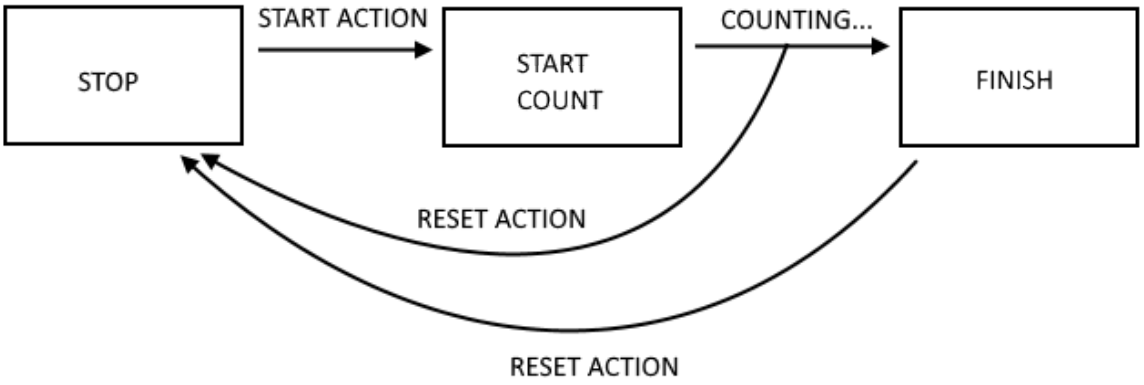
WATER LEVEL = 1.232 m

This syntax can be used more than once in a message text.
Each message has an ID field which is used to associate the message with the alarm in the logical rules.

8.35. TIMER CONFIGURATION PAGE (LOGIC CONFIGURATION)

This section allows you to define up to 100 timers to be used in logic rules.
The ID represents the mnemonic of the timer that must be used in the rules.
"Enable" selects whether the timer is active or not.
"Duration" is the activation value in [ms].

Note
The timers are in stop mode by default, they need an action to start and an action to restore, according to the following scheme:



8.36. RULE SCRIPTS PAGE (LOGIC CONFIGURATION)

On this page you can upload files related to scripts to be executed as actions of logical rules.
The extension for the type of script to be used must be respected:

Script type	***Extension***
Linux Shell	".sh"
PHP	".php"
Python	".py"
Binary	".bin"

You can upload a maximum of 100 Kbyte file.

8.37. AUDIO FILES PAGE (LOGIC CONFIGURATION)

On this page you can upload audio files to your device that will be played in the event of an audio call. Audio files must have the following characteristics:
- ".wav" or ".WAV" extension
- PCM 8 KHz, 16 bit coding per sample
- 1Mbyte maximum size

For each call the file is played 5 times and is interrupted in case of confirmation with the DTMF code (if enabled).

8.38. RULE MANAGEMENT PAGE (LOGIC CONFIGURATION)

In this section you can define a set of logical rules that will implement a program.

The first section contains some general parameters:

Writing Mode
You can choose between “During execution” and “After execution”, these parameters allow you to select when the Tag writing of the Analog/Digital Tag write action occurs. During execution will execute the tag writing immediately after executing the write action, After execution will follow the tag writing at the end of the entire execution of the logical rules list.

Maximum Number of Call Loops
This parameter specifies the maximum number of call loops to group numbers.
Example: If Antonio, Beppe and Giulio are present in the call group and the loop number is 3, if no one confirms the call, each one will be called 3 times. After these, even if no confirmation has been received, the event will be confirmed.

DTMF Acknowledge Enable (#99*)
You can choose between “ON” and “OFF”, in the case of “ON” for the call to be confirmed it is necessary to enter the DTMF tone sequence #99* on the telephone keypad.
In the case of “OFF” for confirmation it is necessary that there has been an answer to the call (even on the answering machine) and that the audio is played at least once.

To configure a rule, the following parameters are available:

8.38.1.RULE CONFIGURATION

Enabled
Indicates whether the rule is enabled or should be excluded from execution

Index
Rule execution order (1 = First rule to be executed)

Description
Mnemonic textual description of the rule

Period [ms]
If the value is = 0, actions are executed only if there is a change in the result of the "OR / AND" (i.e. on change of state).

If the value is different from 0 ms the actions are performed trying to respect the inserted timing.

ATTENTION!

Use appropriate period values for EMAIL / SMS / http / MQTT / AUDIO sending actions!

NOTE:
If Period is > 0 the actions are always performed in "repeat" mode

8.38.2.IF CONDITION: TYPE

This section defines the type of condition, the following types are possible:

None
No conditions to be assessed

Alarm State
The condition refers to the state of an alarm, the following parameters are possible:

Field	***Meaning***
Alarm Name	Selects the alarm from the list of all configured alarms
Alarm State	Alarm status. Possible states are: None Alarm (digital only) Alarm Low Low (analog only) Alarm Low (analog only) Alarm High (analog only) Alarm High High (analog only) Acknowledge Return End Depending on the type (digital or analog) of the selected alarm, some states are disabled
Analog Danger Alarm	Flag indicating whether the alarm level must be "Analog Danger" or not, applies only to alarms on analog tags

Alarm Active
The alarm condition refers to the Active or No state of an alarm, the following parameters are possible:

Field	***Meaning***
Alarm Name	Selects the alarm from the list of all configured alarms
Alarm Active	Indicates whether or not the alarm should be active. The alarm is active if it is in one of these states: Alarm (only for digital tags) Alarm Low Low (only for analog tags) Alarm Low (only for analog tags) Alarm High (only for analog tags) Alarm High High (only for analog tags) Acknowledge

	The alarm is not active if it is in one of the following states: None Return End
Analog Danger Alarm	Flag indicating whether the alarm level should be "Analog Danger" or not, significant only for analog alarms.

Always

The If condition is always true.

Note that the rule is only executed once if Period is = 0 ms or if the actions are in one time mode.

If you need to execute a rule at each cycle, you need to put the actions in "repeat mode".

If you need to run a rule over time (every x ms), you must set Period > 0ms.

Digital Tag

The condition depends on the state of a digital tag:

Field	***Meaning***
Tag	Selects the tag to be used for the condition
Operator	Only "=" may apply
Tag / Constant value	Selects whether the comparison is between another digital tag or a constant boolean value (TRUE or FALSE)

Analog Tag

The condition depends on a comparison with an analog TAG

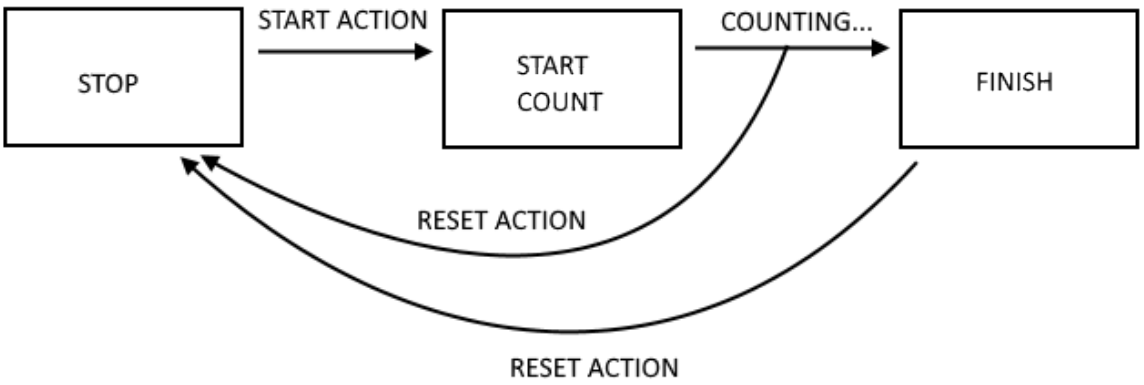
Field	*Meaning*
Tag	Selects the tag to be used for the condition
Operator	It may be: “=” “>” “<” “>=” “<=”
Tag / Constant value	Selects whether the comparison is between another analog tag or a constant value

Timer

The condition depends on the state of the selected timer

Field	*Meaning*
ID	Selects the timer ID to use
Expired	It can be: "OFF" or "ON" With "ON" the condition is only true when the timer expires (FINISH status). With "OFF" the condition is true until the timer is in STOP or COUNTING. When the timer is in FINISH state the condition becomes false.

The operation of the Timer is shown in the following diagram:



Scheduler

The condition depends on the set scheduler (calendar):

Field	*Meaning*
Type	It may be: Every Day, Every week, Every Month, Every Year, Every Hour, Every NMinutes Every Day: the condition is true every day at the configured hour and minute Every Week: the condition is true once a week on the selected day of the week at the selected hour and minute Every Month: the condition is true once a month on the selected day of the month at the selected hour and minute Every Year: the condition is true once a year on the day, month, hour and minute selected Every Hour: the condition is true once an hour at the minute selected Every NMinutes: the condition is true every N minutes selected
Day	If the type is Weekly sets the day of the week: 0 = Sunday 1 = Monday 2 = Tuesday

	3 = Wednesday 4 = Thursday 5 = Friday 6 = Saturday If the type is Monthly: Selects the day of the month from 1 to 31
Hour	Hours
Minute	Minutes

Rule Status
The condition depends on whether a rule is enabled or not:

Field	***Meaning***
ID	Selects the rule ID
Enabled	Selects between "enabled" or "disabled" If "Enabled" the condition is REAL if the selected rule is enabled. If "Disabled" the condition is REAL if the selected rule is disabled.

Bitmask
The condition depends on masking a tag with a hexadecimal constant:

Field	***Meaning***
Tag	Selects the tag to apply the bitmask to from a list containing all tags with data type "16Bit Unsigned"
Mask	The bit mask represented as a string of 4 hexadecimal digits

The "Bit mask" condition is TRUE if the AND operation bit by bit between the Tag and the Data Mask is different from 0; FALSE otherwise.

Example:
Tag=0x1233 (hexadecimal) = 0b 0001 0010 0011 0011 (binary)
Mask=0x8001 (hexadecimal) = 0b 1000 0000 0000 0001 (binary)
It means that the mask analyses bit0 (least significant) and bit 15 (most significant) of the Tag.
The AND bit by bit provides:

0001 0010 0011 0011
1000 0000 0000 0001

0000 0000 0000 0001
So the condition is TRUE.

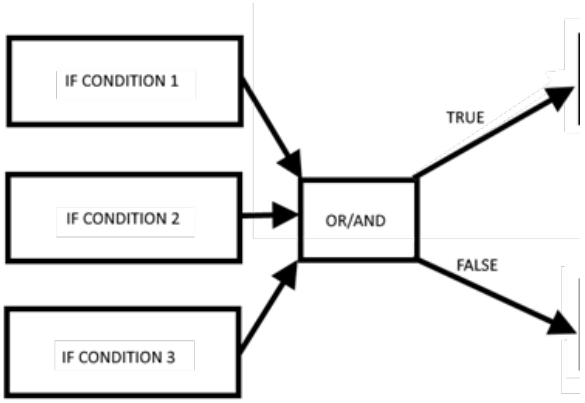
String Tag

The condition depends on a comparison with a string-type TAG

Field	*Meaning*
Tag	Selects the tag to be used for the condition
Operator	It may be: “Equal” “Begins with” “Ends with”
Tag / Constant value	Selects whether the comparison is between another string tag or a constant value

8.38.3.IF CONDITION OPERATOR

The "IF conditions" can be combined together in "OR" or "AND" logic, in practice:



The "IF conditions" linked together by "OR" go to the TRUE state if at least one of the conditions is true.
The "IF conditions" linked together by "AND" only go to the TRUE state if all of them are true.

More details are given in the following table:

IF CONDITION 1	*IF CONDITION 2*	*IF CONDITION 3*	*“OR”*	*“AND”*
FALSE	FALSE	FALSE	FALSE	FALSE
FALSE	FALSE	TRUE	TRUE	FALSE
FALSE	TRUE	FALSE	TRUE	FALSE
FALSE	TRUE	TRUE	TRUE	FALSE
TRUE	FALSE	FALSE	TRUE	FALSE
TRUE	FALSE	TRUE	TRUE	FALSE
TRUE	TRUE	FALSE	TRUE	FALSE
TRUE	TRUE	TRUE	TRUE	TRUE

8.38.4.THEN/ELSE ACTION

In this section you can define the action that must be performed if the conditions result in TRUE (THEN action) or FALSE (ELSE action).

NONE
No action to take

Send Alarm SMS
Send Alarm EMAIL
Send Alarm HTTP POST
Send Alarm MQTT
Send Alarm AUDIO
They allow you to send a text message (defined in the messages section) via client protocols or an audio call (whose file is loaded into the device from the "Audio Files" section or from the default audio files)

Field	*Meaning*
Message	Selects the text message to send from those configured
File	Indicates the audio file to play in the call (only for AUDIO CALL)
Group	Selects the sending group (only for SMS and EMAIL and AUDIO CALL)

For audio calls, there are some sample files pre-loaded on the device.

Digital Tag
Performs a write to a digital tag.

Field	*Meaning*
Action Mode	Allows you to select between "One Time" or "Repeat". With "One Time" the action is executed only if there is a change in the result of the OR / AND conditions. With "Repeat" the action is executed at every loop (if the rule is enabled and if there is no configured period).
Destination Tag	This is the tag where the calculated TRUE/FALSE result is copied
Operator	This is the Boolean operator to use, selected from =, NOT, OR etc ...
Source Tag 1 / Constant value 1	Selects the first tag to use in the boolean calculation. It is also possible to use a boolean constant
Source Tag 2 / Constant value 2	Select the second Tag if the operator needs 2 inputs (For example operator "OR"). It is also possible to use a boolean constant

Analog Tag

Performs a write to an analog type Tag.

Field	***Meaning***
Action Mode	Select from "One Time" or "Repeat". With “One Time” the action is executed only if there is a change in the result of the OR / AND conditions. With "Repeat" actions are performed at each loop (if the rule is enabled and there is no configured period).
Destination Tag	This is the tag where the calculated result is copied to
Operator	It is the mathematical operator to use, you can select from: "=" copies the source tag 1 or the constant value 1 to the destination tag Example: Destination tag = Origin tag 1 Or Target tag = constant value 1 "+ =" Add the value of the source tag1 or the constant value 1 to the target tag and copy the result to the target tag. Example: Destination tag = Destination tag + Origin tag 1 "- =" Subtracts the value of the source tag1 from the target tag and copies the result to the target tag. Example: Destination tag = Destination tag - Origin tag 1 "* =" Multiply the target tag by the value of source tag 1 and copy the result to the target tag. Example: Destination tag = Destination tag * Origin tag 1 "/ ="

	Splits the target tag with the source tag value 1 and copies the result to the target tag. Example: Destination tag = Destination tag / Origin tag 1 "% =" Calculates the rest of the division from the target tag and the value of the source tag1 and copies the result to the target tag. (Note that 53% 7 = 4) Example: Destination tag = Destination tag% Source tag1 "abs" Calculates the absolute value of Source Tag 1 or Constant value 1 and copies the result to the Destination Tag (Note that abs (-4) = 4) Example: Target tag = abs (Source tag 1) "Sqrt" Calculates the square root value of source tag 1 or constant value 1 and copies the result to the target tag. (Note that sqrt (9) = $\sqrt{9}$ = 3) Example: Destination tag = sqrt (origin tag 1) "Sqr" Calculates the square value of the source tag 1 or constant value 1 and copies the result to the target tag. (Note that sqr (3) = 3^2 = 9) Example: Destination tag = sqr (origin tag 1) "Log" Calculates the decimal logarithm of source tag 1 or constant value 1 and copies the result to the target tag. (Note that log (3) = 0.4771212) Example: Destination tag = log (origin tag 1)

	"Ln" Calculates the natural logarithm of the source tag 1 or constant value 1 and copies the result to the target tag. (Note that ln (3) = 1.09861228867) Example: Target tag = ln (Source tag 1) "Exp" Calculate the number of Euler elevated to Source Tag 1 or Constant value 1 and copy the result to the Destination Tag. Please note that: ln (exp 3) = 3 Example: Destination tag = expiration (origin tag 1) "+" Adds Source Tag 1 or Constant value 1 to the value of Source Tag 2 or Constant value 2 and copies the result to the Destination Tag. Example: Target tag = Source tag 1+ Source tag 2 "-" Subtracts the source tag 1 or constant value 1 with the value of source tag 2 or constant value 2 and copies the result to the target tag. Example: Destination tag = Origin tag 1- Origin tag 2 "*" Multiply the source tag 1 or constant value 1 with the source tag 2 or constant value 2 and copy the result to the target tag. Example: Target tag = Source tag 1 * Source tag 2 "/" Splits the source tag 1 or constant value 1 with the source tag 2 or constant value 2 and copies the result to the target tag. Example: Target Tag = Source Tag 1 / Source Tag 2 "%"

	Calculates the rest of the division between source tag 1 or constant value 1 and source tag 2 or constant value 2 and copies the result to the target tag. (Note that 53% 7 = 4) Example: Target tag = Source tag 1% Source tag 2 "Pow" Calculates the Source Tag1 or Constant value 1 elevated to the power of the Source Tag2 / Constant value 2 and copies the result to the destination tag. Example: Target tag = (Source Tag1) ^ (Source Tag2)
Source Tag 1 / Constant value 1	Selects the tag to be used as input 1 for the operator used. You can also use a constant value.
Source Tag 2 / Constant value 2	Selects the Tag to use as input 2 in the calculation if the operator needs 2 inputs. A constant value can also be used

Timer

It is possible to select the action to be performed in the selected timer

Field	***Meaning***
Id	Selects the timer from those configured
Action	Selects the type of action to perform on the selected timer. “Start” performs the start action on the selected timer “Reset” performs the reset action on the timer to the stop state

Rule Status

The action enables or disables a rule.

Field	***Meaning***
Id	Selects the rule
Enable	Selects whether or not the action should enable the selected rule: “OFF” disables the selected rule “ON” enables the selected rule

Datalogger

The action allows you to start or stop the data logger, it is also possible to select the log group to check.

Field	*Meaning*
Group	Selects the data logger group to monitor
Enable	Selects whether or not the action should enable the data logger “OFF” disables the data logger for the selected group “ON” enables the data logger for the selected group

Network

These are actions that allow you to act on the status of the VPN (enable or disable it) or the modem.

Field	*Meaning*
Feature	Allows you to choose which element to perform the ON/OFF action on It is possible to choose between: PPP refers to the mobile modem data connection (if any) VPN refers to the VPN connection Firewall refers to the system firewall OpenVPN refers to the standard OpenVPN connection
Start	You can choose the action to be performed between “ON” and “OFF”.

Set Bits

This action allows you to set a configurable number of bits of a given tag to the value 1 or to the value 0.

Field	*Meaning*
Action Mode	Selects from "One Time" or "Repeat". With “One Time” the action is executed only if there is a change in the result of the OR / AND conditions. With “Repeat” the action is executed at every loop (if the rule is enabled and if there is no configured period).
Destination Tag	It is the tag in which the result of the action is copied, the tag must be of type “16 bit unsigned”
Source Tag	Selects the tag to use in the calculation. It is also possible to insert the same source tag and destination tag in order to perform the action on the same TAG. The tag must be of the “16 bit unsigned” type
Mask	It is the mask in hexadecimal format that allows the masking of the bits to be controlled.
Action	You can choose between “Set” or set the bits to 1, or “Reset” or set the bits to 0.

Data Logger Trigger

Allows the acquisition of a single sample in groups configured as Trigger or Periodic and Trigger. In the case of a group configured with fast logging, it starts the acquisition of max 1000 samples.

Field	*Meaning*
Group	Allows you to select on which log group to execute the action
Source	This is a label that is saved on the data logger in order to discriminate the source of the trigger. the “Source” field can assume the values from “A” to “H”. If the “Data Logger Trigger” action is executed in multiple rules, when different conditions occur, by setting distinct values of “Source” you can discriminate which condition generated the trigger.

Data Logger Send

The action allows the log file to be closed, preparing it for sending via the configured client protocols (valid for protocols that work with files: FTP, EMAIL and SD/USB). It should be used on groups configured with “trigger” sample mode.

Field	*Meaning*
Group	Selects on which group(s) to execute the action

Data Logger Trigger Stop (fast logging)

The action allows you to stop the acquisition set with fast logging before the acquisition stops automatically once 1000 samples are reached.
The start of the fast logging acquisition is given by the data logger trigger action, if it is not stopped by this action the fast logging samples 1000 values and then stops automatically.

Field	*Meaning*
Group	Selects on which group(s) to execute the action

Script Execution

The action allows you to execute a user-defined script. To upload script files to the device, the “Rules Scripts” page is provided.

Field	*Meaning*
Type	Selects the type of script among: Linux Shell Allows you to run a bash script. Required file extension “.sh” Php

	Allows you to run a Php script. Required file extension ". Php". The file must comply with PHP revision 7.3.9 Python Allows you to run a Python script. Required file extension ". py". The file must comply with Python rev 3.7 Binary program Allows you to run an executable program. Extension required for the file ".bin". The file must be compliant with the 32-bit arm v7 version. In scripts, you can access the Tags through a syntax explained in the relevant chapter of the following manual.
File	Allows you to select the script file from those loaded on the device.
Asynchronous	Allows you to select between: OFF The script is executed in synchronous mode, i.e. the execution of subsequent rules is blocked until the end of the script execution. ON The script is executed in asynchronous mode, i.e. the execution of subsequent rules is not blocked by the script execution.

String Tag

Performs a write to a string-type Tag.

Field	***Meaning***
Action Mode	Select from "One Time" or "Repeat". With "One Time" the action is executed only if there is a change in the result of the OR / AND conditions. With "Repeat" actions are performed at each loop (if the rule is enabled and there is no configured period).
Destination Tag	This is the tag the result is copied to
Operator	
Source Tag / Constant value	Selects the tag to be used as input 1 for the operator used. You can also use a constant value.

8.39. GENERAL SETTINGS PAGE (DATALOGGER)

This section allows to define the general parameters of the datalogger, in particular to edit how the content of the logs will look like.
The datalogger works with the following protocols:

-Via copy to USB/SD card
-EMAIL sending
-FTP sending
-Post http (if active only for group 1)
-MQTT sending

RTU Name
It is the name of the RTU, it appears in the file name in the protocols that send files (Mail and FTP).

Transfer Priority
Allows you to select whether the newest or oldest logs should be sent first.

CSV Separator
Allows you to set the separator in the csv type file between ";" "," " ". It is used only in protocols that send files (Mail and FTP).

Decimal Separator
Allows you to select the decimal separator in values between "," or "."

Floating Point Precision
Allows you to select the precision with which floating point TAGs are sent between:
Automatic, No decimal places or from 1 to 10 digits.

Index Column
Allows you to add an INDEX column to the file with the line number, it is used only in protocols that send files (Mail and FTP).

Type Column
Allows you to add a column to the file with the TYPE field. If the log is of the periodic type then the word "LOG" will always appear, if the log is of the periodic and trigger type then the word SYNC (in the case of a line due to the sampling time) ASYNC (in the case of a sampling line due to a trigger).appears It is used only in protocols that send files (Mail and FTP).

Trigger Column

Allows you to add a column to the file with the TRIGGER field. If the log is of the periodic and trigger type, the source that generated the trigger A, B, is indicated (see logical rules). It is used only in protocols that send files (Mail and FTP).

Timestamp Format

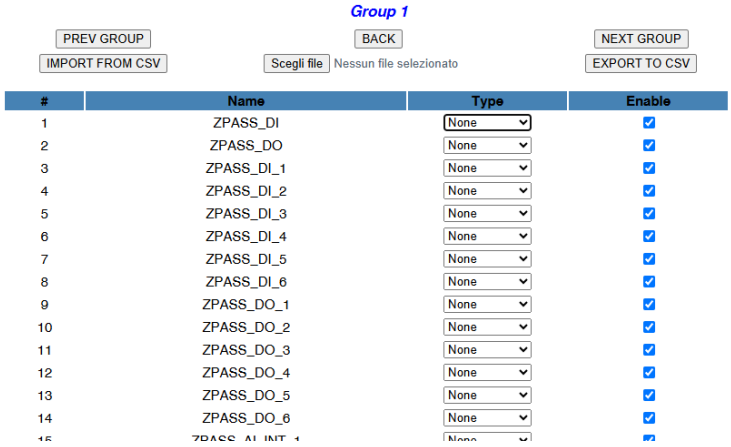
Allows you to set the date and time format in the log. It is used only in protocols that send files (Mail and FTP). In the MQTT protocol, you can choose the timestamp format using the % placeholders.

8.40. GROUP CONFIGURATION PAGE

Here it is possible to select which of the 4 log groups should be activated and the type of log to be made.
It is possible to set a group "disabled".
It is possible to activate the following datalogger modes for each of the 4 groups:

Field	Meaning
Sampling Mode	"Disabled" the group is disabled. "Periodic: All configured tags are acquired with the set time "Periodic and trigger" All configured tags are acquired with the set time and on trigger action. "Trigger" All configured tags are acquired on trigger action. The trigger action can be configured in the logical rules (when a certain series of conditions are fulfilled, the trigger action is executed and the tags are forced to be acquired).
Sampling Period (s)	This parameter defines the sampling period, in seconds.
Transfer Period (min)	This parameter defines the transfer period, in minutes; i.e. each time interval defined by this parameter the log file is closed and transferred.
Number of samples	Indicates the number of samples per file (if a file transfer protocol is used)
SD/USB Enable	Allows you to select the transfer of log files to SD/USB card (if available)
FTP Enable	Allows you to select the transfer of log files to FTP server
EMAIL Enable	Allows you to select the sending of log files via email
HTTP Enable	Allows you to select whether the samples should be sent via http post
MQTT Enable	Allows you to select whether the samples should be sent via mqtt protocol

Time before overflow provides an indication of how much time will pass before unsent data will be overwritten.
For each group, the 'Tag List' button allows tags to be associated with the sampling group (it is also possible to enter the same tag on several groups).
By default, all tags are automatically placed on group1:



#	Name	Type	Enable
1	ZPASS_DI	None	☑
2	ZPASS_DO	None	☑
3	ZPASS_DI_1	None	☑
4	ZPASS_DI_2	None	☑
5	ZPASS_DI_3	None	☑
6	ZPASS_DI_4	None	☑
7	ZPASS_DI_5	None	☑
8	ZPASS_DI_6	None	☑
9	ZPASS_DO_1	None	☑
10	ZPASS_DO_2	None	☑
11	ZPASS_DO_3	None	☑
12	ZPASS_DO_4	None	☑
13	ZPASS_DO_5	None	☑
14	ZPASS_DO_6	None	☑

The 'Type' field allows you to select the type of measure associated with the tag from those available.
This field is used by Cloudbox2 to automatically set the type of widget associated with the tag.

8.41. CLOUD CONFIGURATION PAGE

This page allows you to set the MQTT configuration automatically for the various clouds managed by the device.

Currently, you can configure:
Generic: Through the device's MQTT configurability, it is possible to connect to virtually any cloud
Cumulocity: Sets up the device to connect to the Cumulocity cloud
Direl ADM: Sets up the device to connect to the Direl ADM cloud
On-Board: Sets up the device to connect to the On-Board cloud
Cloudbox2: Sets up the device to connect to the Seneca Cloudbox2 cloud

To add other clouds to the list, you can make a request to Seneca.

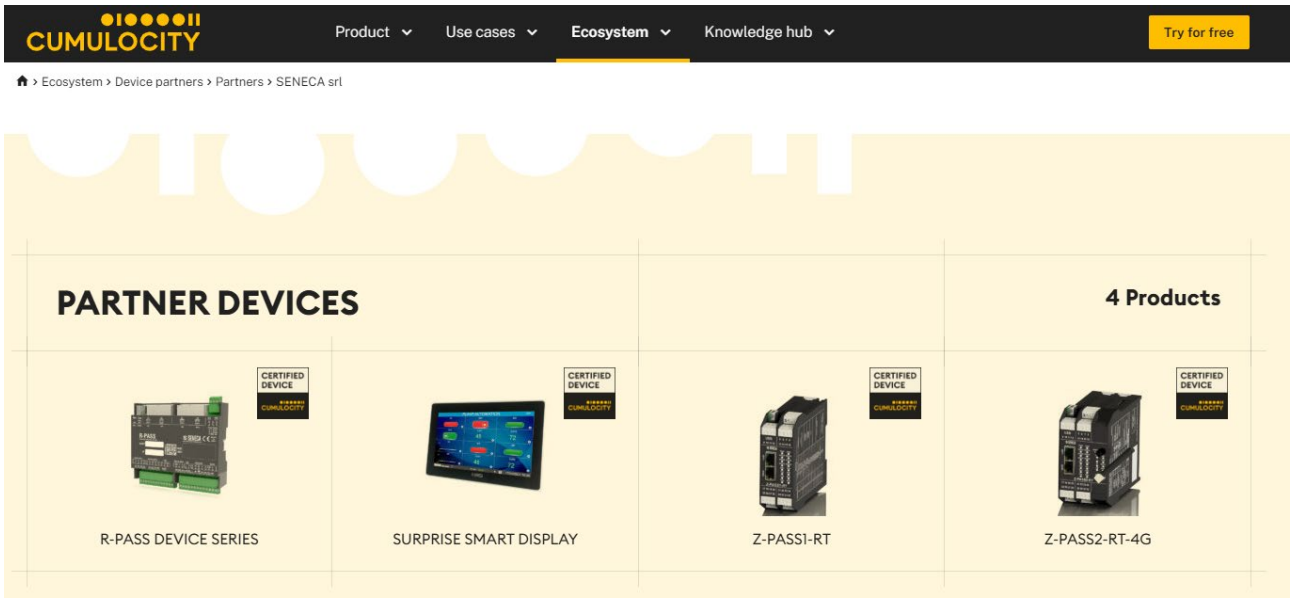
8.41.1.GENERIC

This mode allows you to create MQTT packets payload at will in order to make it compatible with the cloud you want to connect to.
The configuration used can be found on page Client Protocols->MQTT Configuration

8.41.2.CUMULOCITY

The Cumulocity cloud is available at:
https://cumulocity.com/

Seneca devices have passed cumulocity certification tests:



The parameters to be configured are:

Field	Meaning
Enable	Enables or disables connection to the cumulocity cloud
URL	This is the URL where the cloud registration is done
Tenant ID	It is an ID provided by the cumulocity cloud
Username	This is the username for accessing the cloud
Password	This is the password for accessing the cloud

8.41.3.DIREL ADM4.0

The parameters for the Direl cloud (https://www.direl.it/) are as follows:

Field	Meaning
Enable	Enables or disables the connection to the Direl ADM4.0 cloud
Username for Commands	This is the username for writing access from the cloud to the device
Password for Commands	It is the password for writing access from the cloud to the device

8.41.4.ONBOARD

Onboard is the cloud of innovation system s.r.l., for more information refer to the site:

https://www.onsystem-iot.com/onboard

The parameters for the connection are:

Field	Meaning
Enable	Enables or disables the connection to the Onboard cloud
Username	This is the username for accessing the cloud
Password	This is the password for accessing the cloud

8.41.5.CLOUDBOX2

Cloudbox2 is the cloud on-premise di Seneca s.r.l., for more information refer to the site:

https://www.seneca.it

The parameters for the connection are:

Field	Meaning
Enable	Enables or disables the connection to the Seneca Cloudbox2 cloud
Username	This is the username for accessing the cloud
Password	This is the password for accessing the cloud
Site	It is a text representing the site of the system to be monitored
Space	It is a text representing the sub-section of the site of the system to be monitored
Machinery	It is a text representing the machinery of the system to be monitored

8.42. METER-BUS (M-BUS) PROTOCOL

The MBUS protocol is available only if the Straton PLC is active.

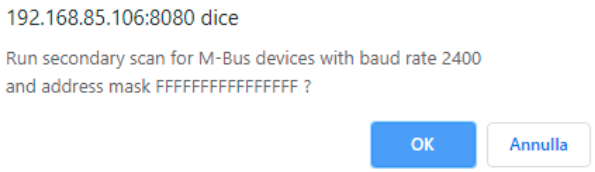
To connect to an M-Bus fieldbus it is necessary to carry out the following steps:
1) connect the optional RS232-MBUS Seneca “Z-MBUS” adapter to the COM1 serial port;
2) setting the COM1 mode to M-BUS.

The following resources are available to manage M-Bus devices:
- the web pages of the "M-Bus" section.
- the MBUS_READ_CTL function
- the MBUS_WRITE_RAW function block

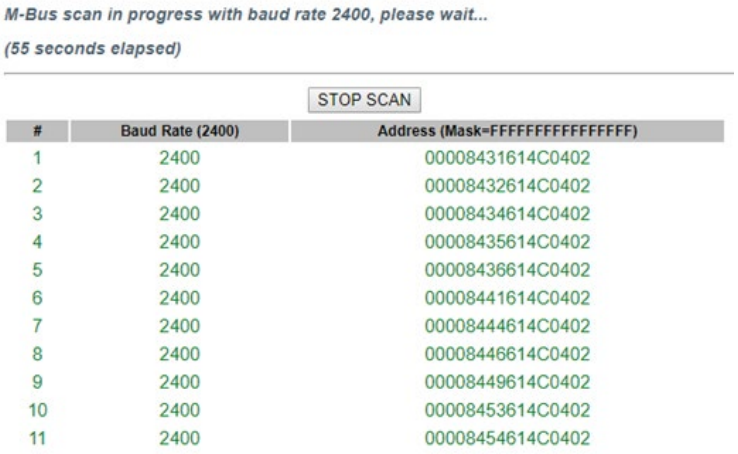
The M-BUS web pages allow you to scan the bus, search for devices, detect their primary or secondary addresses; it also allows you to read data records and slave information from a device and create configuration files for import into the Straton PLC.
The MBUS_READ_CTL FB allows you to start/stop the M-BUS acquisition;
the MBUS_WRITE_RAW FB allows you to build and send a generic M-Bus frame, thus providing a flexible way to send configuration commands to M-Bus devices.

8.42.1. M-BUS SCAN

The "SECONDARY SCAN" button allows you to scan the bus, detecting the M-Bus secondary addresses; select the correct baud-rate for the COM1 serial port or select “ALL” to repeat the scan for each possible baud-rate; then click the button; a confirmation pop-up will appear.

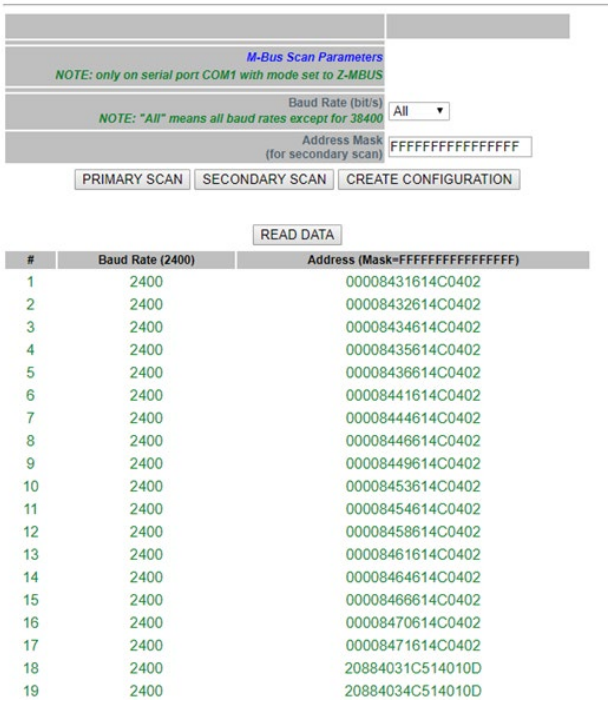


The scanning procedure may take several minutes to complete, so the page shows the number of seconds that have elapsed; devices are displayed in terms of secondary address and baud rate as soon as they are detected.



#	Baud Rate (2400)	Address (Mask=FFFFFFFFFFFFFFF)
1	2400	00008431614C0402
2	2400	00008432614C0402
3	2400	00008434614C0402
4	2400	00008435614C0402
5	2400	00008436614C0402
6	2400	00008441614C0402
7	2400	00008444614C0402
8	2400	00008446614C0402
9	2400	00008449614C0402
10	2400	00008453614C0402
11	2400	00008454614C0402

The "STOP SCAN" button allows you to cancel the procedure; however the partial results are kept. At the end of the procedure, the webserver indicates the end of the scan and then the following page is displayed:

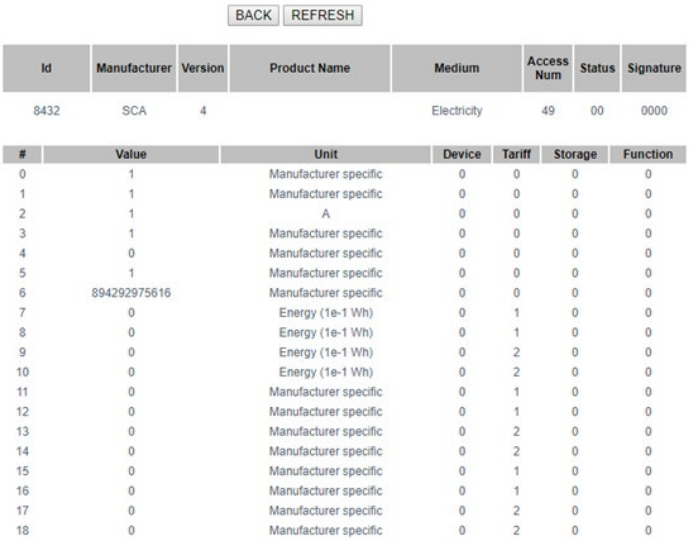


#	Baud Rate (2400)	Address (Mask=FFFFFFFFFFFFFFF)
1	2400	00008431614C0402
2	2400	00008432614C0402
3	2400	00008434614C0402
4	2400	00008435614C0402
5	2400	00008436614C0402
6	2400	00008441614C0402
7	2400	00008444614C0402
8	2400	00008446614C0402
9	2400	00008449614C0402
10	2400	00008453614C0402
11	2400	00008454614C0402
12	2400	00008458614C0402
13	2400	00008461614C0402
14	2400	00008464614C0402
15	2400	00008466614C0402
16	2400	00008470614C0402
17	2400	00008471614C0402
18	2400	20884031C514010D
19	2400	20884034C514010D

The baud rate value shown in the table header reminds you of the parameter choice for the last scan procedure. The table with the detected M-Bus devices is stored permanently, so after switching the device off and on again the results of the last scan are still available; they will be overwritten by the next scan or deleted by a factory reset.

Similarly, the "PRIMARY SCAN" button allows you to scan the bus, detecting the primary M-Bus addresses; select the correct baud-rate for the COM1 serial port or select “All” to repeat the scan for every possible baud-rate.

It is possible to read the data from one of the devices, selecting the corresponding row and clicking on the "READ DATA" button, for example:



Id	Manufacturer	Version	Product Name	Medium	Access Num	Status	Signature
8432	SCA	4		Electricity	49	00	0000

#	Value	Unit	Device	Tariff	Storage	Function
0	1	Manufacturer specific	0	0	0	0
1	1	Manufacturer specific	0	0	0	0
2	1	A	0	0	0	0
3	1	Manufacturer specific	0	0	0	0
4	0	Manufacturer specific	0	0	0	0
5	1	Manufacturer specific	0	0	0	0
6	894292975616	Manufacturer specific	0	0	0	0
7	0	Energy (1e-1 Wh)	0	1	0	0
8	0	Energy (1e-1 Wh)	0	1	0	0
9	0	Energy (1e-1 Wh)	0	2	0	0
10	0	Energy (1e-1 Wh)	0	2	0	0
11	0	Manufacturer specific	0	1	0	0
12	0	Manufacturer specific	0	1	0	0
13	0	Manufacturer specific	0	2	0	0
14	0	Manufacturer specific	0	2	0	0
15	0	Manufacturer specific	0	1	0	0
16	0	Manufacturer specific	0	1	0	0
17	0	Manufacturer specific	0	2	0	0
18	0	Manufacturer specific	0	2	0	0

In this page:

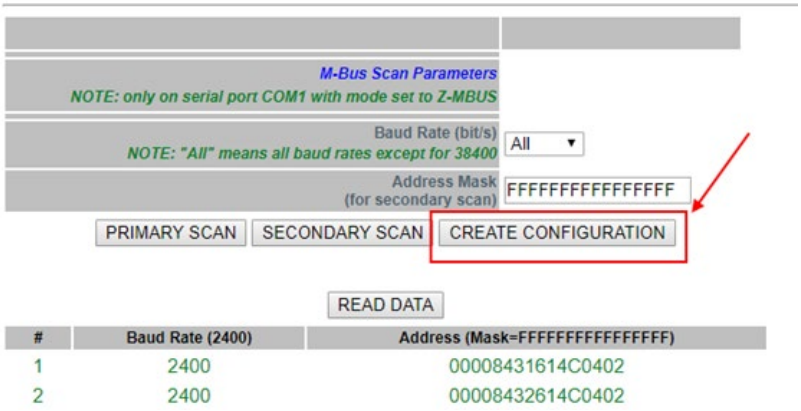
- the first table contains only one line, which provides the "slave information";

- the second table contains a variable number of rows, each of which supplies a "data record".

By clicking on the "REFRESH" button it is possible to update the data; by clicking on the "BACK" button you return to the page with the device table.

8.42.2. "CREATE CONFIGURATION" BUTTON

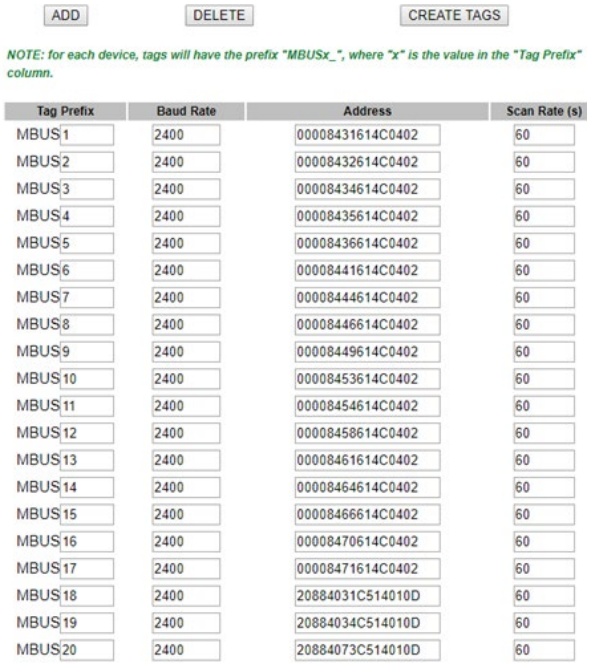
Now you can go back to the previous pages and press the "CREATE CONFIGURATION" button.



This saves the current M-BUS configuration. The web server automatically moves to the next page of "M-Bus Configuration".

8.42.3. M-Bus Configuration

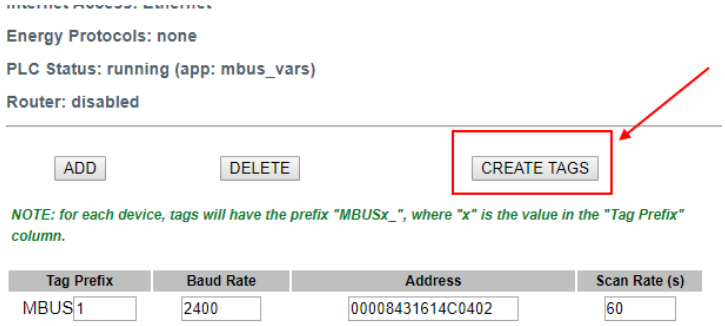
After pressing the "Create configuration" button in the M-Bus Scan page you get the following page in the M-Bus configuration:



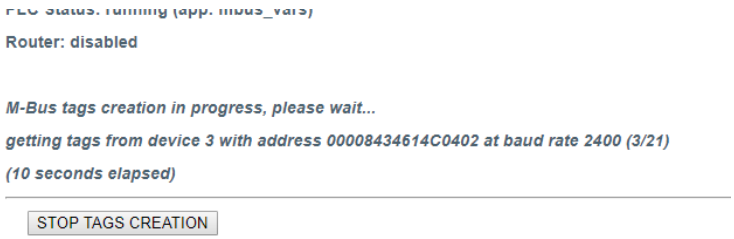
The scan result can now be edited.
The first column represents the Tag Prefix name in Straton
The second column represents the Baud Rate to use.
The third column represents the device address.
The fourth column represents the scan time in seconds for this device.

8.42.4. IMPORTING THE CONFIGURATION INTO STRATON

First of all, we need to export the current configuration.

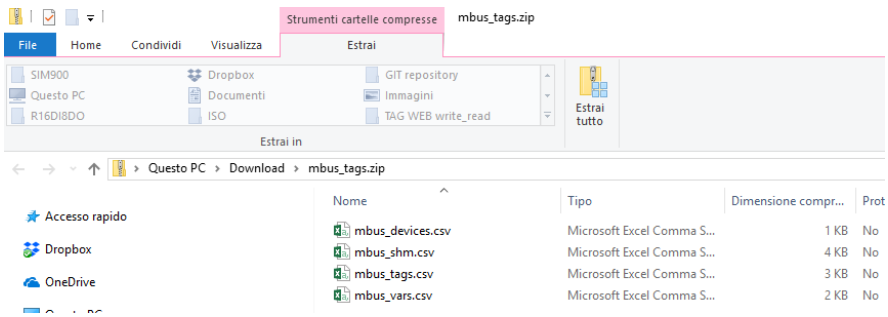


Now the automatic acquisition of tags starts:



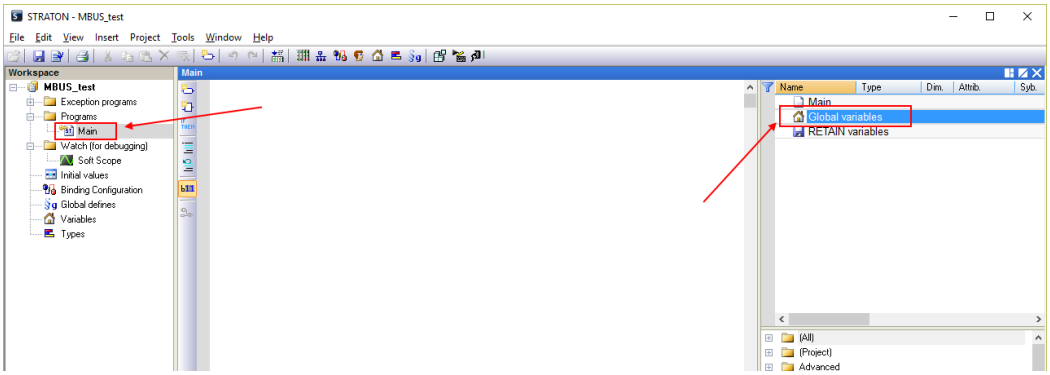
At the end of the process a .zip file (mbus_tags.zip) will be downloaded by the browser:

The .zip file contains 4 files:

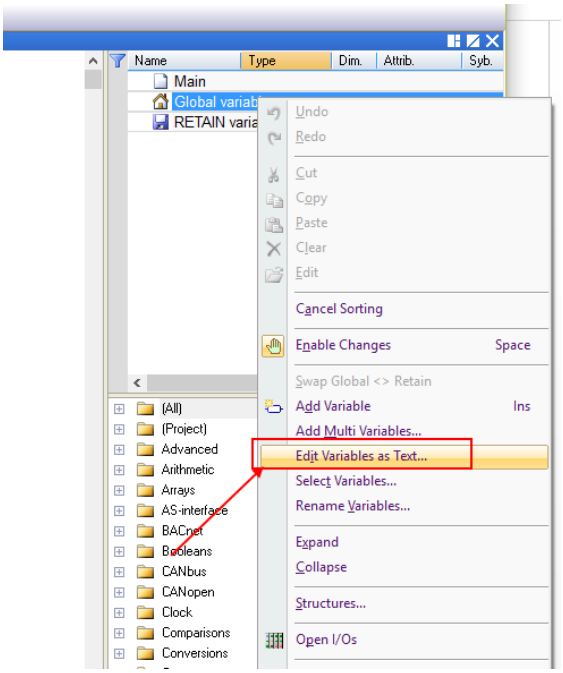


Two of these files are to be used in Straton:
mbus_shm.csv (the shared memory configuration)
mbus_vars.csv (the M-Bus vars)
At this point, perform the following steps:

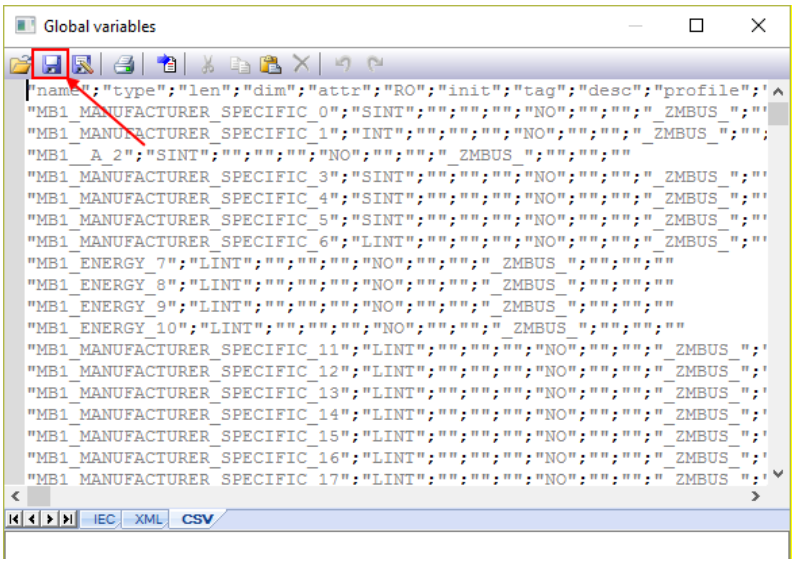
1) Extract the zip file to a directory.
2) Start Straton workbench
3) Select main and then Global variables:



Click the right mouse button and select “Edit Variables as Text”:

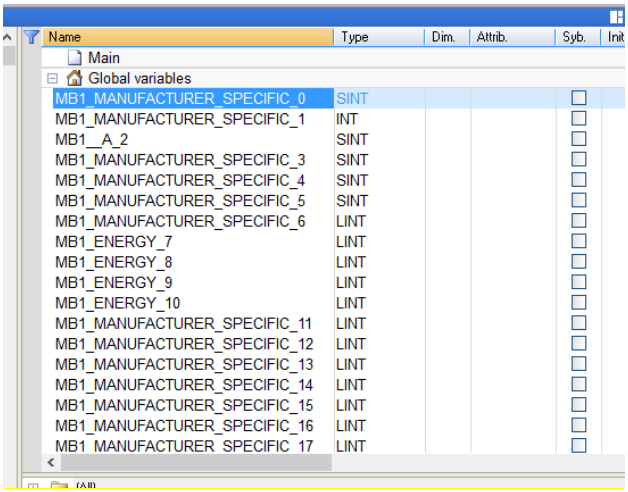


Open the “mbus_vars.csv” file with a text editor, copy and paste the list of variables into the "Global variables" module in Straton then save the configuration with the "disk" icon:

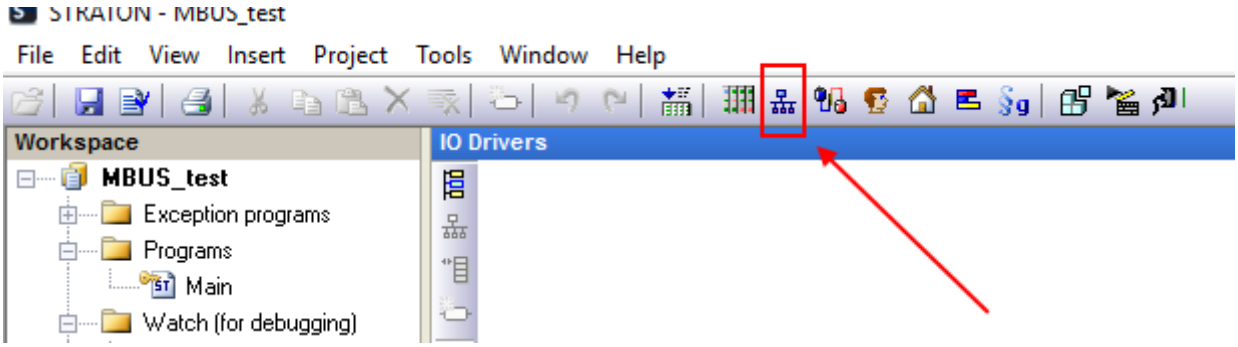


NOTE: The first line
“name”;”type”;”len”;…
must occur only once and only on the first line.

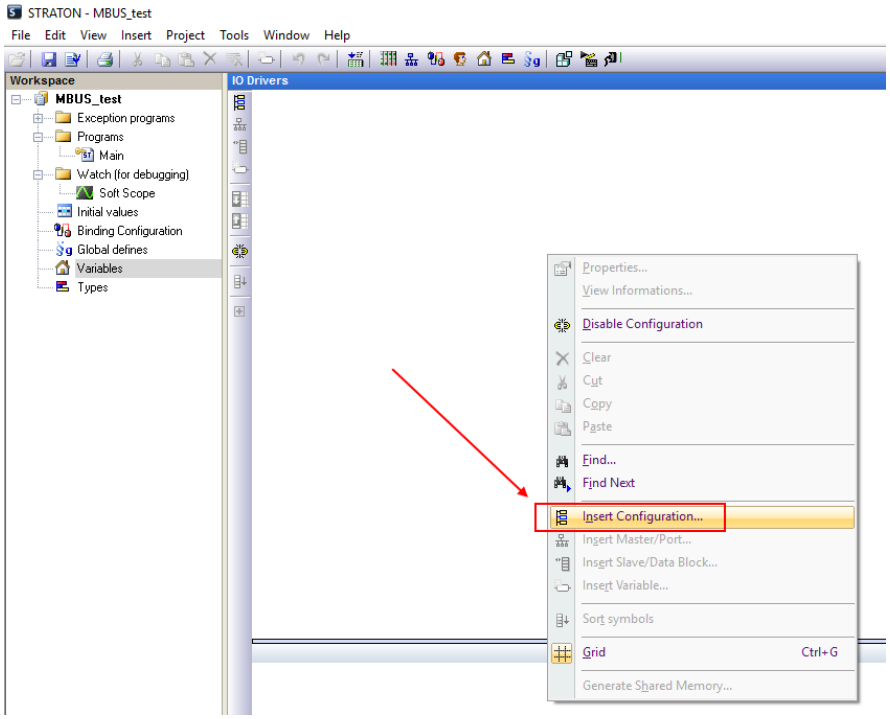
Now the variables are imported:



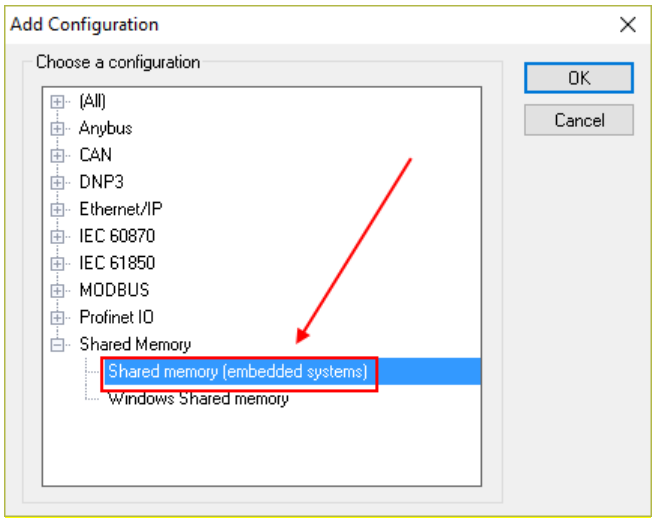
Now we need to create the shared memory used to share data from M-BUS:
Click on the fieldbus icon:



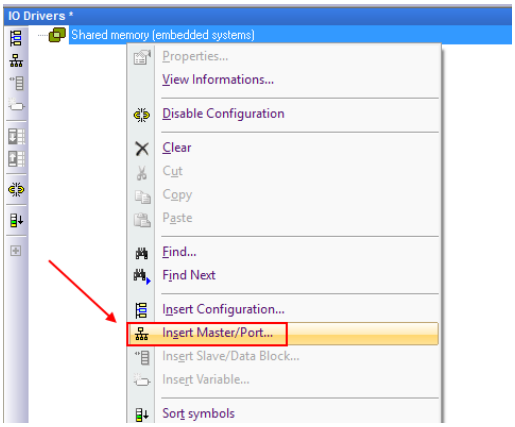
Right-click and select “Insert Configuration”:



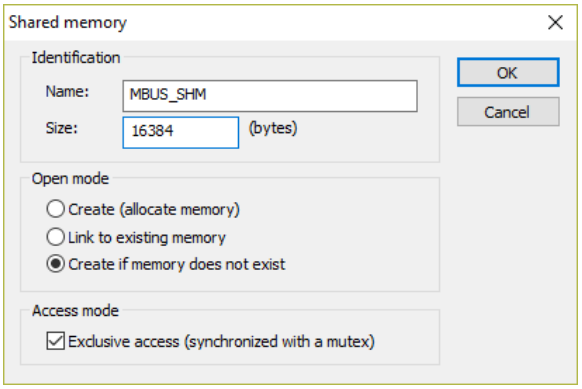
Now create the Shared Memory:



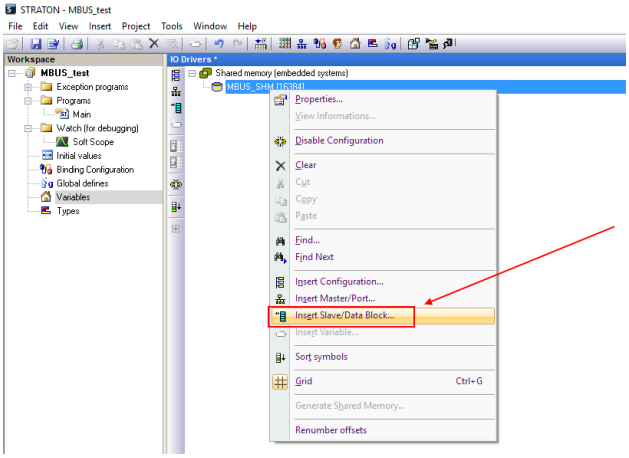
Enter a Master port:



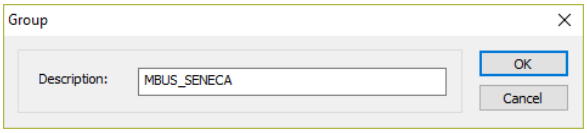
The shared memory configuration must be as shown in the figure (do not change the setting):



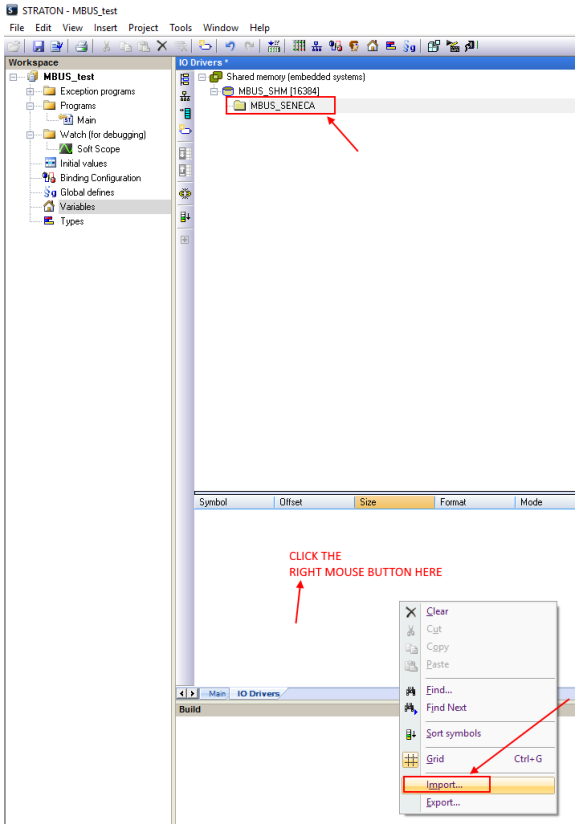
Now insert the data block:



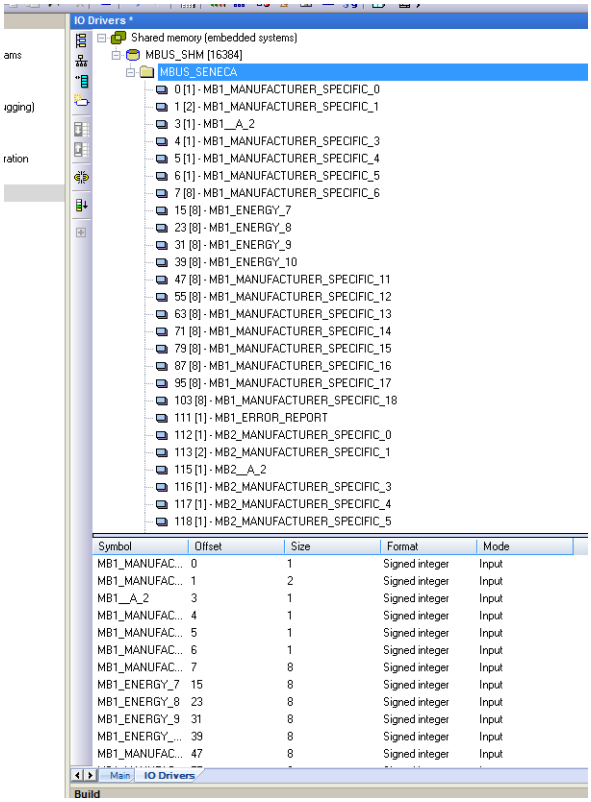
Create a Group and enter a name:



Now import the shared memory file:

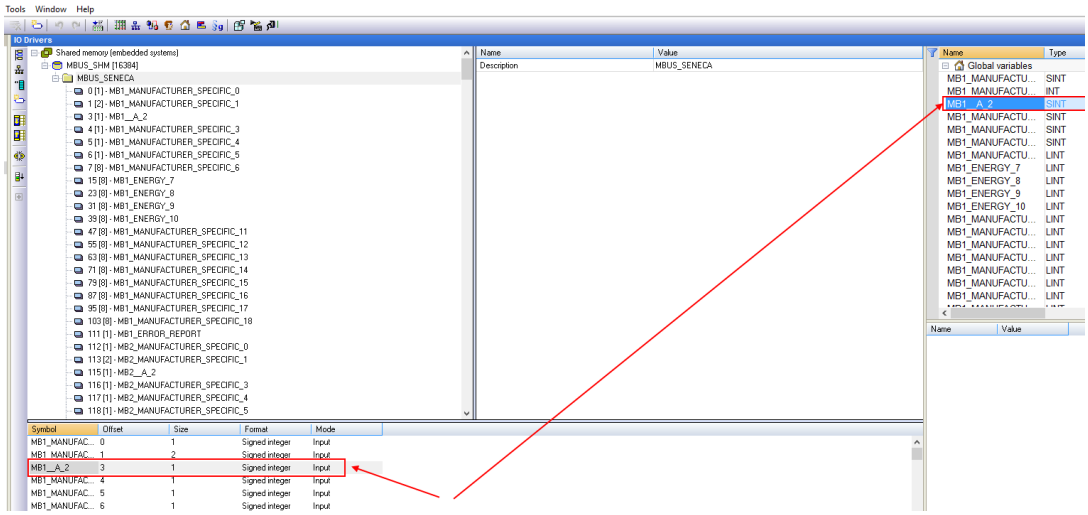


Select the “mbus_shm.csv” file:

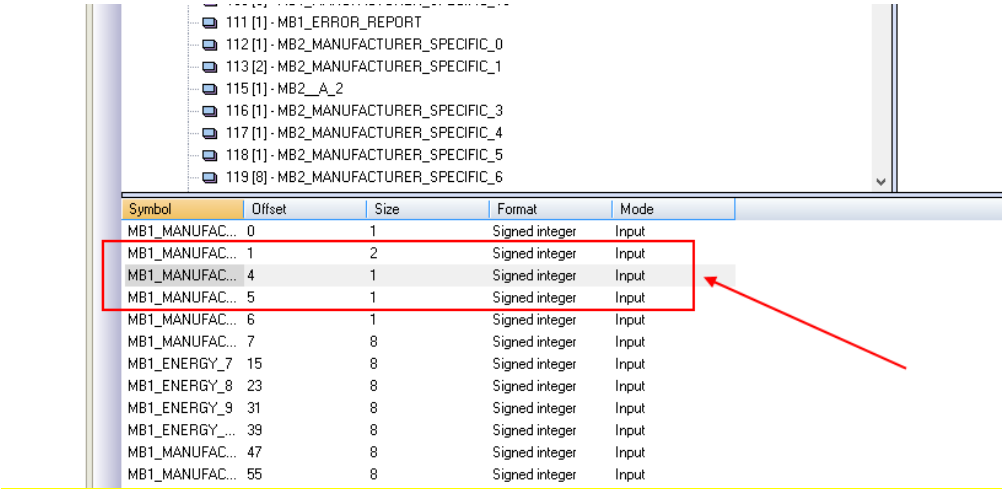


8.42.5. DELETING UNUSED MBUS VARIABLES

To delete one or more variables delete the variables and the corresponding shared memory entry:



Note that in the shared memory the offsets of other variables are not changed:



8.42.6. REPLACING AN M-BUS DEVICE

To replace an existing M-BUS device (e.g. in case of replacement due to failure)

1. Go to M-BUS Scan and do a Secondary or Primary Scan

2. Make a note of the new address

3. Go to M-BUS Configuration and manually change the address from the old to the new device

4. Press the "Create Tag" button.

5. There is no need to make any modifications to the Straton

8.42.7. ADDING AN M-BUS DEVICE

1. Go to “M-BUS Scan” and run a secondary or primary scan
2. Note the new address and baud rate
3. Go to "M-BUS Configuration" and manually add the address and baud rate of the new device with the "ADD" button
4. Press the "Create Tag" button.
5. Import the shared memory file
6. Import the variable file without deleting your local variable (use copy-paste)

8.42.8. DELETING AN MBUS DEVICE

1. Go to M-BUS Scan and do a Secondary or Primary Scan

2. Note the address of the device to be deleted

3. Go to "M-BUS Configuration" and manually delete the device with the "Delete" button.

4. Press the "Create Tag" button.

5. Import the shared memory file

6. Delete the variables from the deleted device

8.42.9. "TAG ERROR REPORT" SPECIAL TAG

When variable tags are imported into Straton, a special "Tag error report" tag is created.
Use this tag to monitor device communication errors:

VALUE OF THE "ERROR REPORT" TAG	*MEANING*
0	READING OK
-2	READING IN TIMEOUT, NO ANSWER FROM THE DEVICE

8.43. CUSTOM IMAGES PAGE (GUI CONFIGURATION)

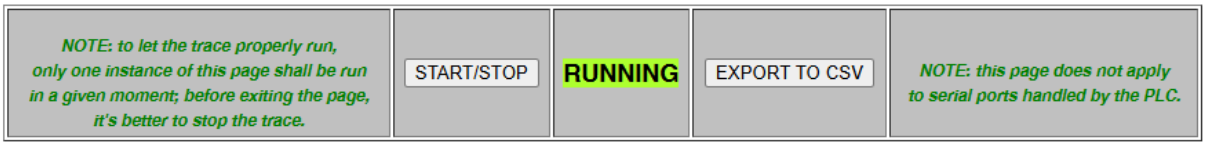
The devices already have a library of hundreds of symbols integrated to be used in their dashboards or synoptics of the physical graphic (in models with a display) or virtual interface.
This page allows you to upload images created by the user (for example to customize the synoptics with company logos etc.).
It is possible to upload .png and .jpg images with a colour depth of 8 bits. It is recommended to upload images with a maximum resolution of 800x 480 pixels.
Once the images are uploaded on this page they will be added to the symbol library.
If you save and export the configuration, the custom images will also be saved.

8.44. ETHERNET INTERFACES PAGE (MAINTENANCE)

The addresses and statistics of the device's Ethernet ports are shown here.

8.45. MODBUS SERIAL TRACE PAGE (MAINTENANCE)

This is a serial sniffer useful for analysing serial traffic. You can also export traffic in csv format for later analysis.



INDEX	TIME DIFF (ms)	PORT	TYPE	LEN	PACKET

8.46. FW VERSION PAGE (MAINTENANCE)

This page lists the revisions of the firmware version in use and the previous installed version.
The device always includes the previous installation.

8.47. FIRMWARE UPGRADE PAGE (MAINTENANCE)

Allows you to update the firmware of the device.

8.48. MANAGEMENT (MAINTENANCE) CONF. PAGE

Allows you to export or import the configuration of the device (useful if you need to copy the configuration to another device).
When exporting a configuration to a file, you are prompted for an encryption passphrase. The same passphrase will be required when importing the configuration. ***If you lose the passphrase, you will no longer be able to import the configuration file.***
Always on the same page you can save the system log files (debug log) to be sent to Seneca support and upload the RSA algorithm key for accessing the ssh service.
It is also possible to reset the ssh key to the default one.

8.49. LICENCE MANAGEMENT (MAINTENANCE)

Here you can check which optional features are enabled under "Optional Features".
It is also possible to enter the activation keys provided by Seneca to add optional features to the device.
For example, it is possible to add Straton PLC function to a device that does not already have it.
For more information please refer to Seneca support.

8.50. MODBUS MODULES (MAINTENANCE)

If you use the PLC in legacy mode and use the legacy Z-NET4 configuration software, the list of connected Modbus devices appears on this page.

8.51. PLC MODE CONFIGURATION (MAINTENANCE)

On this page you can choose the operating mode of the Straton PLC.

Field	Meaning
PLC Mode	"None" the PLC is disabled "Legacy" the PLC is in compatibility mode for use with configurations prior to firmware 3.x.x.x. revision To use the Z-NET4 configuration software it is essential to set the PLC in this mode. This is the default mode for "-S" or "-E" devices. In this mode the virtual display, data logger, alarms etc. are not available "Shared" the PLC is in shared mode, i.e. it can share the TAGs between the PLC and the firmware and therefore take advantage of all the new features of firmware 3.x.x.x. versions Configuration with Z-NET4 is no longer possible

9. VPN

A VPN (Virtual Private Network) is a virtual private network that provides privacy, anonymity and security through a communication channel (VPN tunnel) created over a public network infrastructure.
Devices can create VPNs using Seneca LET'S technology which is based on a VPN BOX 2 server.
It is also possible to connect to standard OpenVPN servers.

For more information on Let's technology visit:

https://www.seneca.it/linee-di-prodotto/comunicazione-industriale-e-telecontrollo/lets-connectivity-solutions/

For further information on OpenVpn, see the website:

https://openvpn.net/

For more information on VPN BOX 2, refer to:

https://www.seneca.it/linee-di-prodotto/comunicazione-industriale-e-telecontrollo/lets-connectivity-solutions/modulo-server-di-connettivita/

The device supports VPN connection using two different servers: Seneca VPN BOX2 and a standard OpenVPN Server.

The main advantages of using a VPN are:

- secure connections so the transmitted data are encrypted;
- the ability to establish connections without interfering with the corporate LAN;
- no need to have a static/public IP address
- on the WAN side; remote configurability via Web Server

Two "VPN modes" are available, respectively named "OpenVPN" and "VPN BOX".

The "OpenVPN" mode can be used when the device is to be installed in an existing VPN. In this case, an OpenVPN server must be available and configuration, certificates and key files for the Seneca Client must be provided by the VPN administrator.

Files can be uploaded to the device using the dedicated web page.

If the VPN infrastructure is not available, the recommended choice is to adopt the "VPN Box2" solution developed by Seneca.
"VPN Box2" is a hardware device (also available in virtual machine version) that allows the user to easily configure two alternative types of VPN:

- "VPN "Single LAN" (Always on for SCADA systems)
- VPN "Point-to-Point" (On demand for remote maintenance of the machine)

In "Single LAN" VPN, all devices and PCs (and associated local sub-networks) configured in VPN are always connected in the same network. In this scenario any VPN Client (PC or Seneca device) can communicate each other but also with the machines/devices connected to any Seneca device LAN, for this reason, all VPN Clients must have different network configuration.

In the "Point-to-Point" VPN, a client PC, at a given moment, can perform a single connection, upon request, to only one device at a time (and to the machines that are connected to the LAN port of the Seneca device). Furthermore, the devices cannot communicate with each other even if they belong to the same VPN.
The advantage of this architecture is that the same subnetwork can be used at all sites. The point-to-point mode is the most used in case of remote maintenance of the systems.

There are two types of Point to Point VPN connection.
- Layer 3 VPN
- Layer 2 VPN

In "Layer 3 VPN", only IP packets (Layer 3) are transported through the VPN tunnel.
On the other side, in "Bridging Layer 2 VPN", all Ethernet frames are transported through the VPN tunnel

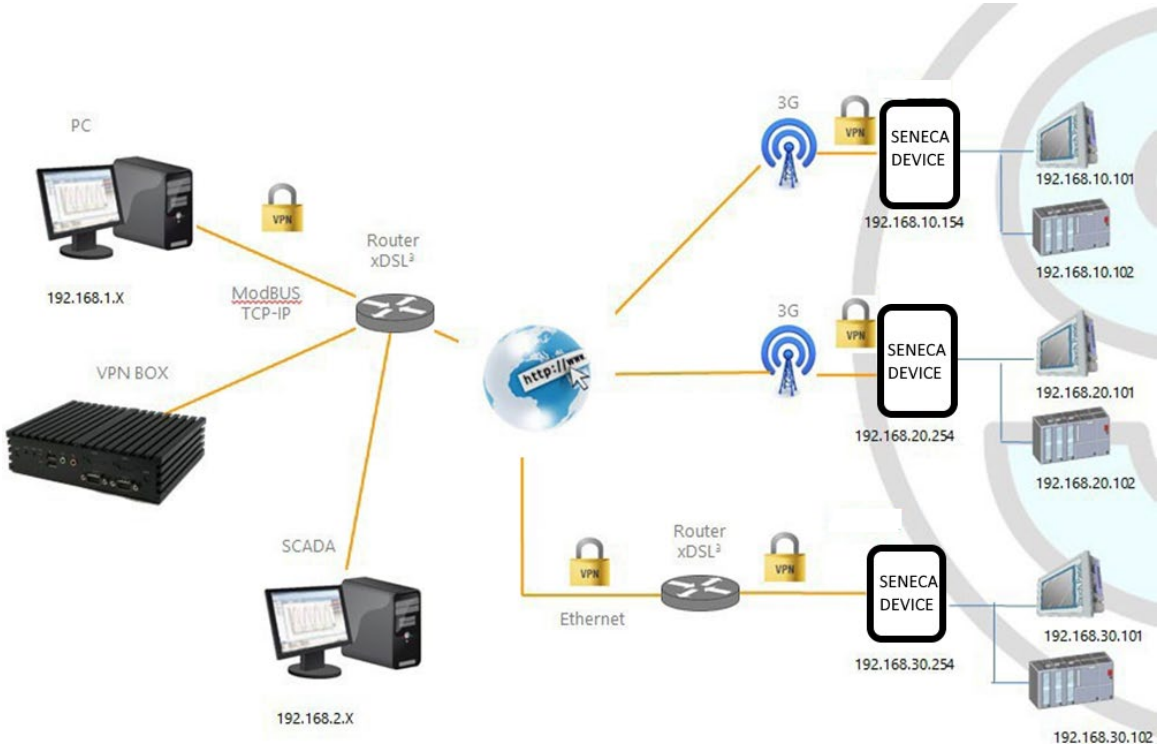
Each mode has advantages and disadvantages:

Layer 2
- can carry any network protocol (e.g. Siemens™ Profinet protocol scanning)
- causes more traffic on the VPN tunnel than layer3

Layer 3
- can carry only IP traffic
- layer2 traffic (e.g.: DHCP) is not transported
- reduces traffic management costs, transports only traffic destined for clients

The "VPN Box2" comes with two Windows applications: "VPN Client Communicator" allows the user to connect the PC to the network (in the "Single LAN" case) or to a specific device (in the "Point-to-Point" case)

9.1. VPN "SINGLE LAN" ALWAYS ON



The figure above provides an example of VPN

The client PC (with IP address 192.168.1.X) can connect, as an example, to the first Seneca device using its local IP address.

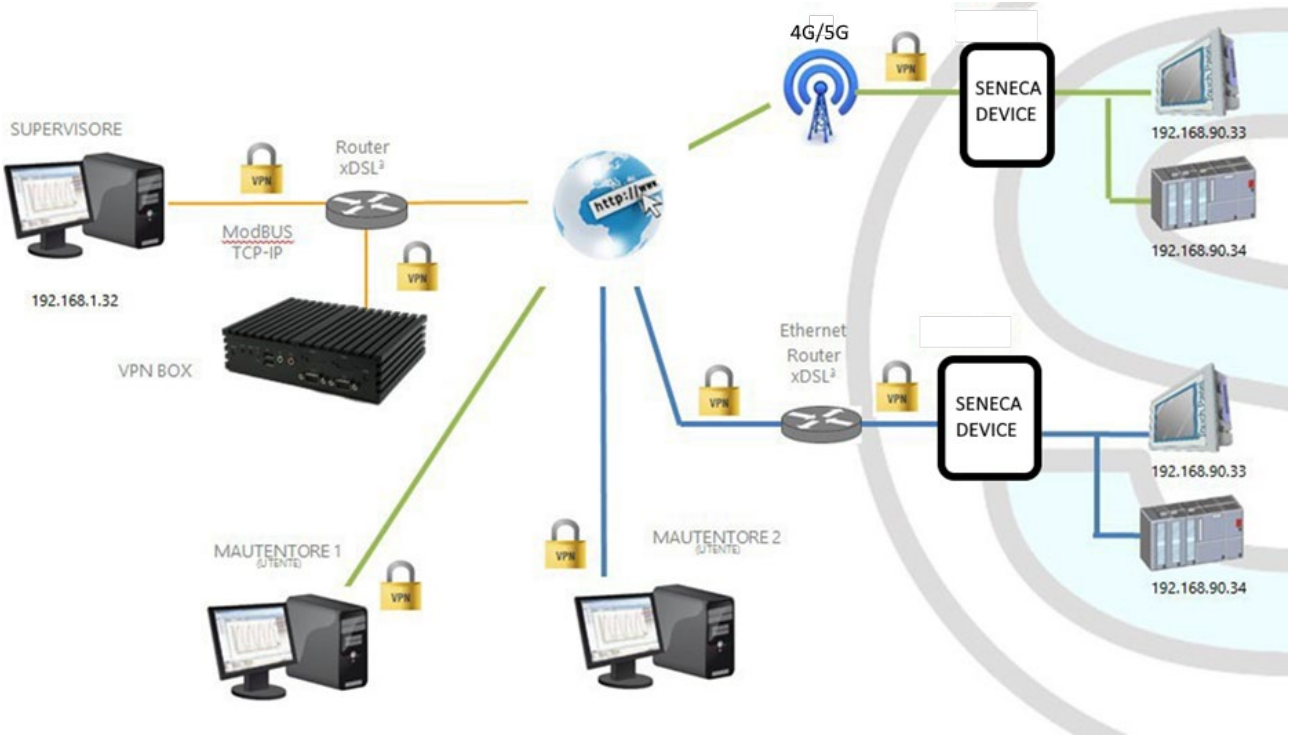
Also, two devices that are in two different LANs of the same VPN network (e.g.: 192.168.10.101 and 192.168.20.102) can connect to each other, again using their local IP addresses.

In order for this scenario to work properly, one essential rule must always be followed: the LANs of the Seneca device and the LAN of the PC must have different subnets and not in collision; therefore, in the figure above, the following is shown

PC LAN	192.168.1.0/24
SCADA LAN	192.168.2.0/24
SENECA DEVICE LAN	192.168.10.0/24
SENECA DEVICE LAN	192.168.20.0/24
SENECA DEVICE LAN	192.168.30.0/24

If conflicts cannot be avoided, it is still possible to use a "Single LAN" VPN because devices can be reached via their VPN IP addresses and machines beyond them can be reached by configuring "port forwarding" rules.

9.2. VPN "POINT TO POINT" ON DEMAND



The figure above provides an example of "Point-to-Point" VPN.

In this scenario a PC (acting as a VPN client) can connect, on demand, to a Seneca device and its subnet using local IP addresses via the VPN Client Communicator application. The software guarantees group management of users to allow only those who belong to a group to access the systems that are part of it

9.3. DISABLING THE VPN CONNECTION

The products provide an integrated digital input and digital output dedicated to control and monitor the remote connection to the device.
In this way it is possible to block access (via digital input) remotely to a particular machine/plant (e.g. if local maintenance operations are being carried out) and be informed of a remote access in progress (via digital output).
When the "Remote Connection Disable" digital input is set to HIGH, the remote connection to the device is disabled; conversely, when the "Remote Connection Disable" digital input is set to LOW, the remote connection to the device is enabled.

The "Remote Connection Active" digital output is set to the HIGH state when the device is connected.
Four security levels can be configured to disable the remote VPN connection:
Level 1: VPN connections are disabled in any VPN mode but the "VPN Box Service" service is still running, so the device can still be monitored on VPN Box Manager;

Level 2: The "VPN Box Service" is disabled, but the device can still access the Internet and send/receive SMS on a possible cellular interface;
Level 3: any Internet access is disabled, but the device can still send/receive SMS on a possible cellular interface;
Level 4: As level 3 but also the cellular interface is switched off

9.4. CONFIGURATION FILE FOR USE WITH OPEN VPN

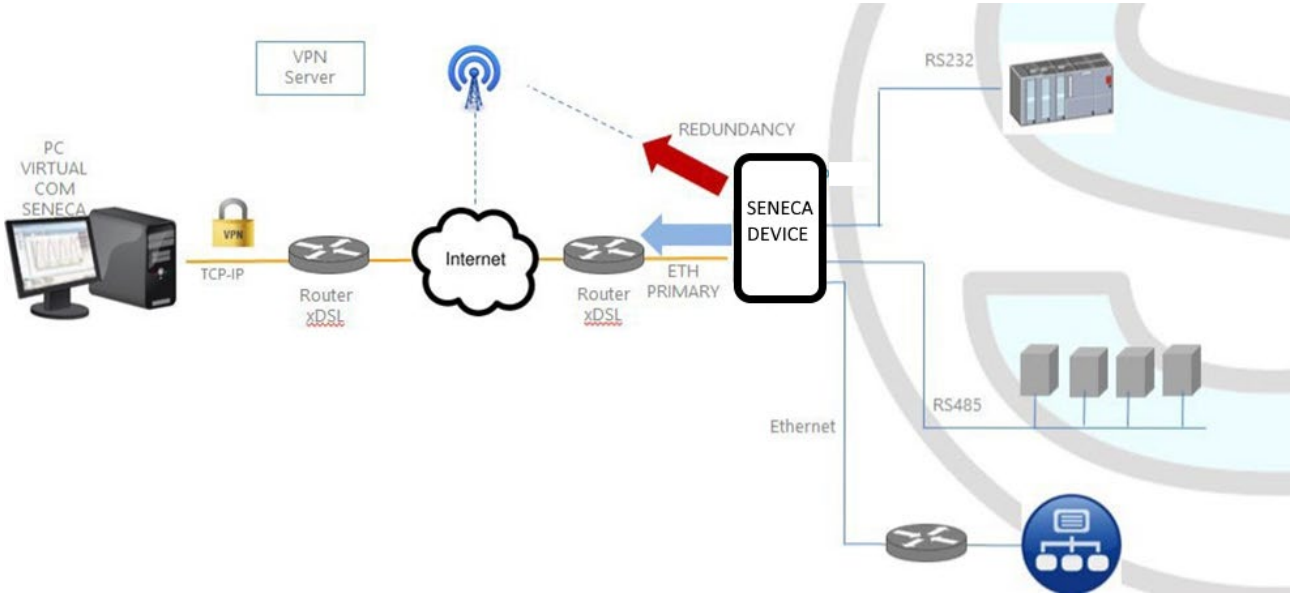
This paragraph provides an example of OpenVPN server configuration.

```
port 1194
proto udp
dev tun
ca ca.crt
cert server.crt
key server.key
dh dh1024.pem
server 10.9.7.0 255.255.255.0
ifconfig-pool-persist ipp.txt
client-config-dir ccd
client-to-client
keepalive 10 120
comp-lzo
persist-key
persist-tun
status openvpn-status.log
verb 3
```

This paragraph provides an example of the device client Open VPN configuration.

```
client
dev tun
port 1194
proto udp
remote 2.192.5.105 1194
nobind
ca ca.crt
cert tws4.crt
key tws4.key
comp-lzo
persist-key
persist-tun
script-security 3 system
verb 3
```

10. COMMUNICATION NETWORK REDUNDANCY



“Network redundancy” is a feature that can be enabled on devices where a mobile or WI-FI modem is available.

This feature is intended to switch the network interface used to access the Internet from Ethernet ("primary" interface) to the secondary interface (Cellular modem or WI-FI), when access to the Internet through the primary interface becomes unavailable, the system draws on the Internet through the configured secondary channel. When the Internet service becomes available again from the primary interface the access returns to the latter.

11. MQTT CLIENT PROTOCOL

MQTT is the most widely used protocol for IOT applications.

"MQTT" stands for MQ Telemetry Transport. It is an extremely simple and lightweight public/subscription messaging protocol designed for devices with low bandwidth, high latency or unreliable networks. The design principles are to minimize network bandwidth and device resource requirements while trying to ensure reliability and a certain degree of delivery guarantee. These principles prove ideal for the emerging machine-to-machine (M2M) or Internet of Things world.

For more information on the MQTT protocol see

The MQTT version supported is 3.1.1

11.1. MQTT PROTOCOL IMPLEMENTATION FEATURES

The MQTT protocol can be enabled together with the other client protocols (USB, FTP, EMAIL, …); however, when the MQTT protocol is enabled, the following changes apply to the behaviour of the Data Logger

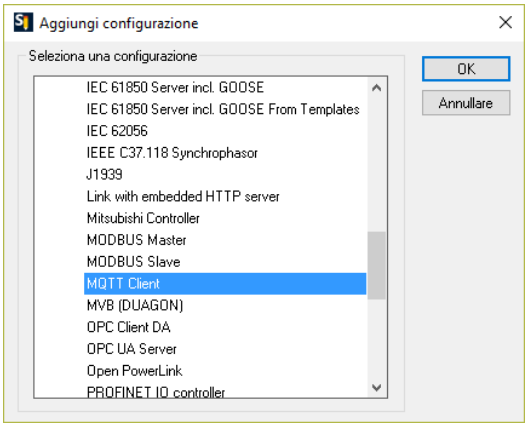
The MQTT protocol also allows you to perform the following actions on the device:

- setting the values of one or more tags
- restarting the device
- save the device configuration on the FTP site of the server
- upload the device configuration from the FTP site of the server
- starting the FW update;

There is also an internal cache for LOG messages sent via MQTT requests, used to store log messages while it is not possible to send them to the broker; this cache can hold up to 3000 messages

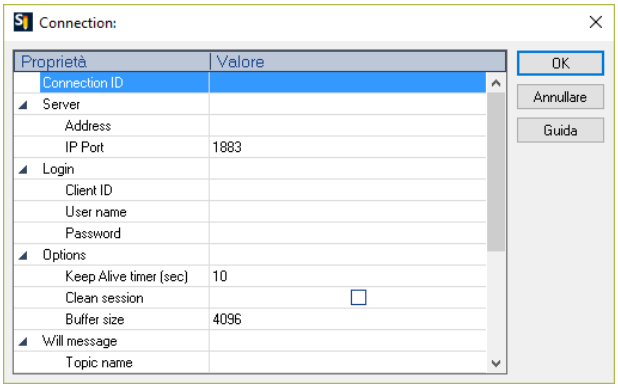
11.2. FEATURES OF THE MQTT PROTOCOL IMPLEMENTATION OF THE STRATON PLC

The MQTT version supported is 3.1.1
To use the MQTT client select it from the Straton Workbench Fieldbus section:



11.2.1. PARAMETERS OF THE MQTT PROTOCOL FROM THE PLC PROGRAM

MQTT setup can be done directly from the workbench:



If it is necessary to configure these parameters from the Straton PLC program, a series of special words can be used which will load the configuration from a file.
The special words are:

In the "Address" field type: mqtt_par_address so that the "Address" field is obtained from the file:

/var/run/mqtt_par_address

In the “Client ID” field type: mqtt_par_clientid so that the “Client ID” field is obtained from the file:
/var/run/mqtt_par_clientid
In the "Username" field type: mqtt_par_username so that the "Username" field is obtained from the file:
/var/run/mqtt_par_username

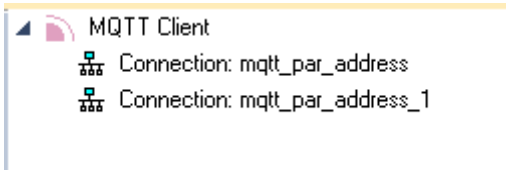
In the "Password" field type: mqtt_par_password so that the "Password" field is obtained from the file: /var/esegui/mqtt_par_password

ATTENTION!

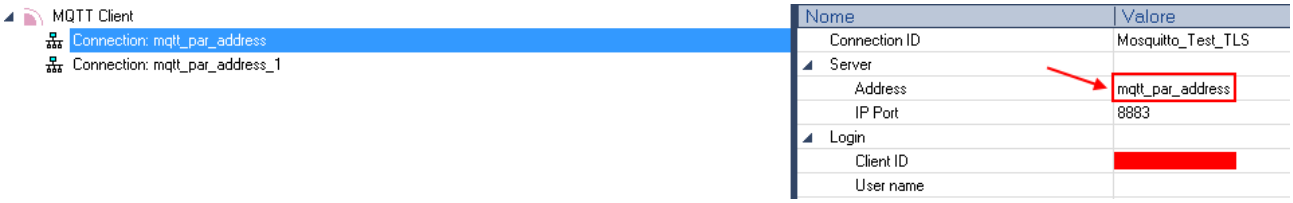
The Address parameter must not contain a FQDN, but the IP address, this is because the MQTTCONNECT FB does not perform DNS resolution.
Alternatively, it can contain the name of the file (e.g.: mqtt_par_address), created in the /var/run directory by the DNS_RESOLVE FB and containing the result of the DNS resolution.

11.2.2. MANAGING MULTIPLE MQTT CONNECTIONS

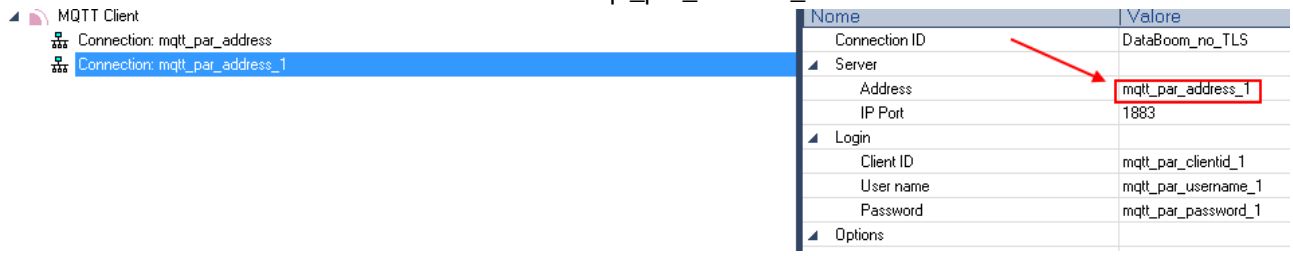
It is possible to manage multiple MQTT connections using parameters starting with the special words (mqtt_par_address123, mqtt_par_address_aaa, …), for example to create 2 mqtt connections:



The first connection uses the Field address "mqtt_par_address"



Then it will load the address from the file:
/var/run/mqtt_par_address
The second connection uses the stored address "mqtt_par_address_1"



this will load the address from the file:
/var/run/mqtt_par_address_1
(the technique can also be used for the other client id, username and password parameters).

11.2.3. MQTT CONFIGURATION OF SSL/TLS RETRYs

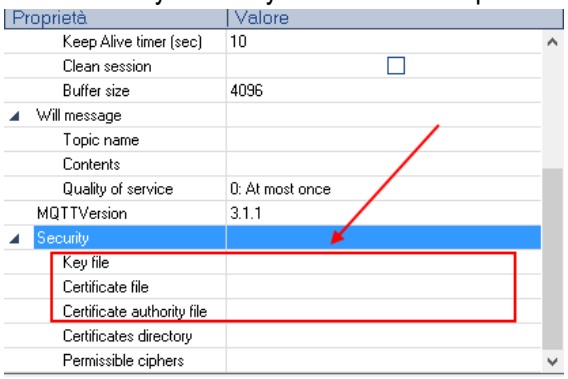
The default configuration for MQTT SSL/TLS connection is:
CONN _TRY_MAX = 10
CONN_TRY_WAIT = 1000 ms
Where:
CONN _TRY_MAX is the number of attempts to connect.
CONN_TRY_WAIT is the timeout of each connection attempt.
If you need to change this default configuration you need to create the file:
"ssl_con_try_params"
In this path:
"/var/esegui/"
With parameter values, for example:

root@Z-PASS2-S:~# cat /var/run/ssl_conn_try_params
50.200

It means CONN _TRY_MAX = 50 and CONN_TRY_WAIT = 200 ms.
NOTE1: At the end of the file you need to add an \n (new line character)
NOTE2: The file is loaded into a RAM filesystem, so you need to create it on every boot.

11.2.4. STATIC AND DYNAMIC CLIENT CERTIFICATES

In the MQTT configuration under the Security section you can enter the path and file name for the certificates:



Seneca suggests using the /config directory for certificates.
The MQTT client certificate can only be uploaded from the FTP server.
The key file is the client's private key file.
The certificate file is the client certificate.
The certification authority file is the certification authority certificate.

ATTENTION!

The "Certificate directory" field is not used so the file name must contain the absolute path example:
"/config/mqtt/client.key"
"/config/mqtt/client.crt"
"/config/mqtt/ca.crt"
If these files and other parameters need to be modified dynamically without recompiling the project, a file can be loaded into the /var/run directory with a file name that must start respectively with:
"mqtt_par_clientkey", "mqtt_par_clientcert", "mqtt_par_cacert"
The content of the files must be a text with the file name without the path.
Note that more than one certificate file can be used in a program, for example "mqtt_par_clientcert00", "mqtt_par_clientcert01" etc...

11.2.5. CHANGING MQTT PARAMETERS IN RUNTIME VIA FILE

You can change the port and the keepalive configuration by overwriting the current configuration with the following files in runtime:
"mqtt_par_port" and "mqtt_par_keepalive".

The content of the files must be a text with the new parameter value.

12. LOGICAL RULES

A logical rule is based on the following concept

“IF -> THEN -> ELSE”

It means:

IF THE CONDITION HAS OCCURRED -> THEN PERFORM THESE ACTIONS -> OTHERWISE PERFORM THESE OTHER ACTIONS

You can define up to 2000 rules.

In each rule can also be configured:

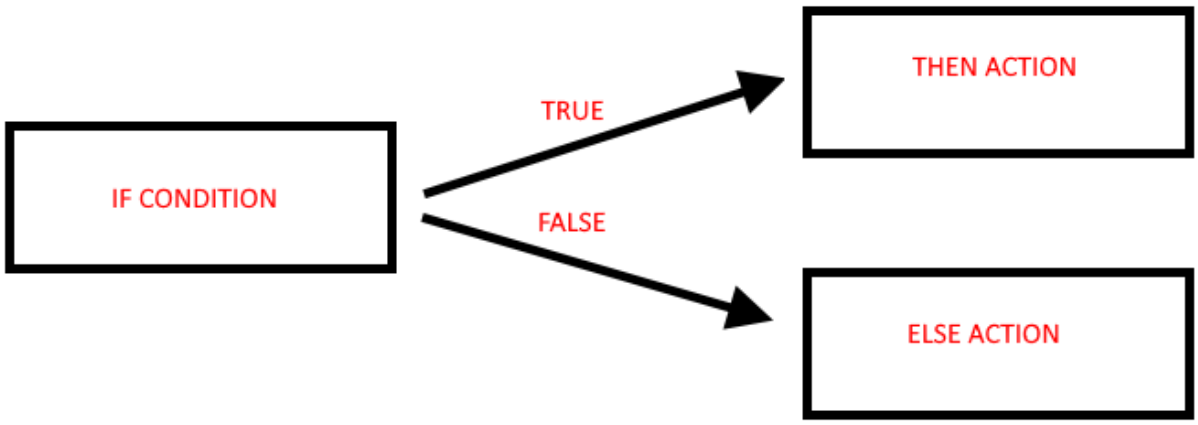
- Combinations of up to three logical conditions (based on alarm states) in AND/OR logical expression;
- up to three actions can be performed

Using logical rules it is therefore possible to execute programs that use internal or external I/O, send text messages and audio call and/or write TAGs via MODBUS / EMAIL / SMS / http / MQTT etc. even using complex mathematical operations.

Rules can also be debugged through step-by-step execution and the use of breakpoints that block program execution on a specific line (rule).

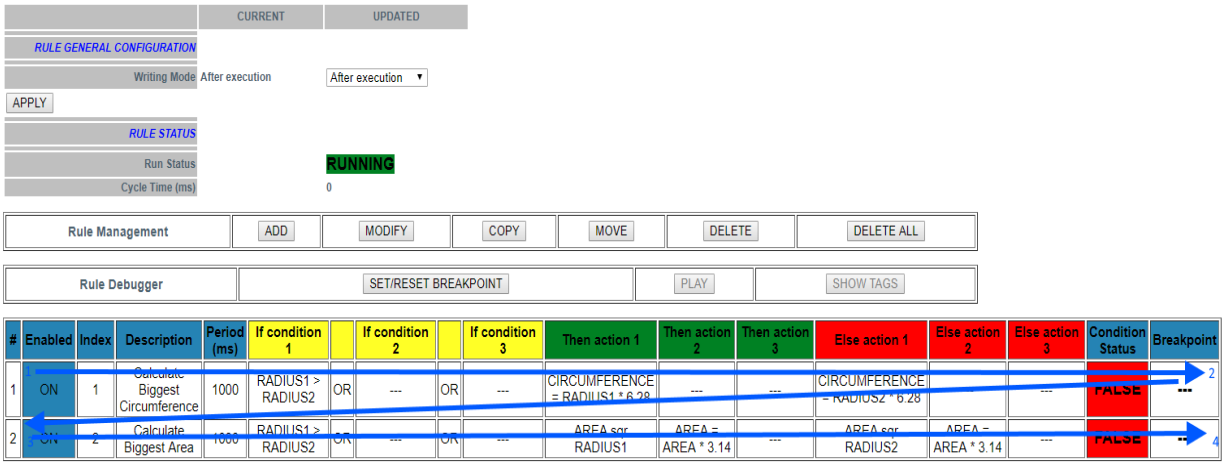
A rule consists of one or more "If Condition", one or more "Then Action" and one or more "Else Action".

Schematically a rule performs the following flow:

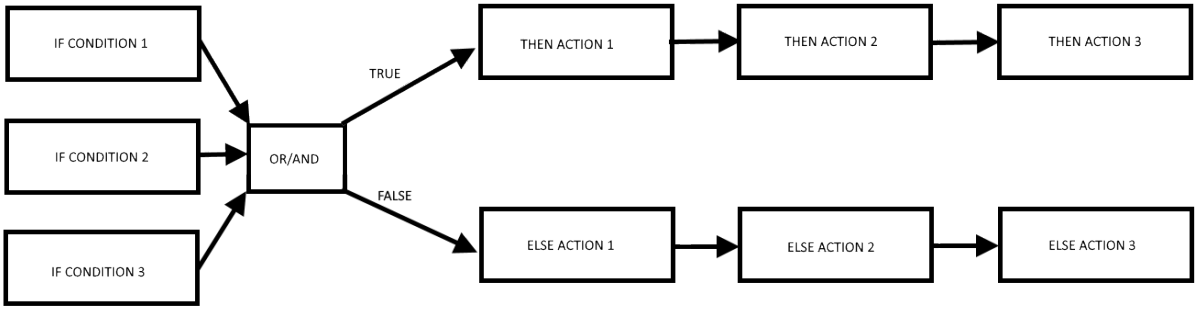


If the "IF" condition is true, the "THEN" action is executed, otherwise the "ELSE" action is executed.

The rules are executed from top to bottom and from left to right (in figure 1-> 2-> 3-> 4):



When all the rules are executed, the execution starts again from the first one.
More in detail the correct diagram is:



It is in fact possible to define up to 3 if conditions and up to 3 actions for both the THEN and ELSE state.

It is possible to create up to 2000 different rules.

In the "Rule General Configuration" you can choose when Tags are written to shared memory, you can choose between "After Execution" or "During Execution".

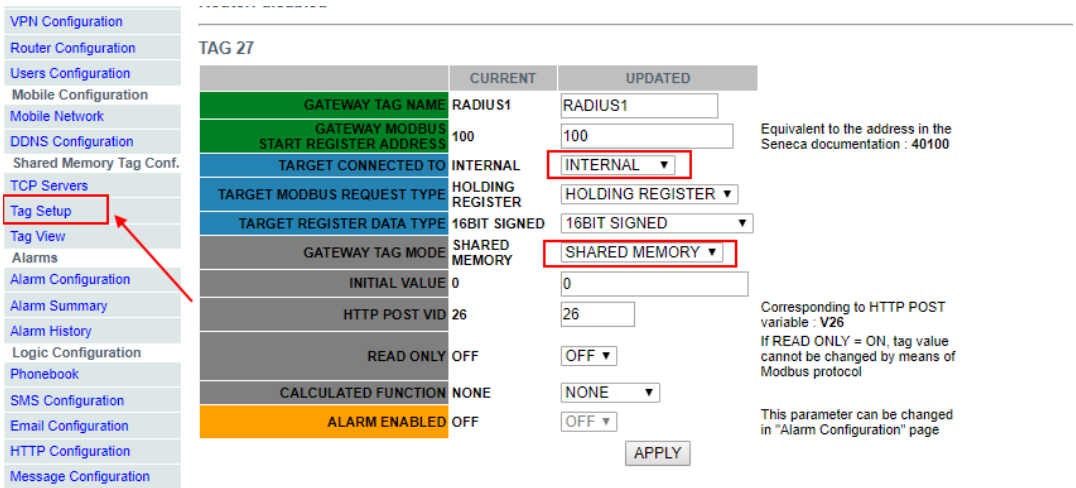
With "After Execution", you get tag values written to shared memory only when all the rules HAVE been executed.
With "During Execution", you get tag values written to shared memory at the end of each single rule.

Therefore, using the "After Execution" mode, the new tag values will only be updated at the end of all rules (even tags that must be written on MODBUS RTU / TCP-IP).
The rule status will show the execution status (if the rules are in execution or pause mode) and the loop time which represents the time taken to execute all the rules (note that if you need to write tags with modbus protocol, the loop time will also include the time taken for this operation).

12.1. CREATION OF A PROGRAM WITH LOGICAL RULES

We will create an example program that, given 2 different radii of a circle, calculates the maximum circumference and the maximum area.

First of all, we add the Tags we need for the program:
We define the Radius1 and Radius2 tags as integer type
Circumference and Area in Real 32 bits (floating point single precision):



	CURRENT	UPDATED	
GATEWAY TAG NAME	RADIUS2	RADIUS2	
GATEWAY MODBUS START REGISTER ADDRESS	101	101	Equivalent to the address in the Seneca documentation : **40101**
TARGET CONNECTED TO	INTERNAL	INTERNAL ▼	
TARGET MODBUS REQUEST TYPE	HOLDING REGISTER	HOLDING REGISTER ▼	
TARGET REGISTER DATA TYPE	16BIT SIGNED	16BIT SIGNED ▼	
GATEWAY TAG MODE	SHARED MEMORY	SHARED MEMORY ▼	
INITIAL VALUE	0	0	
HTTP POST VID	27	27	Corresponding to HTTP POST variable : **V27**
READ ONLY	OFF	OFF ▼	If READ ONLY = ON, tag value cannot be changed by means of Modbus protocol
CALCULATED FUNCTION	NONE	NONE ▼	
ALARM ENABLED	OFF	OFF ▼	This parameter can be changed in "Alarm Configuration" page
		APPLY	

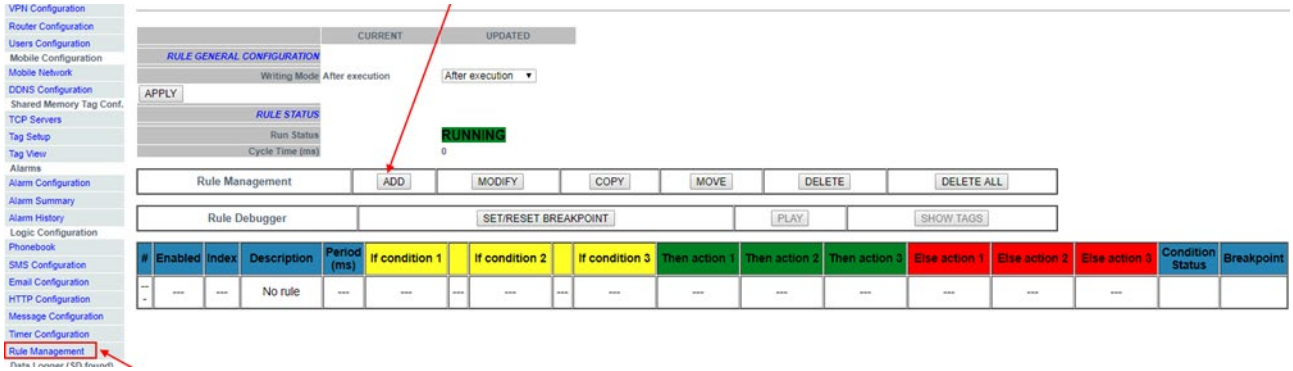
TAG 29

	CURRENT	UPDATED	
GATEWAY TAG NAME	CIRCUMFERENCE	CIRCUMFERENCE	
GATEWAY MODBUS START REGISTER ADDRESS	103	103	Equivalent to the address in the Seneca documentation : **40103**
TARGET CONNECTED TO	INTERNAL	INTERNAL ▼	
TARGET MODBUS REQUEST TYPE	HOLDING REGISTER	HOLDING REGISTER ▼	
TARGET REGISTER DATA TYPE	32BIT REAL MSW	32BIT REAL MSW ▼	
GATEWAY TAG MODE	SHARED MEMORY	SHARED MEMORY ▼	
INITIAL VALUE	0	0	
HTTP POST VID	28	28	Corresponding to HTTP POST variable : **V28**
READ ONLY	OFF	OFF ▼	If READ ONLY = ON, tag value cannot be changed by means of Modbus protocol
CALCULATED FUNCTION	NONE	NONE ▼	
ALARM ENABLED	OFF	OFF ▼	This parameter can be changed in "Alarm Configuration" page
		APPLY	

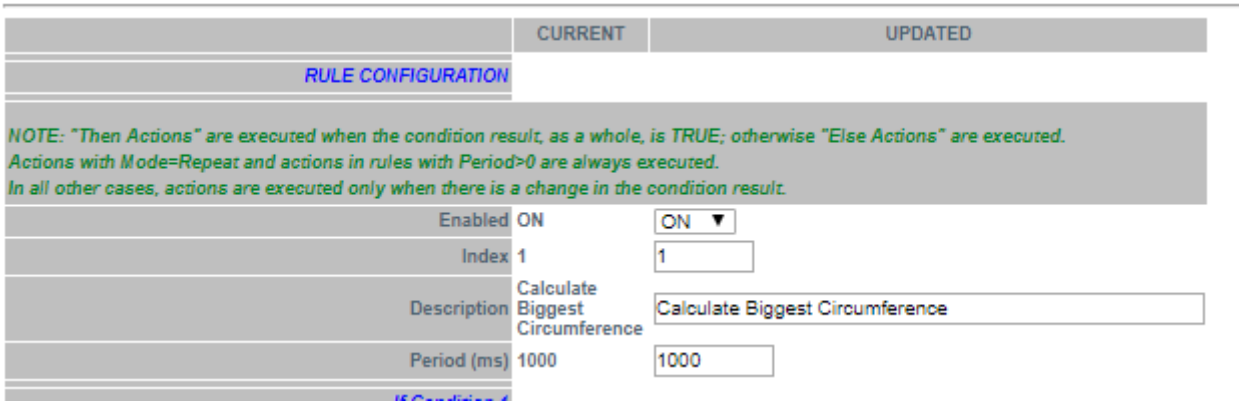
TAG 30

	CURRENT	UPDATED	
GATEWAY TAG NAME	AREA	AREA	
GATEWAY MODBUS START REGISTER ADDRESS	105	105	Equivalent to the address in the Seneca documentation : **40105**
TARGET CONNECTED TO	INTERNAL	INTERNAL ▼	
TARGET MODBUS REQUEST TYPE	HOLDING REGISTER	HOLDING REGISTER ▼	
TARGET REGISTER DATA TYPE	32BIT REAL MSW	32BIT REAL MSW ▼	
GATEWAY TAG MODE	SHARED MEMORY	SHARED MEMORY ▼	
INITIAL VALUE	0	0	
HTTP POST VID	29	29	Corresponding to HTTP POST variable : **V29**
READ ONLY	OFF	OFF ▼	If READ ONLY = ON, tag value cannot be changed by means of Modbus protocol
CALCULATED FUNCTION	NONE	NONE ▼	
ALARM ENABLED	OFF	OFF ▼	This parameter can be changed in "Alarm Configuration" page
		APPLY	

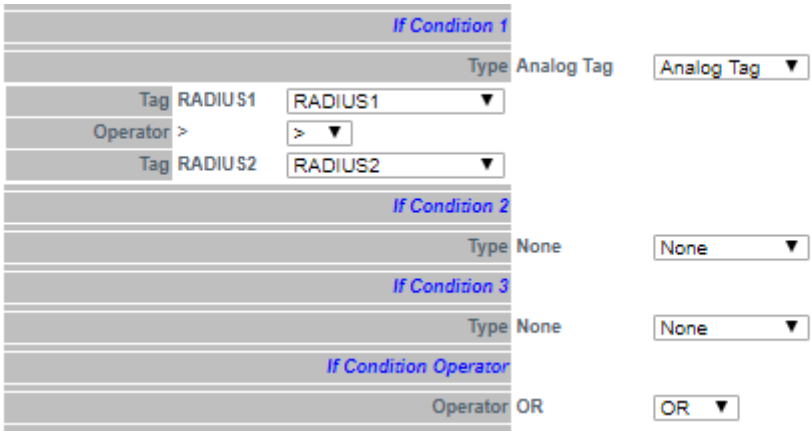
Now click on "Rules Management" and then on ADD to add a new rule:



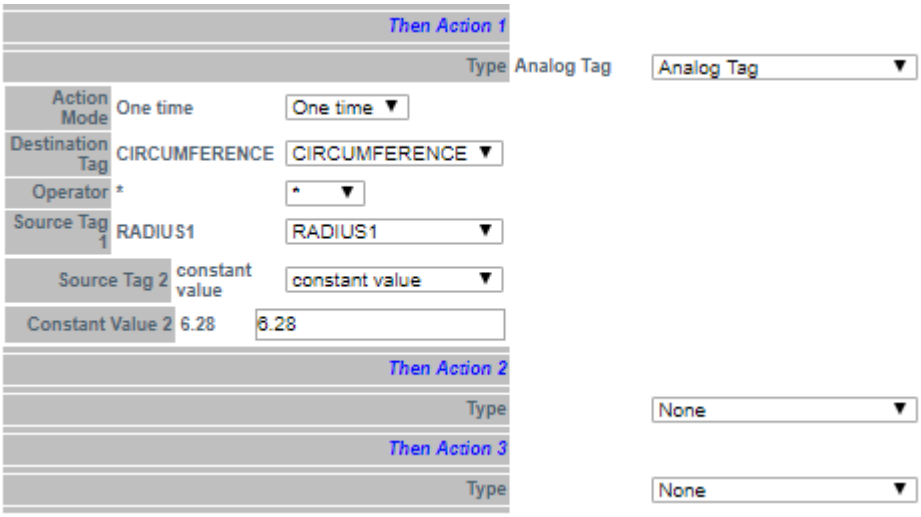
Let us now create the first rule to calculate the circumference using the largest radius between Radius1 and Radius2:
We need the rule to be performed every 1000 ms:



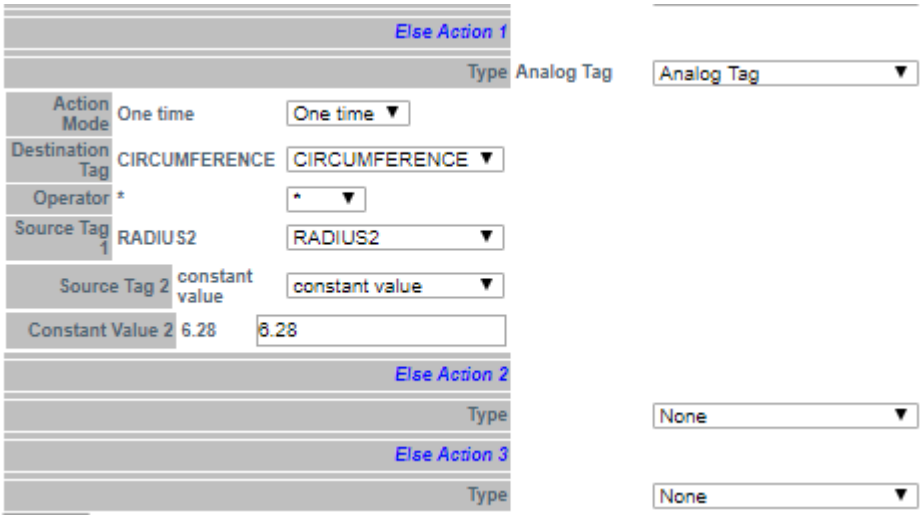
Then we add the "if condition" to determine which is the larger radius between the two provided (we only need 1 if condition):



So, if the condition is true then Radius1> Radius2 we must then calculate the circumference with Radius1: Circumference = Radius1 * 6.28:



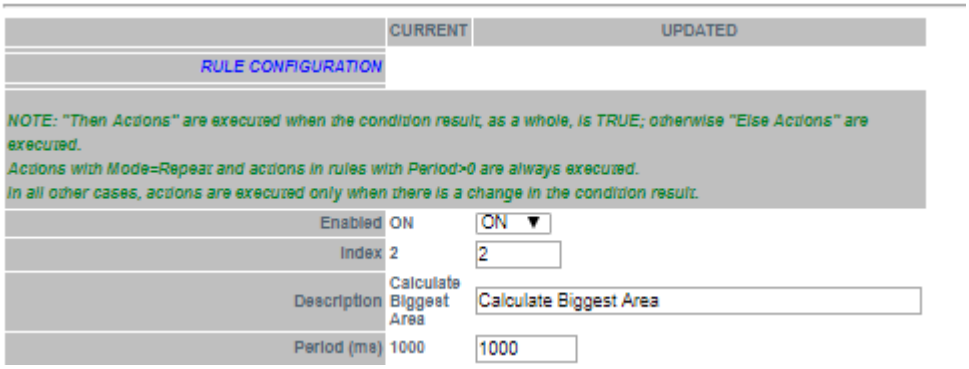
Otherwise, Radius 1 < Radius 2 then we must calculate the circumference with Radius 2 (Circumference = Radius2 * 6.28):



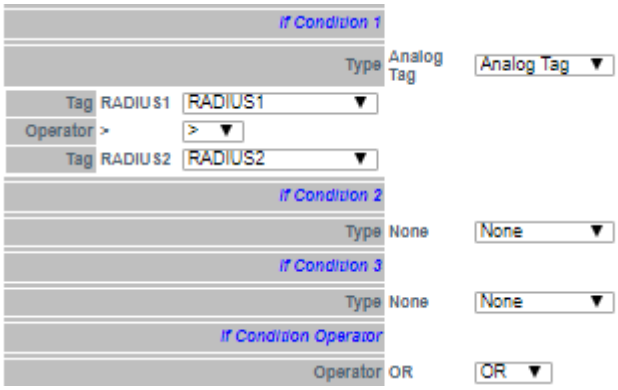
Now click "APPLY" to save the first rule:

#	Enabled	Index	Description	Period (ms)	If condition 1		If condition 2		If condition 3	Then action 1	Then action 2	Then action 3	Else action 1	Else action 2	Else action 3	Condition Status	Breakpoint
1	ON	1	Calculate Biggest Circumference	1000	RADIUS1 > RADIUS2	OR	---	OR	---	CIRCUMFERENCE = RADIUS1 * 6.28	---	---	CIRCUMFERENCE = RADIUS2 * 6.28	---	---	FALSE	---

In the same way we create the Second Rule to calculate the Area with the largest radius:
This rule must also be performed every 1000ms:

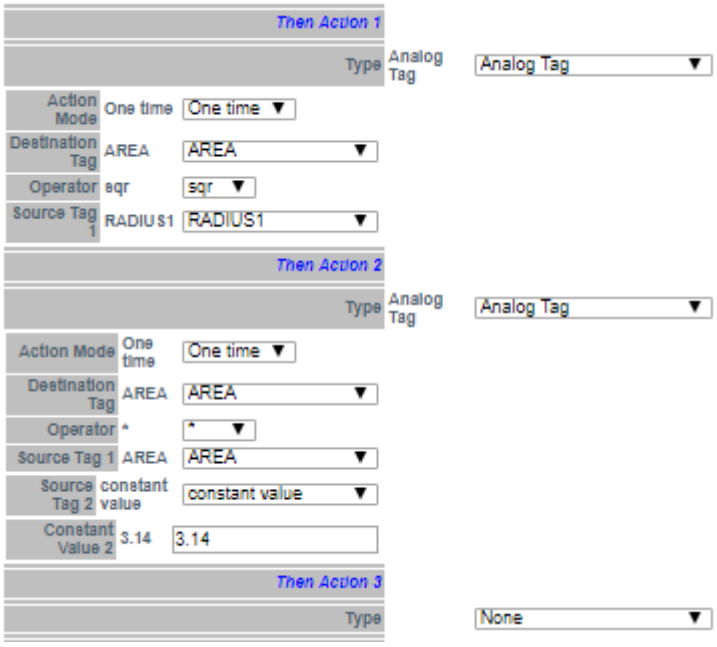


The "if condition" is the same as the first rule:

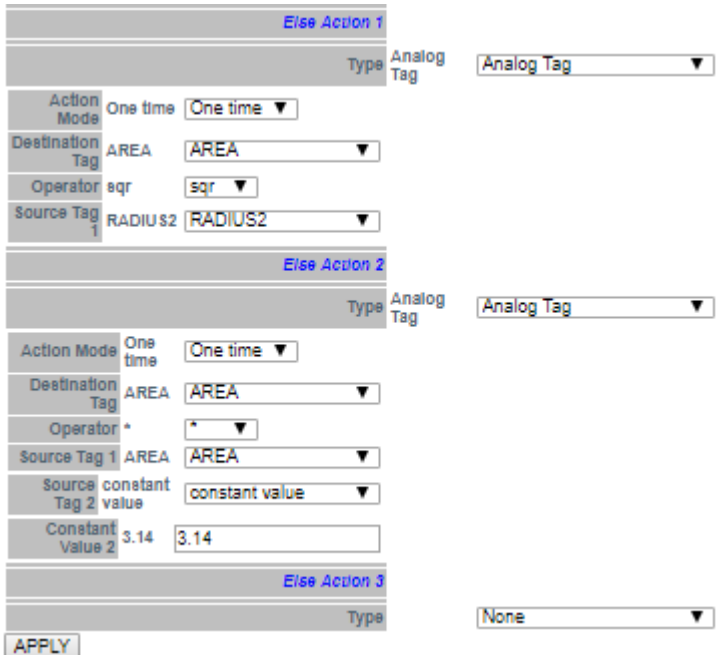


Now we have to calculate the AREA using the following calculation:
AREA = ([RADIUS] ^ 2) * 3.14
We have to break the formula in two phases:
In the first phase we calculate:
AREA = (RADIUS1) ^ 2
And in the second:
AREA = AREA * 3.14

So in our rule if RADIUS1> RADIUS2 we calculate AREA with RADIUS1 using the square function (sqr):
AREA = sqr (RADIUS1)
And then
AREA = AREA * 3.14



So if RADIUS1 <RADIUS2 we calculate AREA with RADIUS2:

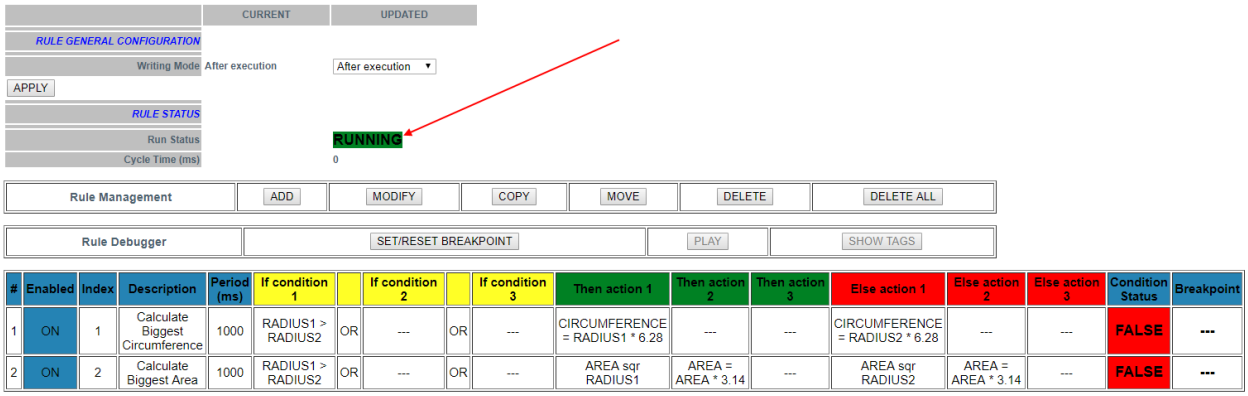


Now click on "APPLY" to save the second rule as well:

#	Enabled	Index	Description	Period (ms)	If condition 1		If condition 2		If condition 3	Then action 1	Then action 2	Then action 3	Else action 1	Else action 2	Else action 3	Condition Status	Breakpoint
1	ON	1	Calculate Biggest Circumference	1000	RADIUS1 > RADIUS2	OR	---	OR	---	CIRCUMFERENCE = RADIUS1 * 6.28	---	---	CIRCUMFERENCE = RADIUS2 * 6.28	---	---	FALSE	---
2	ON	2	Calculate Biggest Area	1000	RADIUS1 > RADIUS2	OR	---	OR	---	AREA sqr RADIUS1	AREA = AREA * 3.14	---	AREA sqr RADIUS2	AREA = AREA * 3.14	---	FALSE	---

Now we can test how our programme works:

When a rule is added, the rule starts automatically (RUNNING):



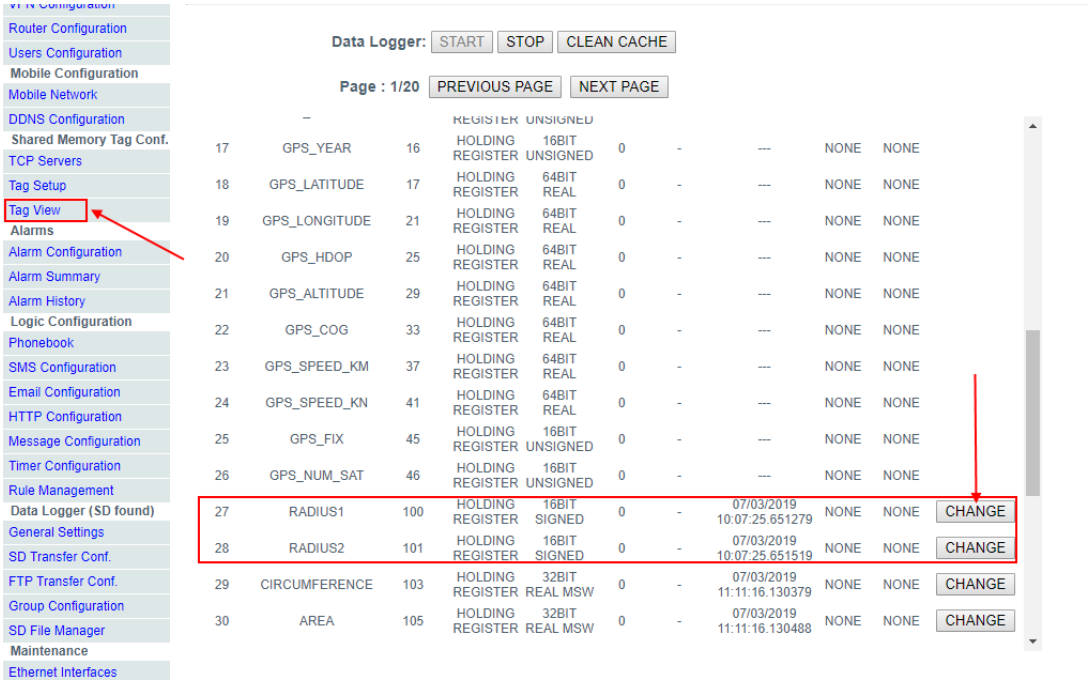
	CURRENT	UPDATED
RULE GENERAL CONFIGURATION		
Writing Mode	After execution	After execution
APPLY		
RULE STATUS		
Run Status		RUNNING
Cycle Time (ms)		0

Rule Management	ADD	MODIFY	COPY	MOVE	DELETE	DELETE ALL

Rule Debugger	SET/RESET BREAKPOINT	PLAY	SHOW TAGS

#	Enabled	Index	Description	Period (ms)	If condition 1		If condition 2		If condition 3	Then action 1	Then action 2	Then action 3	Else action 1	Else action 2	Else action 3	Condition Status	Breakpoint
1	ON	1	Calculate Biggest Circumference	1000	RADIUS1 > RADIUS2	OR	---	OR	---	CIRCUMFERENCE = RADIUS1 * 6.28	---	---	CIRCUMFERENCE = RADIUS2 * 6.28	---	---	FALSE	---
2	ON	2	Calculate Biggest Area	1000	RADIUS1 > RADIUS2	OR	---	OR	---	AREA sqr RADIUS1	AREA = AREA * 3.14	---	AREA sqr RADIUS2	AREA = AREA * 3.14	---	FALSE	---

To test the program we can write the tags RADIUS1 and RADIUS2 from Modbus RTU / MODBUS TCP-IP (registers 40100-40101 in our example) or using the "Tag View" page:



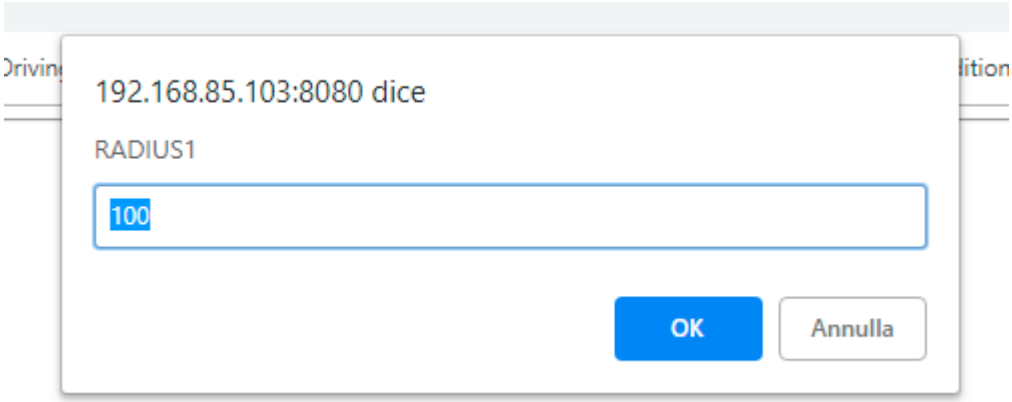
Router Configuration
Users Configuration
Mobile Configuration
Mobile Network
DDNS Configuration
Shared Memory Tag Conf.
TCP Servers
Tag Setup
Tag View
Alarms
Alarm Configuration
Alarm Summary
Alarm History
Logic Configuration
Phonebook
SMS Configuration
Email Configuration
HTTP Configuration
Message Configuration
Timer Configuration
Rule Management
Data Logger (SD found)
General Settings
SD Transfer Conf.
FTP Transfer Conf.
Group Configuration
SD File Manager
Maintenance
Ethernet Interfaces

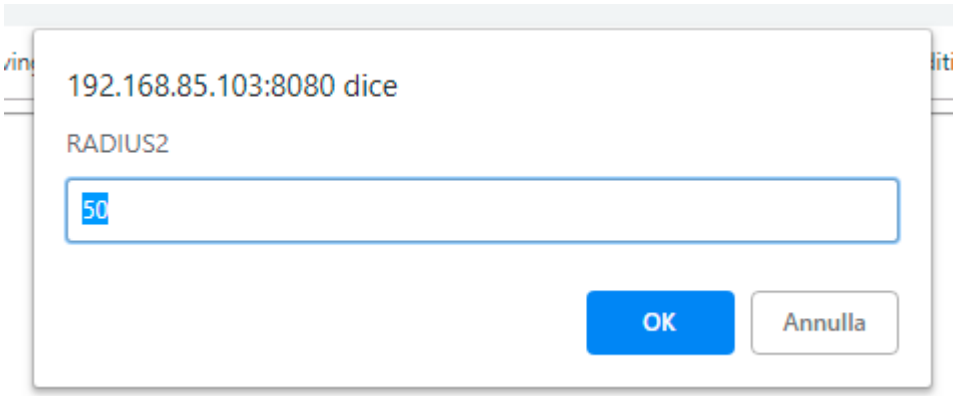
Data Logger: START STOP CLEAN CACHE

Page : 1/20 PREVIOUS PAGE NEXT PAGE

17	GPS_YEAR	16	HOLDING REGISTER	16BIT UNSIGNED	0	-	---	NONE	NONE	
18	GPS_LATITUDE	17	HOLDING REGISTER	64BIT REAL	0	-	---	NONE	NONE	
19	GPS_LONGITUDE	21	HOLDING REGISTER	64BIT REAL	0	-	---	NONE	NONE	
20	GPS_HDOP	25	HOLDING REGISTER	64BIT REAL	0	-	---	NONE	NONE	
21	GPS_ALTITUDE	29	HOLDING REGISTER	64BIT REAL	0	-	---	NONE	NONE	
22	GPS_COG	33	HOLDING REGISTER	64BIT REAL	0	-	---	NONE	NONE	
23	GPS_SPEED_KM	37	HOLDING REGISTER	64BIT REAL	0	-	---	NONE	NONE	
24	GPS_SPEED_KN	41	HOLDING REGISTER	64BIT REAL	0	-	---	NONE	NONE	
25	GPS_FIX	45	HOLDING REGISTER	16BIT UNSIGNED	0	-	---	NONE	NONE	
26	GPS_NUM_SAT	46	HOLDING REGISTER	16BIT UNSIGNED	0	-	---	NONE	NONE	
27	RADIUS1	100	HOLDING REGISTER	16BIT SIGNED	0	-	07/03/2019 10:07:25.651279	NONE	NONE	CHANGE
28	RADIUS2	101	HOLDING REGISTER	16BIT SIGNED	0	-	07/03/2019 10:07:25.651519	NONE	NONE	CHANGE
29	CIRCUMFERENCE	103	HOLDING REGISTER	32BIT REAL MSW	0	-	07/03/2019 11:11:16.130379	NONE	NONE	CHANGE
30	AREA	105	HOLDING REGISTER	32BIT REAL MSW	0	-	07/03/2019 11:11:16.130488	NONE	NONE	CHANGE

Now change RADIUS1 = 100 and RADIUS2 = 50 by clicking the "CHANGE" button:





In the Tag view the CIRCUMFERENCE and AREA calculations are updated:

27	RADIUS1	100	HOLDING REGISTER	16BIT SIGNED	100	-	07/03/2019 11:15:56.934313	NONE	NONE	CHANGE
28	RADIUS2	101	HOLDING REGISTER	16BIT SIGNED	50	-	07/03/2019 11:34:12.465220	NONE	NONE	CHANGE
29	CIRCUMFERENCE	103	HOLDING REGISTER	32BIT REAL MSW	628	-	07/03/2019 11:34:39.634836	NONE	NONE	CHANGE
30	AREA	105	HOLDING REGISTER	32BIT REAL MSW	31400	-	07/03/2019 11:34:39.634973	NONE	NONE	CHANGE

Now we can go to the "Rules Management" page to view the result:

	CURRENT	UPDATED
RULE GENERAL CONFIGURATION		
Writing Mode	After execution	After execution
APPLY		
RULE STATUS		
Run Status		RUNNING
Cycle Time (ms)		0

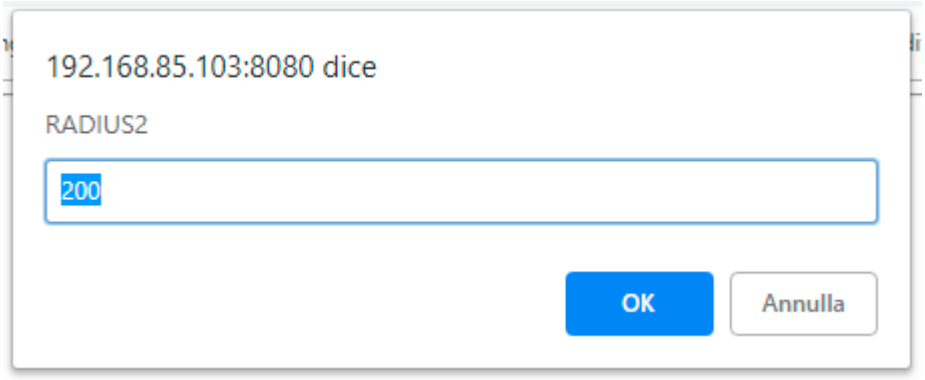
Rule Management	ADD	MODIFY	COPY	MOVE	DELETE	DELETE ALL

Rule Debugger	SET/RESET BREAKPOINT	PLAY	SHOW TAGS

#	Enabled	Index	Description	Period (ms)	If condition 1		If condition 2		If condition 3	Then action 1	Then action 2	Then action 3	Else action 1	Else action 2	Else action 3	Condition Status	Breakpoint
1	ON	1	Calculate Biggest Circumference	1000	RADIUS1 > RADIUS2	OR	---	OR	---	CIRCUMFERENCE = RADIUS1 * 6.28	---	---	CIRCUMFERENCE = RADIUS2 * 6.28	---	---	TRUE	---
2	ON	2	Calculate Biggest Area	1000	RADIUS1 > RADIUS2	OR	---	OR	---	AREA sqr RADIUS1	AREA = AREA * 3.14	---	AREA sqr RADIUS2	AREA = AREA * 3.14	---	TRUE	---

So both conditions if they are TRUE (penultimate column) and then "Then actions" are executed.

Now we change the RADIUS2 value in the tag display pages to 200:



So:

	CURRENT	UPDATED
RULE GENERAL CONFIGURATION		
Writing Mode	After execution	After execution
APPLY		
RULE STATUS		
Run Status		RUNNING
Cycle Time (ms)		0

Rule Management	ADD	MODIFY	COPY	MOVE	DELETE	DELETE ALL

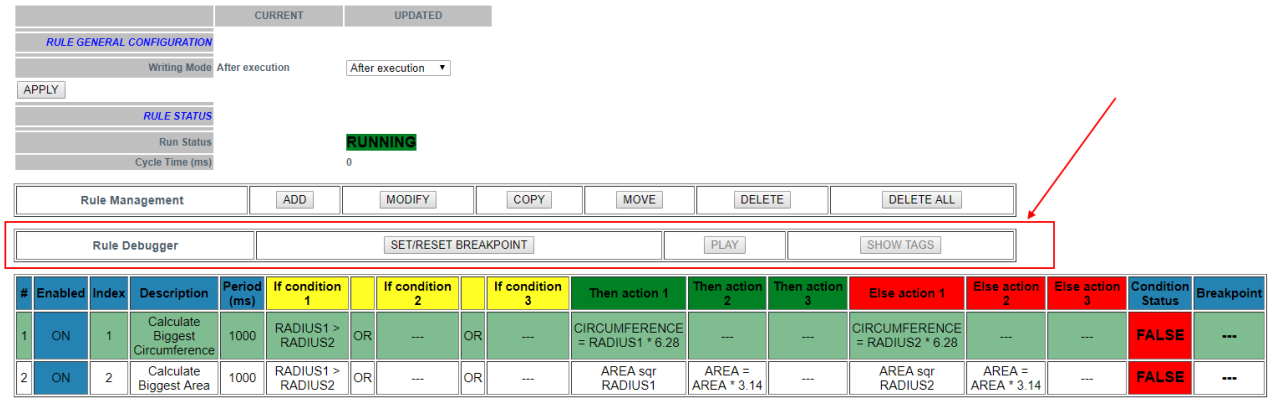
Rule Debugger	SET/RESET BREAKPOINT	PLAY	SHOW TAGS

#	Enabled	Index	Description	Period (ms)	If condition 1		If condition 2		If condition 3	Then action 1	Then action 2	Then action 3	Else action 1	Else action 2	Else action 3	Condition Status	Breakpoint
1	ON	1	Calculate Biggest Circumference	1000	RADIUS1 > RADIUS2	OR	---	OR	---	CIRCUMFERENCE = RADIUS1 * 6.28	---	---	CIRCUMFERENCE = RADIUS2 * 6.28	---	---	FALSE	---
2	ON	2	Calculate Biggest Area	1000	RADIUS1 > RADIUS2	OR	---	OR	---	AREA sqr RADIUS1	AREA = AREA * 3.14	---	AREA sqr RADIUS2	AREA = AREA * 3.14	---	FALSE	---

Now the condition status of the 2 rules is false because RADIUS1 <RADIUS2, so the "Else Actions" are executed

It is also possible to debug the program using the internal rule debugger.
With the internal debugger it is possible:
-Insert a breakpoint before the execution of a rule
-View the tag values before / after the execution of a rule

To add a breakpoint and stop the program flow select the rule and then press "SET / RESET BREAKPOINT":

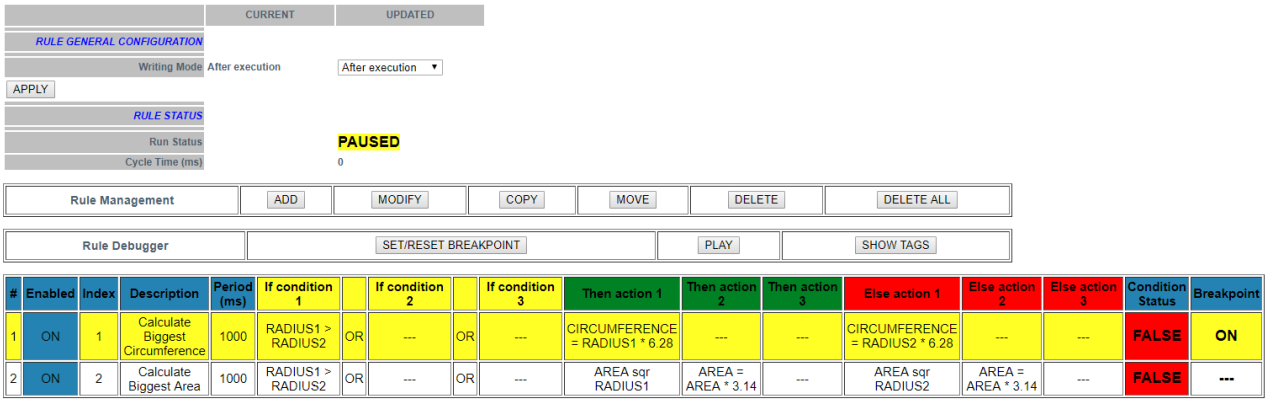


	CURRENT	UPDATED
RULE GENERAL CONFIGURATION		
Writing Mode	After execution	After execution
APPLY		
RULE STATUS		
Run Status		RUNNING
Cycle Time (ms)		0

Rule Management	ADD	MODIFY	COPY	MOVE	DELETE	DELETE ALL

Rule Debugger	SET/RESET BREAKPOINT	PLAY	SHOW TAGS

#	Enabled	Index	Description	Period (ms)	If condition 1		If condition 2		If condition 3	Then action 1	Then action 2	Then action 3	Else action 1	Else action 2	Else action 3	Condition Status	Breakpoint
1	ON	1	Calculate Biggest Circumference	1000	RADIUS1 > RADIUS2	OR	---	OR	---	CIRCUMFERENCE = RADIUS1 * 6.28	---	---	CIRCUMFERENCE = RADIUS2 * 6.28	---	---	FALSE	---
2	ON	2	Calculate Biggest Area	1000	RADIUS1 > RADIUS2	OR	---	OR	---	AREA sqr RADIUS1	AREA = AREA * 3.14	---	AREA sqr RADIUS2	AREA = AREA * 3.14	---	FALSE	---



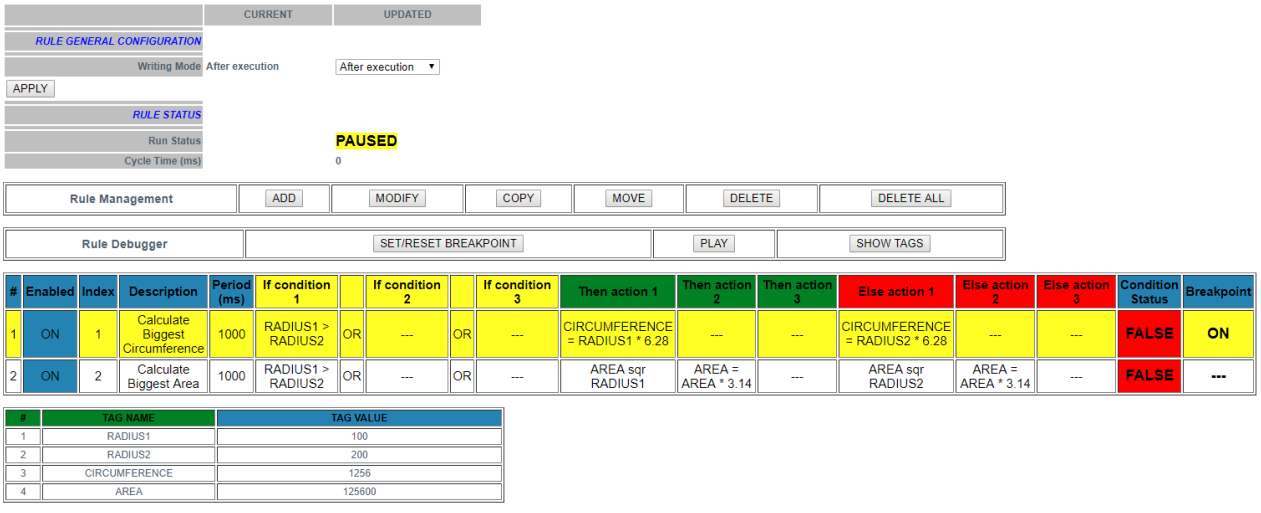
	CURRENT	UPDATED
RULE GENERAL CONFIGURATION		
Writing Mode	After execution	After execution
APPLY		
RULE STATUS		
Run Status		PAUSED
Cycle Time (ms)		0

Rule Management	ADD	MODIFY	COPY	MOVE	DELETE	DELETE ALL

Rule Debugger	SET/RESET BREAKPOINT	PLAY	SHOW TAGS

#	Enabled	Index	Description	Period (ms)	If condition 1		If condition 2		If condition 3	Then action 1	Then action 2	Then action 3	Else action 1	Else action 2	Else action 3	Condition Status	Breakpoint
1	ON	1	Calculate Biggest Circumference	1000	RADIUS1 > RADIUS2	OR	---	OR	---	CIRCUMFERENCE = RADIUS1 * 6.28	---	---	CIRCUMFERENCE = RADIUS2 * 6.28	---	---	FALSE	ON
2	ON	2	Calculate Biggest Area	1000	RADIUS1 > RADIUS2	OR	---	OR	---	AREA sqr RADIUS1	AREA = AREA * 3.14	---	AREA sqr RADIUS2	AREA = AREA * 3.14	---	FALSE	---

The rule turns yellow and the rule status changes to "Paused". Note that the breakpoint is before the rule execution.
Clicking "Show tags" displays the tag values before the selected rule is executed.



	CURRENT	UPDATED
RULE GENERAL CONFIGURATION		
Writing Mode	After execution	After execution
APPLY		
RULE STATUS		
Run Status		PAUSED
Cycle Time (ms)		0

Rule Management	ADD	MODIFY	COPY	MOVE	DELETE	DELETE ALL

Rule Debugger	SET/RESET BREAKPOINT	PLAY	SHOW TAGS

#	Enabled	Index	Description	Period (ms)	If condition 1		If condition 2		If condition 3	Then action 1	Then action 2	Then action 3	Else action 1	Else action 2	Else action 3	Condition Status	Breakpoint
1	ON	1	Calculate Biggest Circumference	1000	RADIUS1 > RADIUS2	OR	---	OR	---	CIRCUMFERENCE = RADIUS1 * 6.28	---	---	CIRCUMFERENCE = RADIUS2 * 6.28	---	---	FALSE	ON
2	ON	2	Calculate Biggest Area	1000	RADIUS1 > RADIUS2	OR	---	OR	---	AREA sqr RADIUS1	AREA = AREA * 3.14	---	AREA sqr RADIUS2	AREA = AREA * 3.14	---	FALSE	---

#	TAG NAME	TAG VALUE
1	RADIUS1	100
2	RADIUS2	200
3	CIRCUMFERENCE	1256
4	AREA	125600

Now you can move the breakpoint to the next rule, then select the next rule and press the "SET / RESET BREAKPOINT" button:
Pressing the "PLAY" button will stop the execution before the next rule is executed:

	CURRENT	UPDATED
RULE GENERAL CONFIGURATION		
Writing Mode	After execution	After execution ▾
APPLY		
RULE STATUS		
Run Status		PAUSED
Cycle Time (ms)		0

Rule Management	ADD	MODIFY	COPY	MOVE	DELETE	DELETE ALL

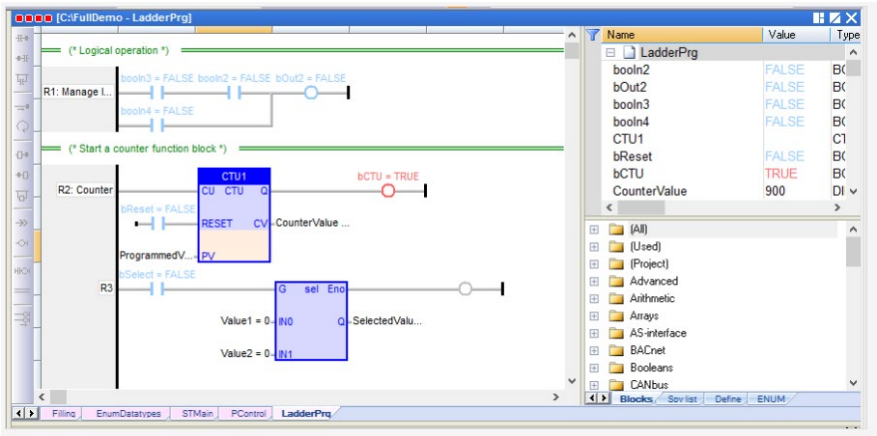
Rule Debugger	SET/RESET BREAKPOINT	PLAY	SHOW TAGS

#	Enabled	Index	Description	Period (ms)	If condition 1		If condition 2		If condition 3	Then action 1	Then action 2	Then action 3	Else action 1	Else action 2	Else action 3	Condition Status	Breakpoint
1	ON	1	Calculate Biggest Circumference	1000	RADIUS1 > RADIUS2	OR	---	OR	---	CIRCUMFERENCE = RADIUS1 * 6.28	---	---	CIRCUMFERENCE = RADIUS2 * 6.28	---	---	FALSE	---
2	ON	2	Calculate Biggest Area	1000	RADIUS1 > RADIUS2	OR	---	OR	---	AREA sqr RADIUS1	AREA = AREA * 3.14	---	AREA sqr RADIUS2	AREA = AREA * 3.14	---	FALSE	ON

#	TAG NAME	TAG VALUE
1	RADIUS1	100
2	RADIUS2	200
3	CIRCUMFERENCE	1256
4	AREA	125600

13. THE STRATON PLC

The Straton PLC provides full support for the IEC 61131-3 PLC standard; an integrated development environment (IDE) is available for Windows™ PCs.
The Straton IDE includes several tools such as: a fieldbus configuration tool, an analog signal editor and program editors compliant with the five languages of the IEC 61131-3 standard: Sequential Function Chart (SFC), Function Block Diagram (FBD), Ladder Diagram (LD), Structured Text (ST), Instruction List (IL).
With Straton IDE, it's easy to write, download and debug the IEC 61131-3 code.



Depending on the model, the device may or may not have the PLC activated by default. By contacting Seneca it is always possible to activate the PLC by entering an activation code.
The PLC directly manages the following protocols: Modbus RTU, Modbus TCP-IP, MQTT, OPC-UA Client, MeterBus (MBUS), S7 Client, SNMP.
To use the MeterBus protocol, it is necessary to purchase the optional Z-MBUS device

For more information, refer to the STRATON PLC manual.

https://straton-plc.com/en/downloads/

To allow the PLC developer to easily create Straton applications for Seneca gateways, the following libraries are available:

• a Function Block (FB) and Functions library, which provides some frequently used functions, in particular related to communication and data transfer activities, compiled in the CPU firmware; the direct use of these FBs and functions is aimed at expert PLC developers (a detailed description of the FBs and functions is given in the relevant chapter of this manual);
• a "Profile" library, which allows access to CPU I/OS via "profile" variables
• a "User Defined Function Block" (UDFB) library, in ST language, which simplifies the use of the aforementioned FBs, providing simpler and "higher level" access to their functions.

An installation program called "Seneca Straton Package" is available which automatically installs the Seneca libraries and templates. The installation program also includes Straton IDE and other tools.
The installation program is available at the following link:

http://www.seneca.it/products/seneca-straton-package

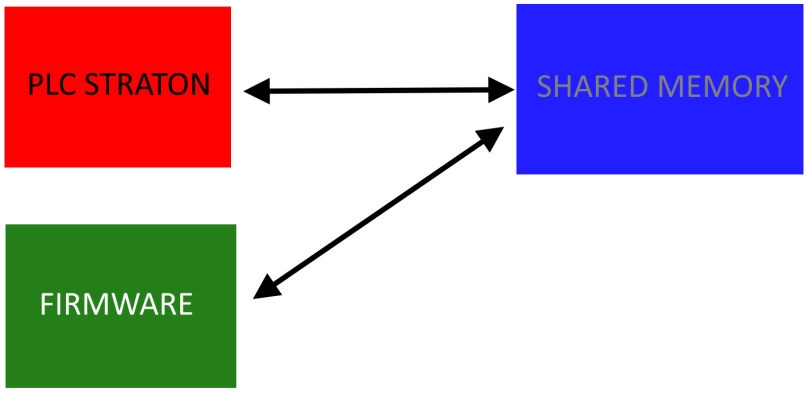
If, for some reason, the installation program cannot be run, the above libraries and templates can also be installed manually.

The Straton PLC in Seneca gateways can operate in the following modes:

"NONE" MODE
The Straton PLC is disabled (default mode for SSD, Z-PASS1-RT, Z-PASS2-RT-4G, R-PASS models)

"LEGACY (STAND-ALONE)" MODE
The Straton PLC operates in mode compatible with firmware versions prior to 3000, i.e. the communication protocols are managed only by the PLC (default mode for SSD-S, Z-TWS4-RT-S, Z-PASS2-RT-4G-S, R-PASS-S models).

"SHARED" MODE
The Straton PLC operates in shared mode, i.e. the Straton PLC and the firmware communicate with each other via a shared memory on OPC-UA protocol.



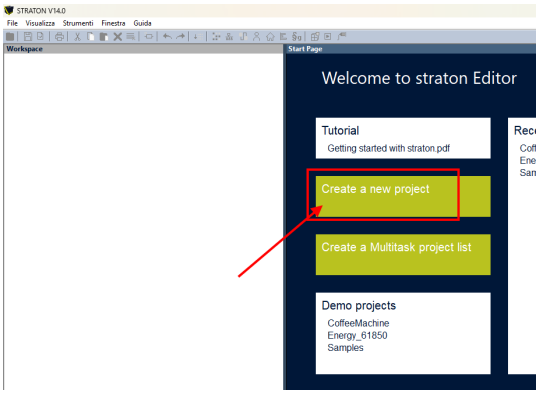
In this mode it is possible to activate the data logger, alarms, display / virtual display and communication protocols already present in the firmware and to read and write the TAGs directly from the PLC.

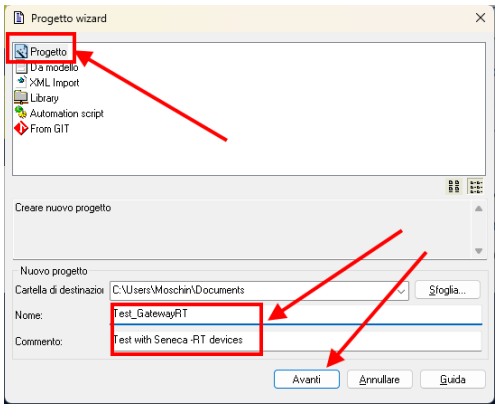
13.1. IMPORTING TAGS INTO THE PLC (PLC MODE = SHARED)

In this chapter we will see how to:

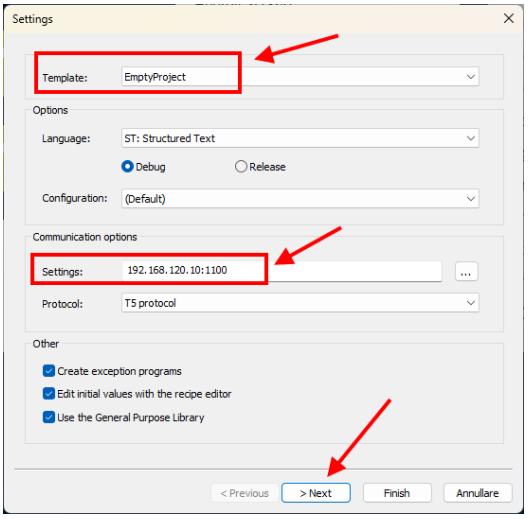
- Create a new Straton PLC project
- Import Tags written by the Gateway firmware on the PLC to be able to read them
- Create Tags written by Straton and be able to read them in the gateway firmware (for example to be displayed on the display synoptics / virtual display).

Run the Straton IDE and create a new project:

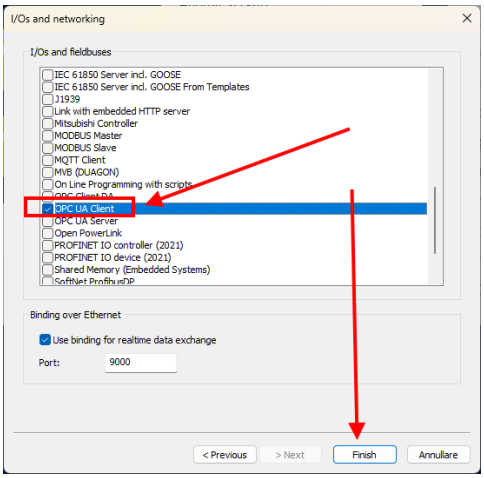




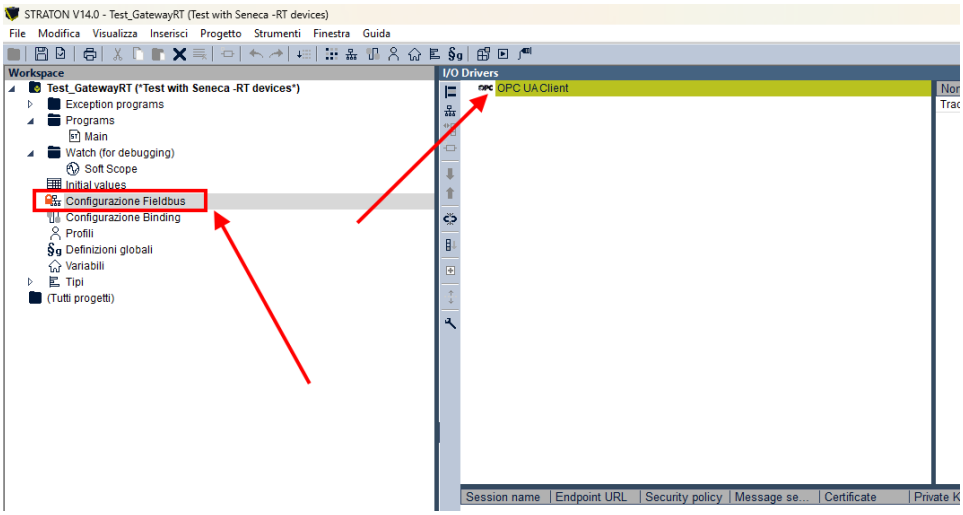
Let's start from an empty project and insert the gateway IP address (in the example 192.168.120.10):



OPC-UA is used as the internal fieldbus for tag exchange, so let's select it and press finish:

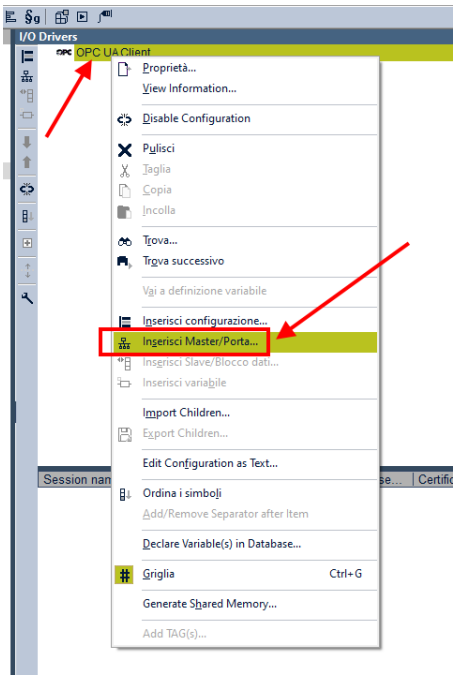


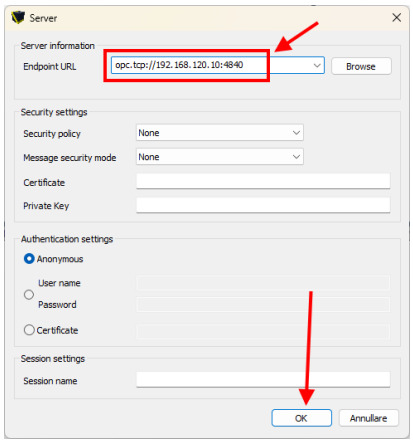
At this point in the fieldbus configuration we will have the OPC-UA client in the IDE:



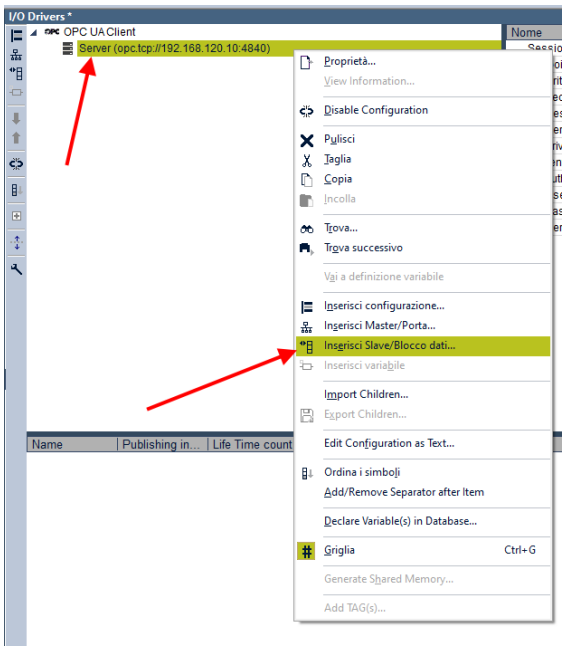
Now we will import the TAGs defined in the gateway to be imported into Straton.
The import is done simply by performing a TAG scan.

First we insert the OPC-UA master and as the server address the Gateway address (in our case 192.168.120.10):

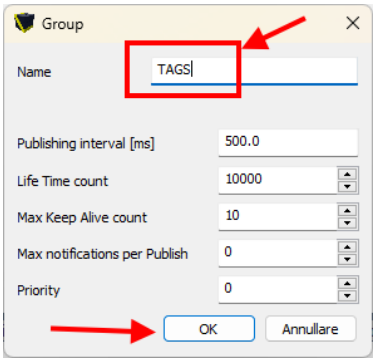




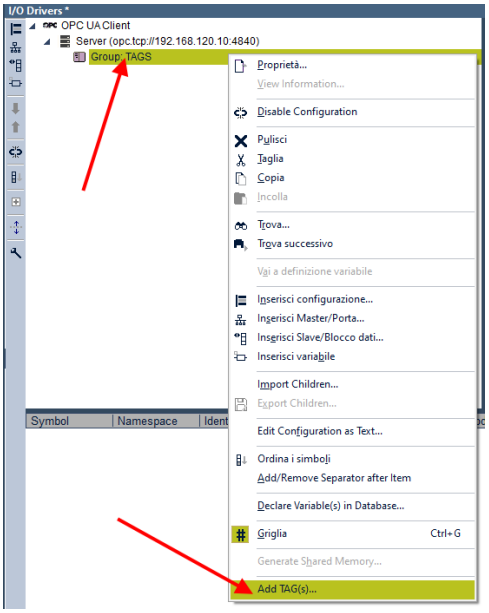
At this point we prepare the data block where the TAGs will be inserted:



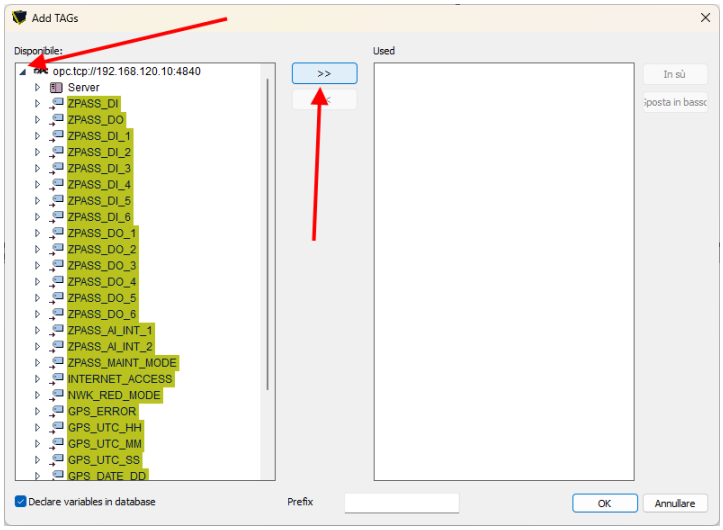
And we call the data group with a name of our choice, in our case TAGS:

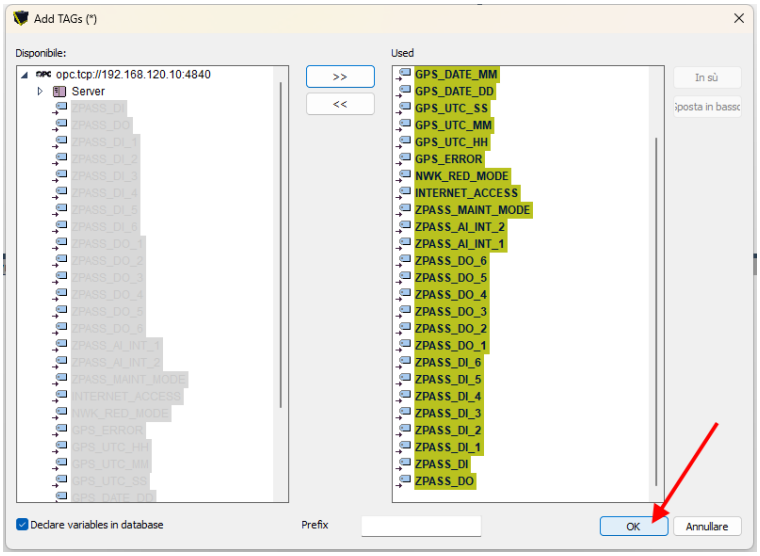


Now we are ready to import the TAGs by clicking on Add TAGS:

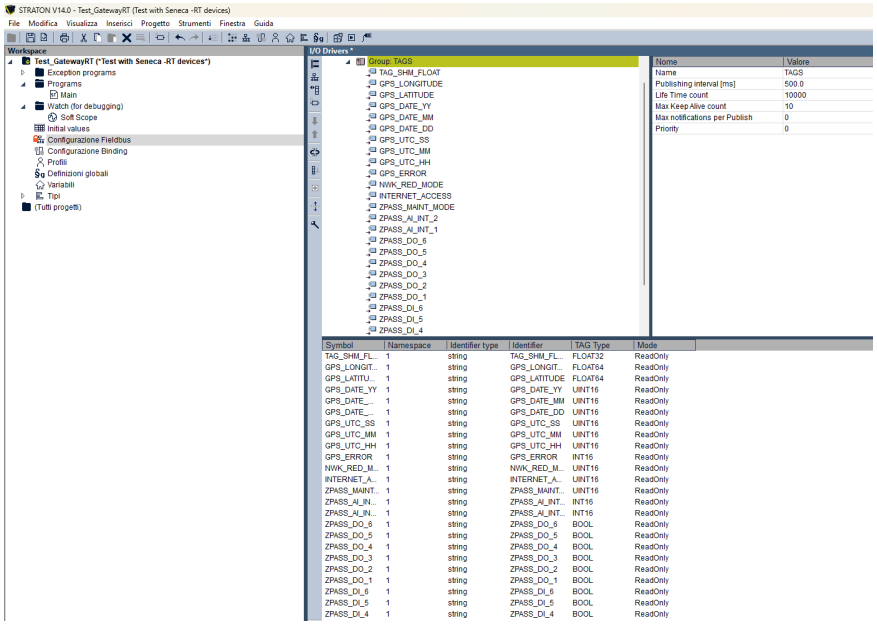


The operation lists all the tags defined by the gateway (including the embedded type Tags).
To import them into Straton press the >> icon:

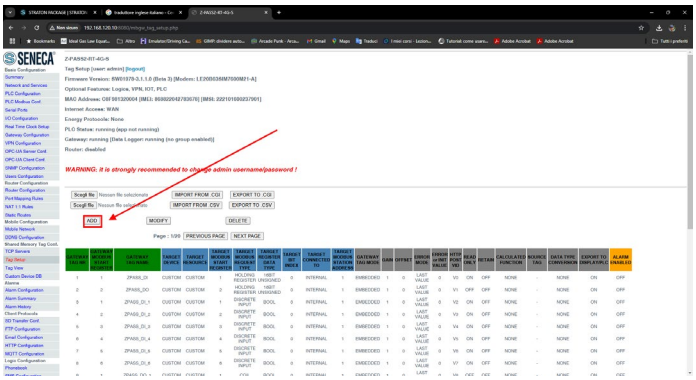


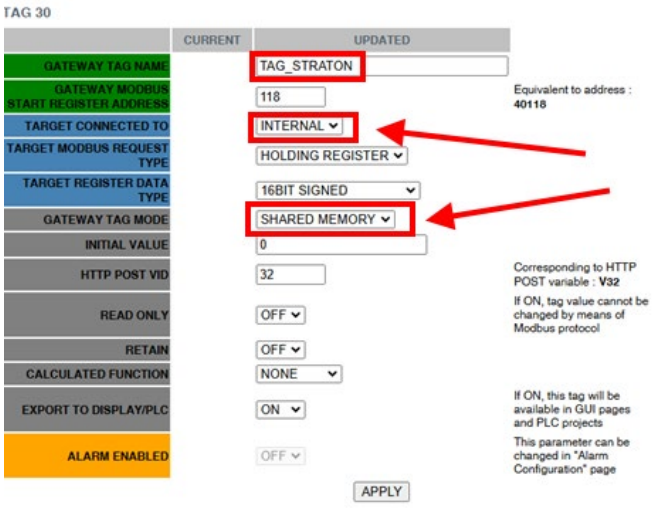


At this point the TAGs are imported into the PLC, note how all are set by default as ReadOnly:



If we want to create a tag that can be written by Straton and displayed for example on the physical or virtual display, we must first create a TAG of the “internal” type in “shared memory” and enable the export to Display/PLC:

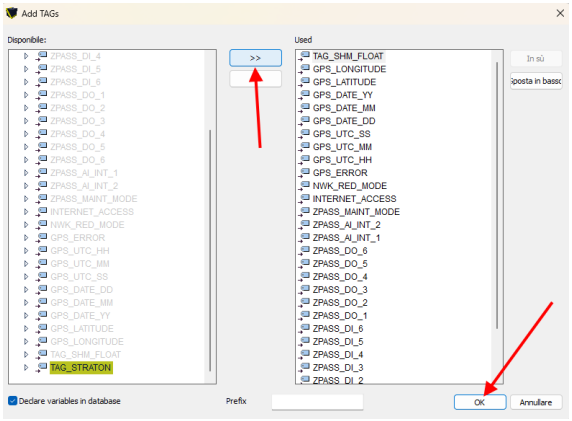




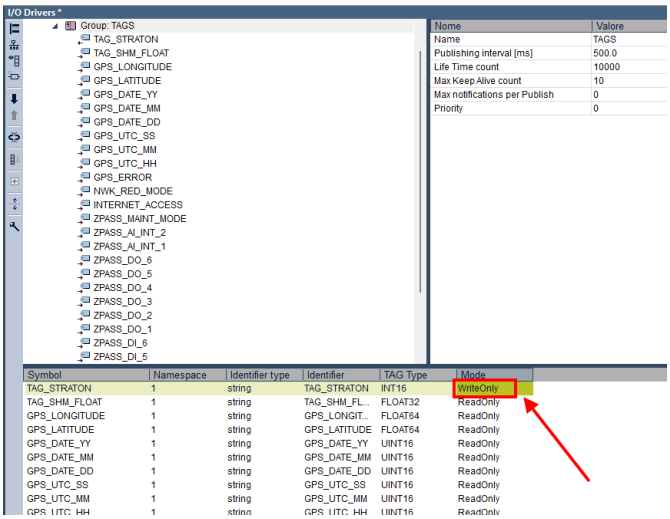
The new tag appears on the Tag view page.

30	118	TAG_STRATON	-	-	-	HOLDING REGISTER	16BIT SIGNED	0	INTERNAL	-	SHARED-MEMORY	1	0	ERROR VALUE	0	V32	OFF	OFF	NONE	-	NONE	ON	OFF

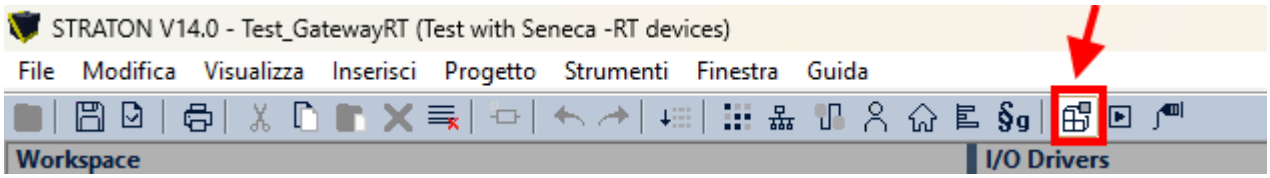
Now we return to Straton and import the new tag with the “Add Tags” option:



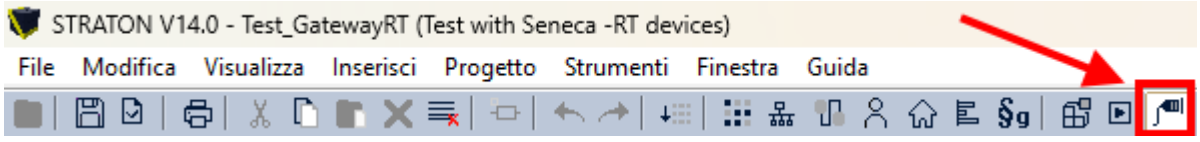
Since we want to write the Tag from Straton, we set it to write:



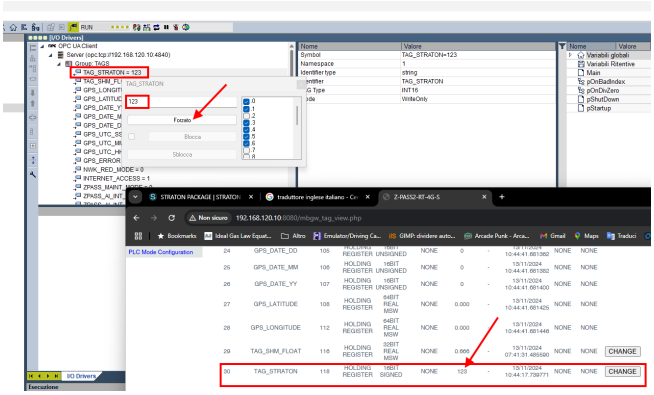
We compile:



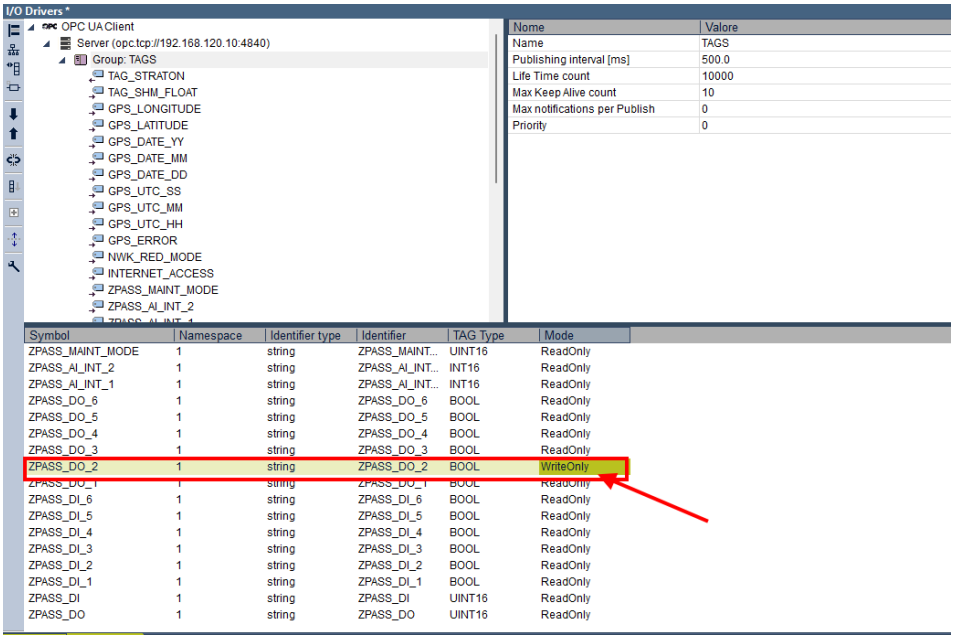
And send the Project to the target:



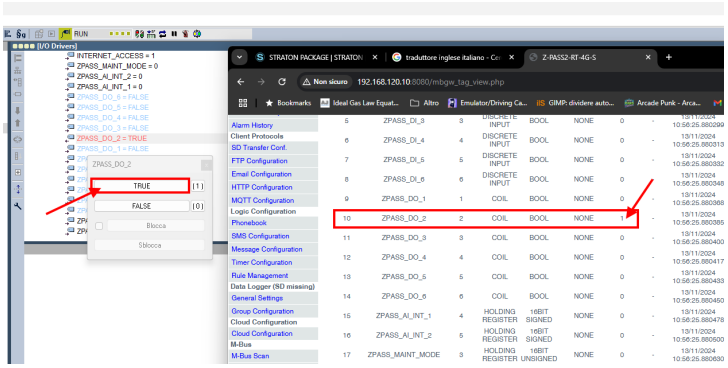
Now if the TAG is written by Straton, we see the effect on the tag view of the web server:



Now we write an embedded I/O from Straton, we change the TAG for example of DO2 in WriteOnly:



We compile and send the project.
If we force the TAG from Straton, we see the effect on the web page (and on the device LED):



Beware that this TAG is write-only on Straton, so it is not possible to write it for example from logical rules.

14. SCRIPT EXECUTION IN LOGICAL RULES

Devices allow you to execute scripts as Then/Else actions in logical rules.

ATTENTION!

Scripts are a very powerful tool and as such can modify the proper operation of the device. It is the user's responsibility to verify that this does not happen. It is also necessary to verify that the script does not allow modification of the cybersecurity of the device, for example by opening sockets that are not foreseen.

14.1. READING AND WRITING A TAG FROM A SCRIPT

Reading and writing a tag from a script are performed using the commands:
“tag_read” and “tag_write”.

14.1.1. TAG_READ

The tag_read command allows you to read the value of a tag.
The syntax is as follows:

tag_read <tag_name>

returns:

<res>;<tag_value>;<is_valid>

Where:
<res>
It may be:

0: success
-1: invalid argument
-2: operation failed

<tag_value>
It is the value of the tag in string format

<is_valid>
0: the value of the tag is in fail
1: the value of the tag is valid

example:

tag_read TAG_SHM_CNT

returns:

0;172;1

It means that the tag exists, the tag value is 172 and the tag is not failing

14.1.2.TAG_WRITE

Using the tag_write command you can write a tag.
The syntax is as follows:

tag_write <tag_name> <tag_value>

returns:

<res>
0: success
-1: invalid arguments
-2: operation failed

Example:

tag_write TAG_SHM_CNT 173

returns
0

It means that the tag exists, the write operation was successful.

14.2. EXAMPLE OF A PYTHON SCRIPT

The following script reads the value of the tag “TAG_SHM_CNT” increases it by 1 and rewrites the new value in the same tag. For further information, see link:
https://www.w3schools.com/python/python_intro.asp

```
from subprocess import run

tag_read_prog  = "/disk/bin/tag_read"
```

```
tag_write_prog = "/disk/bin/tag_write"

tag_name="TAG_SHM_CNT"

read_cmd = tag_read_prog + " " + tag_name
data = run(read_cmd, capture_output=True, shell=True, text=True) #read the tag
out_str = data.stdout
res_str = out_str.rstrip() # strip strailing newline character
res = res_str.split(";")
if res[0] == "0":
        print("tag_read success !")
        print("tag_value: " + res[1])
        print("tag_valid: " + res[2])
        val = int(res[1])
        read_ok = True
else:
        print("tag_read failure !")
        read_ok = False

if read_ok == True:
        new_val = val + 1 # increment by 1
        write_cmd = tag_write_prog + " " + tag_name + " " + str(new_val)
        data = run(write_cmd, capture_output=True, shell=True, text=True) #write the tag
        out_str = data.stdout
        res = out_str.strip() # strip strailing newline character
        if res == "0":
                print("tag_write success !")
        else:
                print("tag_write failure !")
```

14.3. PYTHON MODULES INSTALLED

```
__future__      _threading_local    grp             secrets
_abc            _tracemalloc        gzip            select
_ast            _uuid               hashlib         selectors
_asyncio        _warnings           heapq           shelve
_bisect         _weakref            hmac            shlex
_blake2         _weakrefset         html            shutil
_bootlocale     _xxtestfuzz         http            signal
_bz2            abc                 idlelib         site
```

_codecs aifc imaplib smtpd
_codecs_cn antigravity imghdr smtplib
_codecs_hk argparse imp sndhdr
_codecs_iso2022 array importlib socket
_codecs_jp ast inspect socketserver
_codecs_kr asynchat io spwd
_codecs_tw asyncio ipaddress sqlite3
_collections asyncore itertools sre_compile
_collections_abc atexit json sre_constants
_compat_pickle audioop keyword sre_parse
_compression base64 ldb ssl
_contextvars bdb lib2to3 stat
_crypt binascii linecache statistics
_csv binhex locale string
_ctypes bisect logging stringprep
_ctypes_test builtins lzma struct
_curses bz2 macpath subprocess
_curses_panel cProfile mailbox sunau
_datetime calendar mailcap symbol
_dbm cgi marshal symtable
_decimal cgitb math sys
_dummy_thread chunk mimetypes sysconfig
_elementtree cmath mmap syslog
_functools cmd modulefinder tabnanny
_hashlib code multiprocessing talloc
_heapq codecs netrc tarfile
_imp codeop nis tdb
_io collections nntplib telnetlib
_json colorsys ntpath tempfile
_ldb_text compileall nturl2path termios
_locale concurrent numbers textwrap
_lsprof configparser opcode this
_lzma contextlib operator threading
_markupbase contextvars optparse time
_md5 copy os timeit
_multibytecodec copyreg ossaudiodev tkinter
_multiprocessing crypt parser token
_opcode csv pathlib tokenize
_operator ctypes pdb trace
_osx_support curses pickle traceback
_pickle dataclasses pickletools tracemalloc
_posixsubprocess datetime pipes tty

_py_abc dbm pkgutil turtle
_pydecimal decimal platform turtledemo
_pyio difflib plistlib types
_queue dis poplib typing
_random distutils posix unicodedata
_sha1 doctest posixpath unittest
_sha256 dummy_threading pprint urllib
_sha3 email profile uu
_sha512 encodings pstats uuid
_signal ensurepip pty venv
_sitebuiltins enum pwd warnings
_socket errno py_compile wave
_sqlite3 faulthandler pyclbr weakref
_sre fcntl pydoc webbrowser
_ssl filecmp pydoc_data wsgiref
_stat fileinput pyexpat xdrlib
_string fnmatch queue xml
_strptime formatter quopri xmlrpc
_struct fractions random xxlimited
_symtable ftplib re xxsubtype
_sysconfigdata_m_linux_arm-linux-gnueabi functools readline
zipapp
_tdb_text gc reprlib zipfile
_testbuffer genericpath resource zipimport
_testcapi getopt rlcompleter zlib
_testimportmultiple getpass runpy
_testmultiphase gettext samba
_thread glob sched

15. IEC 61850 E 6070-5 PROTOCOLS FOR PLC STRATON

In the devices it is possible to activate (together with the Straton PLC) other additional protocols; it is possible to activate:

IEC61850 Server
IEC61850 Client
IEC60870-5-104 Server
IEC60870-5-104 Client
IEC60870-5-101 Master
IEC60850-5-101 Slave

IEC 61850 is a standard for the design of automation systems for electrical substations. It is part of the International Electrotechnical Commission.

IEC 60870 Part 5 is one of the IEC 60870 standards that define systems used for telecontrol (supervisory control and data acquisition) in electrical engineering and power system automation applications. Part 5 provides a communication profile for sending basic telecontrol messages between two systems, which uses permanent data circuits directly connected between the systems.
IEC 60870-5-104 (aka IEC 104 or Protocol 104) has a TCP/IP-based data transmission mode, IEC 60870-5-101 (aka IEC 101 or Protocol 101) has a serial-based data transmission mode.

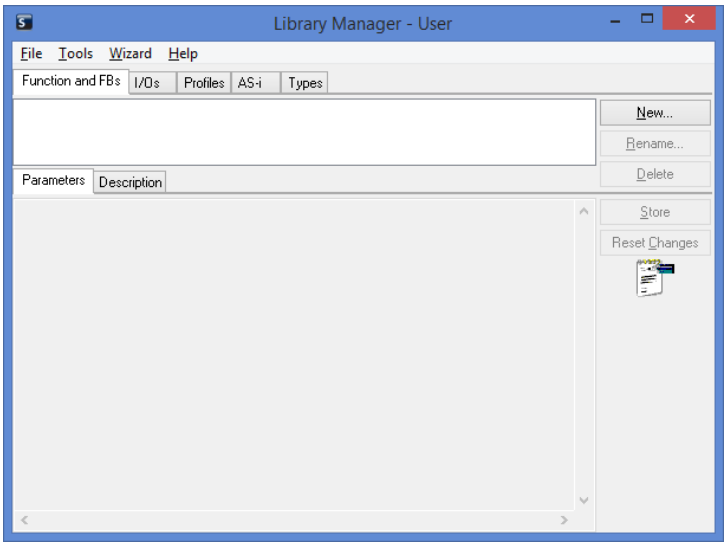
For more information, refer to the STRATON PLC manual.

https://straton-plc.com/en/downloads/

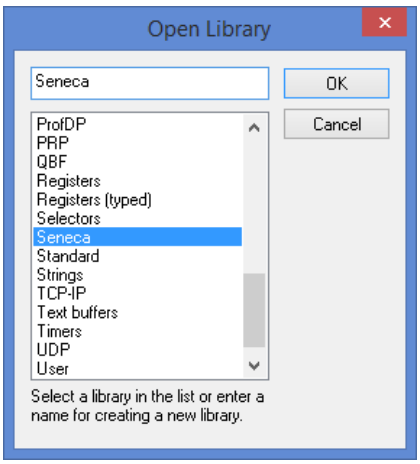
16. MANUAL INSTALLATION OF LIBRARIES IN STRATON

The following steps are required to integrate the libraries into the Straton IDE in case you do not want to use the Straton package software.

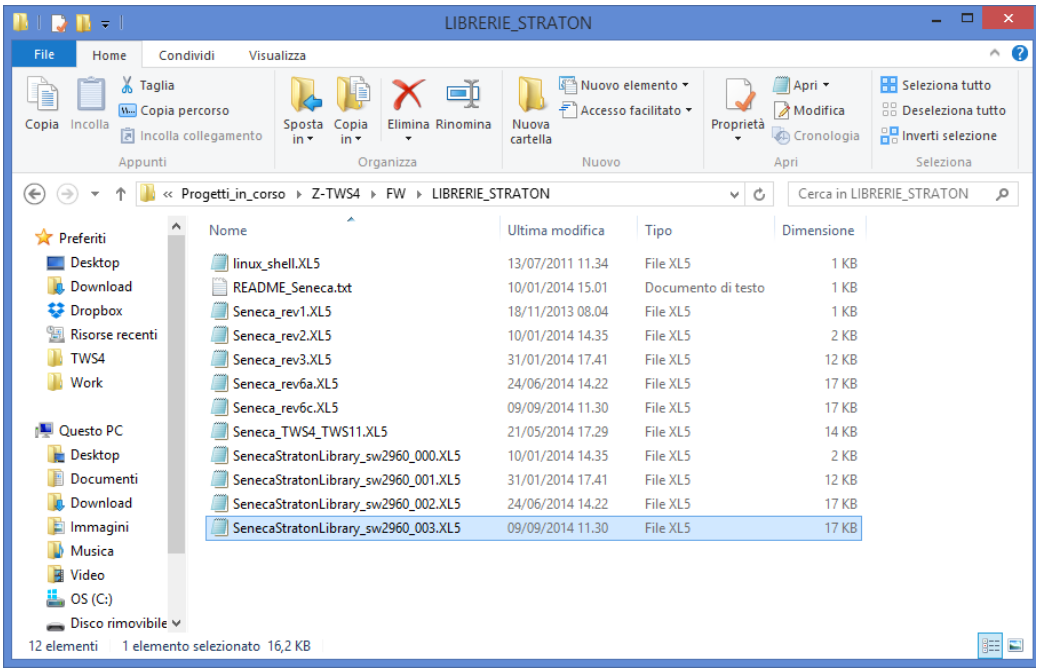
First, we need to add the Seneca FB library (SenecaStratonLibrary.XL5 file) to the IDE, using the "Library Manager" tool:

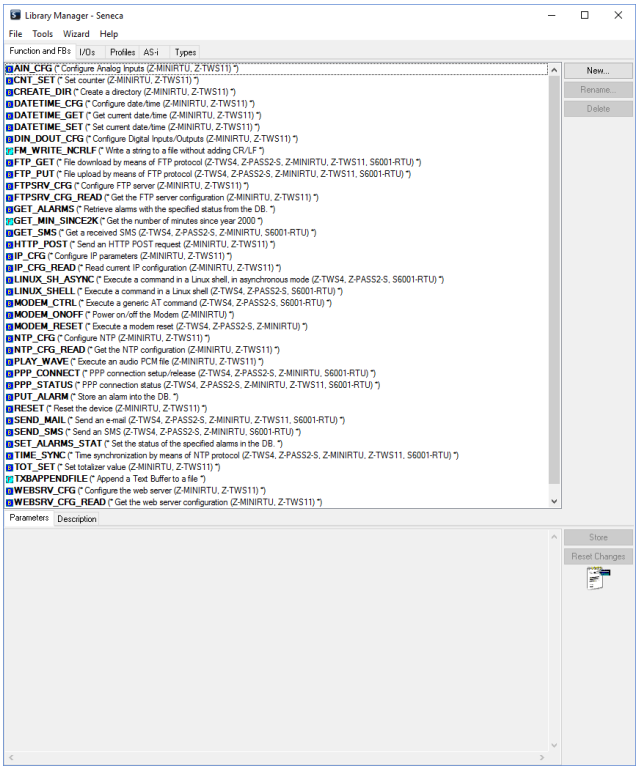


Select the "File / Open Library" option and enter the name "Seneca" to create the new Seneca library.



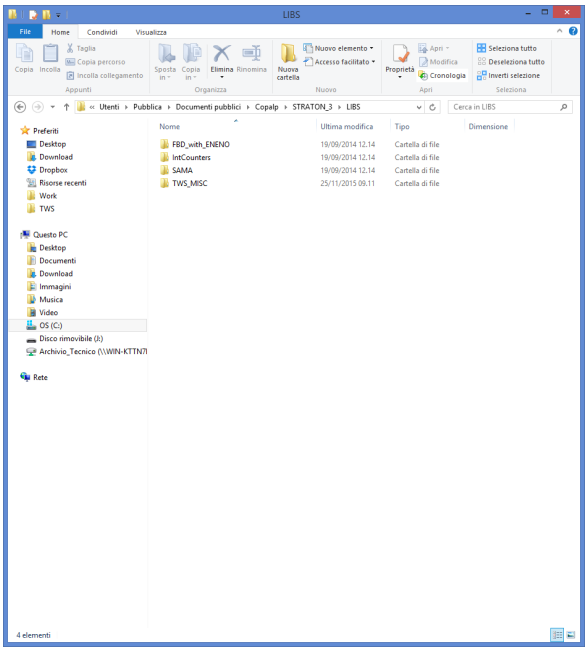
Then, import the Library (“Tools / Import” menu):



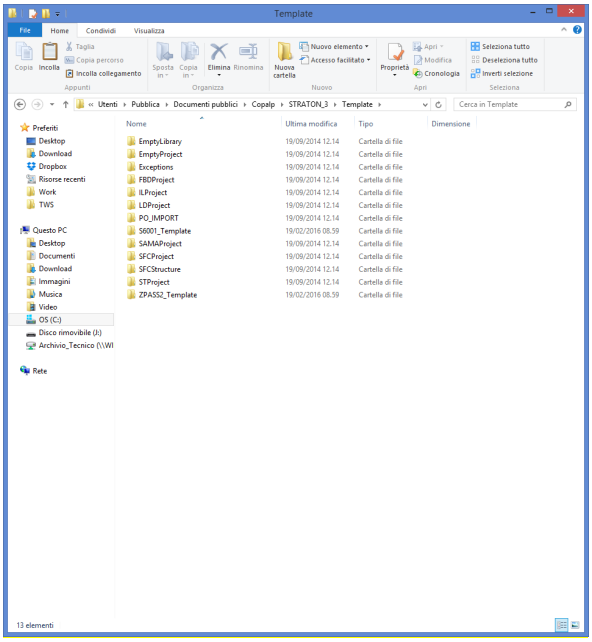


Save the library (“File / Save Library” menu).
Now that the "low level" FBs are available, we need to install the UDFB library.
The UDFB library is provided as a zip file.

The TWS_MISC folder, contained in the zip file, must be copied to the following directory:
C:\Users\Public\Documents\CopaIp\STRATON\LIBS:



The template folders must be copied to the following directory:
C:\Users\Public\Documents\CopaIp\STRATON\Template



17. CYBERSECURITY

Seneca IIOT Gateway devices are regularly subjected to severe tests by third-party companies, in order to verify the effectiveness of data protection systems and unauthorized access by an external attacker.
Continuous monitoring allows greater control over all firmware that is gradually released.

18. WRITING FROM CLOUD TO DEVICE

18.1. WRITING TAGS FROM CLOUD TO DEVICE VIA MQTT

Tags can be written via MQTT in two basic ways.
In the first, the tag name does not appear in the payload, in the second, the tag name is made explicit in the payload.

To write a tag without making its name explicit in the payload, you must subscribe to the topic:

seneca/Z-PASS MQTT Client/info/#

A publish with topic will then be received from the device:

seneca/Z-PASS MQTT Client/info/<nome tag>

and payload.

{"val": <valore tag>}

or

{"value": <valore tag>}

For example:

making the publish to the topic:

seneca/Z-PASS MQTT Client/info/Pippo

with payload:

{“val”: 1234}

The decimal value 1234 is written in the Tag named “Pippo” (be careful with case sensitivity).

To write a tag explicitly stating the name in the payload, you need to subscribe to the topic:

seneca/Z-PASS MQTT Client/info

A publish with topic will then be received from the device:

seneca/Z-PASS MQTT Client/info
and payload.

{"tags": [{"<nome tag>": <valore tag>}]}

For example:

{“tags”: [{"Pippo_fp": 123.46}]}

Writes the floating point value 123.46 in the tag “Pippo_fp”

Or it is possible instead of defining the tag name to use the ID (number that appears in the Tag Vid column (see Tag setup configuration web page):

{"tags_id": [{"<(vid+1)>": <valore tag>}]}

For example:

{"tags_id": [{"25": 789}]}

Writes in the tag with vid = 24 the decimal integer value 789

It is also possible to write more than one tag at the same time with the syntaxes:

{"tags": [{"<nome tag1>": <valore tag1>}, {"<nome tag2>": <valore tag2>},….] }

Or:

{"tags_id": [{"<(vid tag1)+1>": <valore tag1>}, {"< (vid tag2)+1>": <valore tag2>},….] }

For example:

{"tags": [{"Pippo": 1234}, {"Pippo_fp": 123.46}]}
{"tags_id": [{"25": 1234}, {"26": 123.46}]}

They write both tags at the same time.

18.2. SENDING ACTION COMMANDS FROM THE CLOUD TO THE DEVICE VIA MQTT

To send commands to the device via MQTT, the device must receive a PUBLISH, like this:

seneca/Z-PASS MQTT Client/info
{"act": 1}

where:
seneca/Z-PASS MQTT Client/info
is the value of the “Subscribe Topic” parameter of the “MQTT Configuration” webserver configuration page.

The possible “actions” are:

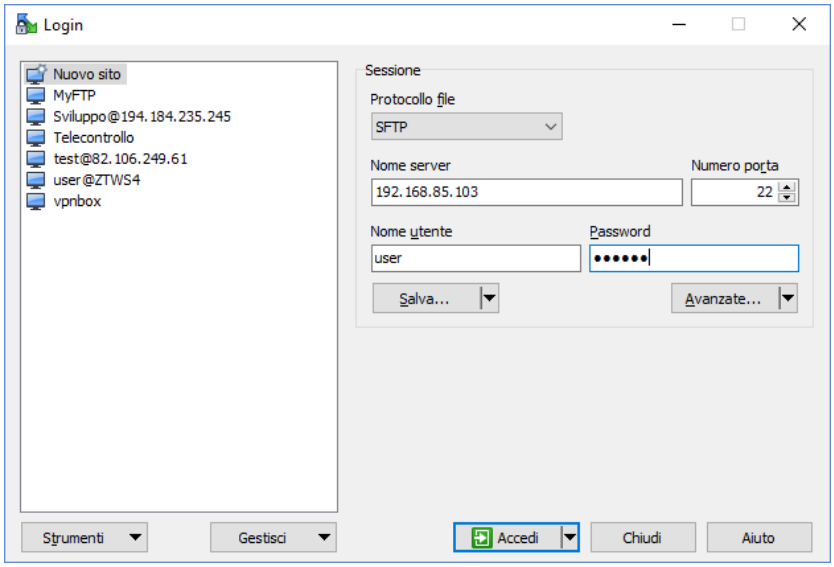
ACT	***COMMAND***
1	Restarts the device
2	Makes the device save the configuration in the URL defined by the parameter “Save Configuration URL” Defined in the configuration webserver page “MQTT Configuration”.
3	Reads the configuration from the URL defined in the “Load Configuration URL” parameter Defined in the “MQTT Configuration” configuration web server page.
4	Downloads the firmware contained in the URL defined by the parameter “FW Update URL” Defined in the “MQTT Configuration” configuration web server page and performs the update.
5	Enables the VPN BOX 2 feature and also activates the mobile network data connection.
6	Enables the VPN BOX 2 function
7	Disables the VPN BOX 2 function
8	Enables the OPEN VPN function
9	Disables the OPEN VPN function
10	Deletes the Data logger files (equivalent to pressing the “Clean Cache” button on the “tag view” configuration web server page.

19. SFTP ACCESS

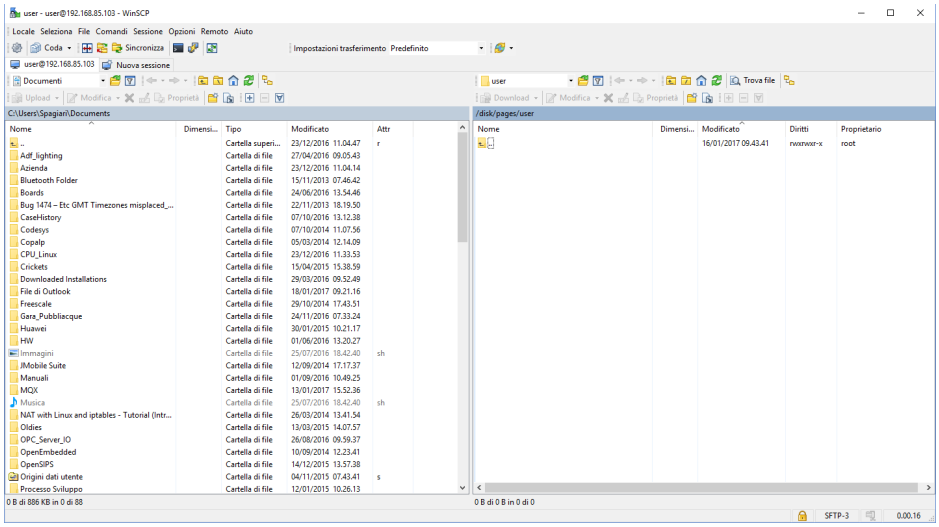
To easily access the device via SFTP, you can for example use the WINSCP program; you can download WINSCP for free from:
http://winscp.net/eng/download.php

Set the connection as in the following figure (the screen shows a connection to IP address 192.168.85.103):

The credentials (username and password) are those ("user", "123456") set for "FTP USER".
After clicking on the "Login" button, a new window will appear, as in the following screen; on the right you can copy and delete files directly on / from the device.



The WinSCP program is used to transfer files from/to the device.



20. MAINTENANCE MODE

The maintenance mode can be activated via webserver or via modbus tcp-ip/RTU.
In maintenance mode, tags cannot be written via the physical or virtual display but only via protocols (ethernet and serial).
To enable the "maintenance mode" set the value of the "Maintenance Mode" register to 1.

21. SMS COMMANDS

On devices with a mobile modem, a number of functions can be controlled using “SMS commands”; these include setting up a mobile data connection (PPP), activating the VPN Box 2 feature, setting up a digital output, etc.

SMS commands can be sent via the phone numbers in the device’s Address Book as “admin” or “manager” users; alternatively, any phone number can send an SMS command, provided that the command contains a “password”; the password is the last four digits of the modem’s IMEI; therefore, the command will have the following format (there must be a blank space between the “password” and the command text):

```
<last four IMEI digits> <command text>
```

Example:

```
6172 PPP ON
```

Please note that the command text can be all uppercase, all lowercase, or a combination of these characters.

Any SMS command received from a number not recognized as the “admin” or “manager” user and not containing the password will be ignored; optionally, these messages and all messages not recognized as valid commands can be “relayed” to the “admin” user.

Example:

```
PPP ON RELAYED
```

SMS commands fall broadly into two categories:
“set” commands that perform an action
“get” commands that request some information
While “get” commands always have a response, “set commands” may or may not be acknowledged, depending on the configuration parameter.
Any response to a command, be it “set” or “get,” will contain the text of the original message plus a result string, for example:
“EXECUTING”
to indicate that the command has been processed successfully; the “ING” form is used to indicate that the procedure started with the command may not have been completed yet
“FAILED”

to indicate that the command could not be processed or something went wrong; in this case there is an error string that provides the reason for the failure

Examples:

```
PPP ON EXECUTING (100.70.179.88)
PPP ON FAILED (System PPP ON)
```

Of course, the response to a “get” command also contains the requested information, if the command was processed successfully.

Example:

```
GET DIN EXECUTING (1,0,0,0)
```

Finally, you can disable the entire SMS command function if not needed via a configuration parameter.
The following paragraphs provide a complete list of supported commands along with their corresponding responses.

21.1. PPP ON

This command can be used to configure the mobile data connection (PPP); the connection is configured with the system configuration parameters (APN Mode, APN, Auth Type, etc.).
If the command is processed correctly, the response contains the IP address assigned to the PPP network interface.
This command is rejected in the following case:

- if the “Remote Connection Disable” (RCD) digital input is HIGH and the “Security Level/Service Disable” parameter is set to “Internet Connection”, the command will fail and generate the “Security Level error”.

Furthermore, if the connection configuration procedure is not completed after the timeout period (currently set to 30 seconds), the command will fail and generate the “Timeout error”.

Please note that failure to activate the mobile data connection with this command is permanent; therefore, if the device is rebooted, the mobile data connection (PPP) will not be re-established.

Example:

```
→    PPP ON
←    PPP ON EXECUTING (100.70.179.88)
```

21.2. PPP OFF

This command can be used to disable the mobile data connection (PPP) set with a previous “PPP ON” command.

Please note that this command does not disable the mobile data connection permanently; therefore, if the device is rebooted, the mobile data connection (PPP) will not be re-established.

This command is never rejected.

Example:

```
→     PPP OFF
←     PPP OFF EXECUTING
```

21.3. PPP IP

This command can be used to get the IP address assigned to the mobile data connection (PPP); if the PPP connection is not active, the “dummy” IP address (0.0.0.0) will be shown.

This command is never rejected.

Example:

```
→     PPP IP
←     PPP IP EXECUTING (100.70.179.88)
```

21.4. PPP CNF

This command can be used to change the value of the system configuration parameters related to mobile data connection (PPP); the changes are permanent.

The command will have the following format and the parameter values must be separated by a blank space:

```
PPP CNF <APN mode> <APN> <Authentication Type> <Username> <Password> <PPP Connection Testing IP Address>
```

All parameters must be present in the above order; no parameter can be left empty.

Regarding the meaning of these parameters: <APN> and <Authentication Type> are numeric fields with the following values:

```
APN Mode
0:    Automatic
1:    Manual

Authentication Type
0:    None
1:    CHAP/PAP
2:    CHAP only
3:    PAP only
```

This command is rejected in the following case:

If any of the command parameters is missing or invalid, the command will fail, resulting in the error “Command parameter error”.

Example:

```
→    PPP CNF 0 mobile.vodafone.it 0 user pass www.google.com
←    PPP CNF EXECUTING
```

21.5. VPN ON

This command can be used to activate the VPN Box function; the function is activated with the system configuration parameters (Server, Password, Tag Name).
The command has two optional parameters, so its format is as follows:
`VPN ON [PPP] [NOFWL]`[1]
"PPP"
If this parameter is present, the mobile data connection (PPP) is configured (if not already active), before activating the VPN Box function
"NOFWL"
If this parameter is present, "Mobile Network Firewall" is disabled in the system configuration
This command is rejected in the following cases:

- if the "custom" VPN function is enabled in the system configuration (parameter "VPN/Enable" = ON, "VPN Mode" = "OpenVPN"), the command will not be executed, generating the "System VPN ON" error;
- if the "Remote Connection Disable" (RCD) digital input is HIGH and the "Security Level/Service Disable" parameter is set to VPN Connection", "VPN Service" or "Internet Connection", the command will fail and generate the "Security Level error".

Please note that this command does not activate the VPN Box function permanently; therefore, if the device is rebooted, the function will not be reactivated.

Examples:

```
→    VPN ON
←    VPN ON EXECUTING

→    VPN ON PPP
←    VPN ON PPP EXECUTING

→    VPN ON NOFWL
←    VPN ON NOFWL EXECUTING

→    VPN ON PPP NOFWL
←    VPN ON PPP NOFWL EXECUTING
```

[1] Square brackets indicate that the parameter is optional.

21.6. VPN OFF

This command can be used to disable the VPN Box function activated with a previous “VPN ON” command; it also disables the mobile data connection (PPP) configured with a previous “VPN ON PPP” command or with the “PPP ON” command.

This command is never rejected.

Please note that this command does not disable the VPN Box function permanently; therefore, if the device is rebooted, the function will be re-enabled.

Example:

```
→     VPN OFF
←     VPN OFF EXECUTING
```

21.7. VPN CNF

This command can be used to change the value of the system configuration parameters related to the VPN Box function; the changes are permanent.

The command will have the following format and the parameter values must be separated by a blank space:

```
VPN CNF <Server> <Password> <Tag Name>
```

All parameters must be present in the above order; no parameter can be left empty.

As for the meaning of these parameters.

This command is rejected in the following case:

If any of the command parameters is missing or invalid, the command will fail and generate the “Command parameter error”.

Example:

```
→     VPN CNF myvpnbox.seneca.it myvpnbox zpass2-GSP
←     VPN CNF EXECUTING
```

21.8. FWL ON

This command can be used to enable “Mobile Network Firewall” in the system configuration (parameter “Mobile Network Firewall/Enable” = ON).
This command is never rejected.

Example:

```
→     FWL ON
←     FWL ON EXECUTING
```

21.9. FWL OFF

This command can be used to disable “Mobile Network Firewall” in the system configuration (parameter “Mobile Network Firewall/Enable” = OFF).
This command is never rejected.

Example:

```
→     FWL OFF
←     FWL OFF EXECUTING
```

21.10. GET DIN

This command can be used to get the status of one or all digital inputs of the device; if a digital input is not available (because it is used as an output)[2], the value “0” is returned.

The command can have two formats:

`GET DIN<n>` with <n>=1..N gets the status of a single digital input

where:
N=4 per R-PASS+R-COMM
N=6 per Z-PASS2-RT-4G

`GET DIN` gets the status of all digital inputs

This command is rejected in the following cases:

- if the digital I/O number is out of range (for example: 0 or N+1), the command will not be executed and generate the “Command parameter error”.

Examples:

```
→     GET DIN
←     GET DIN EXECUTING (1,0,0,0)

→     GET DIN1
←     GET DIN1 EXECUTING (1)
```

[2] This condition can be true for Z-PASS2-RT-4G.

```
→     GET DIN2
←     GET DIN2 EXECUTING (0)
```

21.11. GET DOUT

This command can be used to get the state of one or all of the device's digital outputs; if a digital output is not available (because it is used as an input)[3], the value "0" is provided.

The command can have two formats:

`GET DOUT<n>` con <n>=1..N gets the status of a single digital output

where:
N=4 per R-PASS+R-COMM
N=6 per Z-PASS2-RT-4G

`GET DOUT` gets the status of all digital outputs

This command is rejected in the following cases:

- if the digital I/O number in the command is out of range (for example: 0 or N+1), the command will not be executed and generate the "Command parameter error".

Examples:

```
→     GET DOUT
←     GET DOUT EXECUTING (0,1,0,0)

→     GET DOUT1
←     GET DOUT1 EXECUTING (0)

→     GET DOUT2
←     GET DOUT2 EXECUTING (1)
```

21.12. SET DOUT

This command can be used to set the state of one of the device's digital outputs.

The command can have two formats:

`SET DOUT<n>.CLOSE` with <n>=1..N sets the digital output to HIGH status

`SET DOUT<n>.OPEN` with <n>=1..N sets the digital output to LOW status

where:
N=4 per R-PASS+R-COMM
N=6 per Z-PASS2-RT-4G

[3] This condition can be true for Z-PASS2-RT-4G.

This command is rejected in the following cases:

- if the digital output is not configured as “General output” or the digital I/O is used as input[4], the command will not be executed generating the “Digital I/O mode error”;
- if the digital I/O number in the command is out of range (for example: 0 or N+1), the command will not be executed, generating the “Command parameter error”;
- if the requested state is neither “.CLOSE” nor “.OPEN”, the command will not be executed, generating the “Command parameter error”.

Example:

```
→     SET DOUT2.CLOSE
←     SET DOUT2.CLOSE EXECUTING
```

21.13. SET PULSE

This command can be used to generate a pulse on one of the device's digital outputs.

The command can have two formats:

`SET PULSE<n>.CLOSE <duration>` with <n>=1..N
where:
N=4 per R-PASS+R-COMM
N=6 per Z-PASS2-RT-4G
to generate a LOW-HIGH-LOW pulse, with the HIGH state set for the number of seconds specified by the <duration> parameter

`SET PULSE<n>.OPEN <duration>` with <n>=1..N
where:
N=4 per R-PASS+R-COMM
N=6 per Z-PASS2-RT-4G
to generate a LOW-HIGH-LOW pulse, with the LOW state set for the number of seconds specified by the <duration> parameter

This command is rejected in the following cases:

- if the digital output is not configured as “General output” or the digital I/O is used as input[5], the command will not be executed generating the “Digital I/O mode error”;
- if the digital I/O number in the command is out of range (for example: 0 or N+1), the command will not be executed, generating the “Command parameter error”;
- if the requested state is neither “.CLOSE” nor “.OPEN”, the command will not be executed, generating the “Command parameter error”;
- if the <duration> parameter is missing or invalid, the command will not be executed, generating the “Command parameter error”;

[4] This condition can be true for Z-PASS2-RT-4G.
[5] This condition can be true for Z-PASS2-RT-4G.

- if the ".CLOSE" parameter is specified and the digital output is already in the HIGH state, the command will not be executed, generating the "No pulse generated" error;
- if the ".OPEN" parameter is specified and the digital output is already in the LOW state, the command will not be executed, generating the "No pulse generated" error.

Example:

```
→    SET PULSE2.CLOSE 10
←    SET PULSE2.CLOSE 10 EXECUTING
```

21.14. SET USER.PHONE

This command can be used to insert a user with specified phone number, type and group list into the Address Book; it can also be used to modify the type and/or group list of an existing user.

The command has the following format:

`SET USER.PHONE +<number> <type> <group list>,` with <type>=ADM|MGR|USR

Please note that the phone number must always be indicated with "international format", therefore the initial character '+' must always be present.

"group list" is a list of non-negative integers, separated by the character "-", that defines the groups to which the user belongs. An example of valid group lists is the following:

"1-2-3"
"1-4"
"1"
"0"

The value "0" indicates that the user does not belong to any group.

This command is rejected in the following cases:

- if the <number> parameter already exists in the Address Book, with <type> and <group list> specified, the command will fail, generating the "Item already exists" error;
- if the <number> parameter is missing or invalid (including the case where the '+' character is missing), the command will fail, generating the "Command parameter error";
- if the <type> parameter is missing or invalid, the command will fail, generating the "Command parameter error";
- if the <group list> parameter is missing or invalid, the command will fail, generating the "Command parameter error".

Example:

```
→    SET USER.PHONE +390123456789 ADM 1-2-3
←    SET USER.PHONE +390123456789 ADM 1-2-3 EXECUTING
```

21.15. RESET PHONE

This command can be used to delete a user with the specified phone number from the Address Book.

The command has the following format:

```
RESET PHONE +<number>
```

Please note that the phone number must always be indicated with "international format", therefore the initial character '+' must always be present.

This command is rejected in the following cases:

- if the specified <number> parameter does not exist in the Address Book, the command will fail with the "Item does not exist" error;
- if the <number> parameter is missing or invalid (including the case where the '+' character is missing), the command will fail with the "Command parameter error".

Example:

```
→    RESET PHONE +390123456789
←    RESET PHONE +390123456789 EXECUTING
```

Please note that if the user in the Address Book with the specified phone number also has an email address, this command will also delete it.

21.16. SET USER.EMAIL

This command can be used to insert a user with specified email address, type and group list into the Address Book; it can also be used to change the type and/or group list of an existing user.

The command has the following format:

`SET USER.EMAIL <email address> <type> <group list>`, with <type>=`ADM|MGR|USR`

"group list" is a list of non-negative integers, separated by the character "-", that defines the groups to which the user belongs. An example of valid group lists is the following:

"1-2-3"
"1-4"
"1"
"0"

The value "0" indicates that the user does not belong to any group.

This command is rejected in the following cases:

- if the <email address> parameter already exists in the Address Book, with <type> and <group list> specified, the command will not be executed, generating the "Item already exists" error;

- if the <email address> parameter is missing or invalid, the command will not be executed, generating the “Command parameter error”;
- if the <type> parameter is missing or invalid, the command will fail, generating the “Command parameter error”;
- if the <group list> parameter is missing or invalid, the command will fail, generating the “Command parameter error”.

Example:

```
→    SET USER.EMAIL admin@zpass.it ADM 1-2-3
←    SET USER.EMAIL admin@zpass.it ADM 1-2-3 EXECUTING
```

21.17. RESET EMAIL

This command can be used to delete a user with a specified email address from the Address Book.

The command has the following format:

```
RESET EMAIL <email address>
```

This command is rejected in the following cases:

- if the specified <email address> parameter does not exist in the Address Book, the command will not be executed, generating the “Item does not exist” error;
- if the <email address> parameter is missing or invalid, the command will not be executed, generating the “Command parameter error”.

Example:

```
→    RESET EMAIL admin@zpass.it
←    RESET EMAIL admin@zpass.it EXECUTING
```

Please note that if the user in the Address Book with the specified email address also has a phone number, this number will also be deleted using this command.

21.18. STATUS

This command can be used to get status information from the device.

The status information provided in the response has the following format:

R-PASS+R-COMM:

```
R-PASS<hwrev> <date> <time> RUNNING <service status>,<vpn status>
<DI1>,<DI2>,<DI3>,<DI4>,<DO1>,<DO2>,<DO3>,<D04>
```

Z-PASS2-RT-4G:

```
Z-PASS2-RT-4G<hwrev> <date> <time> RUNNING <service status>,<vpn status> <DIDO1>,<DIDO
2>,<DIDO3>,<DIDO4>,<DIDO5>,<DIDO6>
```

where:

<hwrev>:
<date> is in the format “yyyy/mm/dd”
<hour> is in the format “hh:mm:ss”
<service status> indicates the state of “SRV” LED[6] (“OFF”|”ON”|”FAIL”)
<vpn status> reports the status of the “VPN” LED (“OFF”|”ON”””FAIL”)
<DI1>,<DI2>,…, <DIDO5>,<DIDO6>, status (“LO”|”HI”) of the digital I/Os

This command is never rejected.

Example:

```
→    STATUS
←    STATUS EXECUTING (Z-PASS2-RT-4G 2018/03/09 08:01:31 RUNNING
OFF,OFF HI,LO,HI,LO,LO,LO)
```

21.19. GET GPS

This command can be used to get GPS location information from the device.

The response is provided as a URL on Google Maps™:
https://www.google.com/maps/?q=<latitude>,<longitude>

This command is rejected in the following cases:

- if the GPS signal is not available, the command will not be executed generating the “GPS not fixed” error.

Example:

```
→    GET GPS
←    GET                          GPS                          EXECUTING
(https://www.google.com/maps/?q=45.3742,11.94557)
```

21.20. RESET

This command can be used to reboot the device.

This command is never rejected.

Example:

```
→    RESET
←    RESET EXECUTING
```

21.21. GET TAG

This command can be used to get the value of a tag (see the “Modbus Shared Memory Gateway” feature).

[6] See the “Signal LED” Chapter.

The command has the following format:

```
GET TAG <tag name>
```

Note that "tag name" is case-sensitive; also, this command assumes that each tag has a distinct name; if there are multiple tags with the same name, this command returns the value of the first tag encountered with the specified name.
The value is shown in the response with the following format:

```
<tag value>,VALID
```

or:

```
<tag value>,INVALID
```

The "INVALID" status may occur for tags with "GATEWAY MODE"="GATEWAY", when the last Modbus read request failed.

This command is rejected in the following cases:

- if no serial port has "Gateway Mode"="Modbus Shared Memory", the command will not be executed generating the error "Modbus Gateway not active";
- if no tags with the specified name are found, the command will not be executed generating the "Tag does not exist" error;
- if the requested tag has "GATEWAY MODE"="BRIDGE" and the Modbus read request fails, the command will not be executed generating the "Tag operation failed" error.

Example:

```
→    GET TAG GPS_LONGITUDE
←    GET TAG GPS_LONGITUDE EXECUTING (11.94528,VALID)
```

21.22. SET TAG

This command can be used to set the value of a tag (see the "Modbus Shared Memory Gateway" feature).
The command has the following format:

```
SET TAG <tag name> <tag value>
```

Note that "tag name" is case-sensitive; also, this command assumes that each tag has a distinct name; if there are multiple tags with the same name, this command attempts to set the value of the first tag encountered with the specified name.
For non-integer tag values, the decimal point character '.' will be used.

This command is rejected in the following cases:

- if no serial port has "Gateway Mode"="Modbus Shared Memory", the command will not be executed generating the error "Modbus Gateway not active";

- if no tags with the specified name are found, the command will not be executed generating the “Tag does not exist” error;
- if the specified value does not match the “Data Type” of the target tag (for example, the value “2” for a “BOOLEAN” tag), the command will fail with an “Invalid value for tag” error;
- if, for any reason, the write operation fails, the command will fail with a “Tag operation failed” error; this includes the following cases:
    - Modbus write request fails for “GATEWAY” or “BRIDGE” tags;
    - the tag value cannot be changed because it is not a “General output” tag, for digital I/O tags (“EMBEDDED”);
    - the tag value cannot be changed because it is a “GPS info” tag (“EMBEDDED”).

Example:

→ `SET TAG ZPASS_DO 10`

← `SET TAG ZPASS_DO 10 EXECUTING`

21.23. OVPN ON

This command can be used to activate the standard OPEN VPN function; the function is activated with the system configuration parameters (Server, Password, Tag Name).
Please note that this command does not activate the OPEN VPN function permanently; therefore, if the device is rebooted, the function is not reactivated.

Examples:

→ `OVPN ON`

← `OVPN ON EXECUTING`

21.24. OVPN OFF

This command can be used to disable the OPEN VPN feature activated with a previous “OVPN ON” command.

Please note that this command does not disable the OPEN VPN feature permanently; therefore, if Z-PASS is restarted, the feature will be re-enabled.

Example:

→ `OVPN OFF`

← `OVPN OFF EXECUTING`

21.25. CLEAN LOGS

This command will delete all data logs.

→ `CLEAN LOGS`

← `CLEAN LOGS EXECUTING`

22. DEVICE FIRMWARE UPDATE

The firmware can be updated from the web page (FW UPDATE section) or with a USB stick formatted with the FAT32 file system.

22.1. AUTOMATIC UPDATE NOTIFICATION

From firmware revision 4.0.0.0, the system connects to the Seneca servers via the internet to autonomously check if new firmware is available for download.
If so, a notification appears on the web page, including a link to download the new firmware directly to your PC.
The presence of the new firmware is shown in a pop-up on the display/virtual display; an icon also appears in the bottom bar.

22.2. FIRMWARE UPDATE FROM USB STICK

For updating fw from USB port, the procedure is as follows:

Check that the USB host port is enabled by the webserver

Download the FW file from the Seneca website

the downloaded file is a .zip file; extract the .bin file; the FW file must be of the following type:

SW00xxxx_xxx.bin

1) Copy this file to the root of the USB pen
2) Turn off the device
3) Insert the USB pen into the USB port
4) Turn on the device

The update procedure will take a few minutes to complete; during this time, the device MUST NOT be turned off.

23. FACTORY RESET

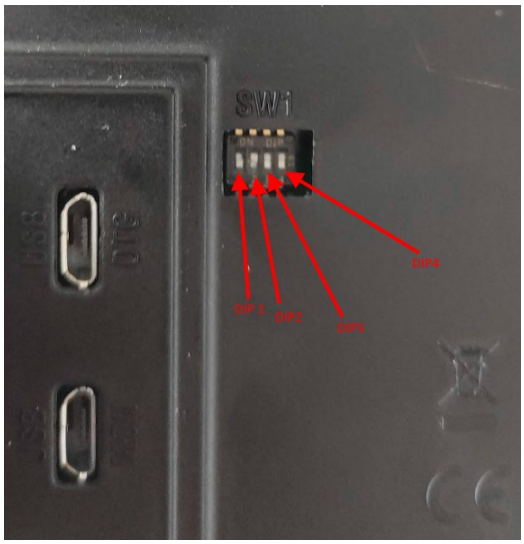
With this procedure it is possible to obtain

1) All the parameters at the factory
2) All folders are cleared (and therefore all data log files and debugging files are deleted).

23.1. FACTORY RESET FOR SSD

To obtain a factory reset follow the following procedure:

1) Turn off the device
2) Reach the back of the device and locate the dip switches as shown in the picture:



3) Bring the dip switches in: DIP1 = OFF, DIP2 = ON, DIP3 = ON, DIP4 = ON
4) Switch the device on and wait until it has completed charging
5) With the device switched on, bring the dips in: DIP1 = OFF, DIP2 = ON, DIP3 = OFF, DIP4 = OFF

23.2. FACTORY RESET FOR R-PASS AND R-PASS-S

To obtain a factory reset follow the following procedure:

1) Turn off the device
2) Reach the back of the device and locate the dip switches as shown in the picture:

3) Bring the dip switches in: DIP1 = OFF, DIP2 = ON, DIP3 = ON, DIP4 = ON
4) Switch the device on and wait until it has completed charging
5) With the device switched on, bring the dips in: DIP1 = OFF, DIP2 = ON, DIP3 = OFF, DIP4 = OFF

23.3. FACTORY RESET FOR Z-PASS1-RT, Z-PASS2-RT, Z-TWS4-RT-S, Z-PASS2-RT-S

To obtain a factory reset follow the following procedure:

1) Turn off the device
2) Reach the back of the device by removing the cover on the bottom of the device and locate the DIP SW1 set
3) Bring the dip switches in: DIP1 = ON, DIP2 = ON, DIP3 = ON, DIP4 = OFF, DIP5=ON, DIP6 =ON
4) Switch the device on and wait until it has completed charging
5) Returns dips to: DIP1 = ON, DIP2 = ON, DIP3 = ON, DIP4 = OFF, DIP5=OFF, DIP6 =OFF

24. I/O EMBEDDED

Depending on the model embedded I/Os are available on devices:

MODEL	*CONFIGURABLE DIGITAL I/O*	*ANALOG INPUTS*
SSD	2 DIDO	NO
SSD-S	2 DIDO	NO
SSD-E	2 DIDO	NO
R-PASS	4DI 4DO	2
R-PASS-S	4DI 4DO	2
R-PASS-E	4DI 4DO	2
Z-PASS1-RT	6 DIDO	2
Z-TWS4-RT	6 DIDO	2
Z-TWS4-RT-E	6 DIDO	2
Z-PASS2-RT-4G	6 DIDO	2
Z-PASS2-RT-4G-S	6 DIDO	2
Z-PASS2-RT-4G-E	6 DIDO	2

24.1. EMBEDDED I/Os UPDATE TIMES

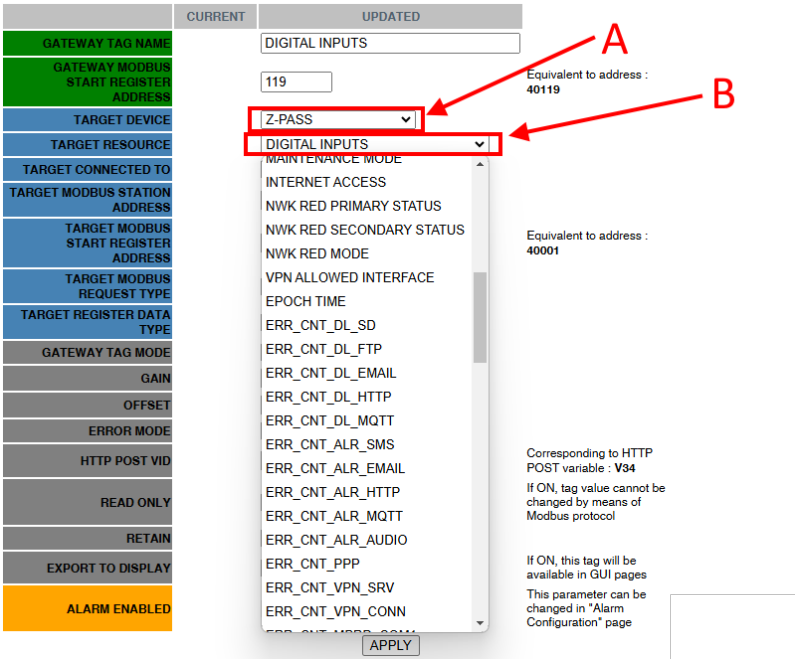
EMBEDDED I/Os UPDATE TIMES ARE:

I/O	*TIME (LOGICAL RULES or TAG) [ms]*	*TIME (ON IO Profile PLC STRATON) [ms]*
DIGITAL INPUTS	500	PLC cycle time
DIGITAL OUTPUTS	500	100
ANALOG INPUTS	500	100
DIGITAL COUNTERS	Max 1000 Hz	Max 1000 Hz

The embedded I/Os of the devices can also be accessed externally via the Modbus TCP-IP or RTU protocol through the addresses given in the following chapters (default station address is 254)
In addition to these, there are other TAGS that are disabled by default; on the “Setup Tag” web page it is possible to enable them (for example the serial master modbus error counter).

24.2. ENABLE NOT ACTIVE BY DEFAULT EMBEDDED TAG

It is possible to enable other TAGs that are disabled by default from the “Setup Tag” web page, for example to be used in logical rules.
To do this, select the “Target Device” based on the device you are using and consequently set the desired tag in the “Target Resource” field.



In addition to the embedded IOs, tags are available for GPS position, redundancy, VPN and error counters.

24.3. AVAILABLE MODBUS ADDRESSES FOR SSD DEVICE

Default Address	Offset	Register Type	I/O
40001	0	Holding Registers	Bit 0: DI1 (LSB) Bit 1: DI2
40002	1	Holding Registers	Bit 0: DO1 (LSB) Bit 1: DO2
40003	2	Holding Registers	Bit 0: Maintenance Mode
40015	14	Holding Registers	Analog Input 1 (INT16)
40016	15	Holding Registers	Analog Input 2 (INT16)
40051	50	Holding Registers	Internet Access (0 = None, 1 = ETH, 2 = WIFI)

10001	0	Discrete Inputs	DI1
10002	1	Discrete Inputs	DI2
0	0	Coils	DO1
1	1	Coils	DO2
40061-40062	60-61	Holding Registers	Counter 1
40063-40064	62-63	Holding Registers	Counter 2

24.4. MODBUS ADDRESSES OF R-PASS I/Os

Default Address	Offset	Register Type	I/O
40001	0	Holding Registers	Bit 0: DI1 (LSB) Bit 1: DI2 Bit 2: DI3 Bit 3: DI4
40002	1	Holding Registers	Bit 0: DO1 (LSB) Bit 1: DO2 Bit 2: DO3 Bit 3: DO4
40003	2	Holding Registers	Bit 0: Maintenance Mode
40015	14	Holding Registers	Analog Input 1 (INT16)
40016	15	Holding Registers	Analog Input 2 (INT16)
40051	50	Holding Registers	Internet Access (0 = None, 1 = ETH, 2 = WIFI)
10001	0	Discrete Inputs	DI1
10002	1	Discrete Inputs	DI2
10003	2	Discrete Inputs	DI3
10004	3	Discrete Inputs	DI4
0	0	Coils	DO1
1	1	Coils	DO2

2	2	Coils	DO3
3	3	Coils	DO4
40061-40062	60-61	Holding Registers	Counter 1
40063-40064	62-63	Holding Registers	Counter 2
40065-40066	64-65	Holding Registers	Counter 3
40067-40068	66-67	Holding Registers	Counter 4

24.5. MODBUS ADDRESSES OF Z-PASS1-RT, Z-PASS2-RT I/Os

Default Address	Offset	Register Type	I/O
40001	0	Holding Registers	Bit 0: DI1 (LSB) Bit 1: DI2 Bit 2: DI3 Bit 3: DI4 Bit 4: DI5 Bit 5: DI6
40002	1	Holding Registers	Bit 0: DO1 (LSB) Bit 1: DO2 Bit 2: DO3 Bit 3: DO4 Bit 4: DO5 Bit 5: DO6
40003	2	Holding Registers	Bit 0: Maintenance Mode
40015	14	Holding Registers	Analog Input 1 (INT16)
40016	15	Holding Registers	Analog Input 2 (INT16)
40051	50	Holding Registers	Internet Access (0 = None, 1 = ETH, 2 = WIFI)
10001	0	Discrete Inputs	DI1
10002	1	Discrete Inputs	DI2

10003	2	Discrete Inputs	DI3
10004	3	Discrete Inputs	DI4
10005	4	Discrete Inputs	DI5
10006	5	Discrete Inputs	DI6
0	0	Coils	DO1
1	1	Coils	DO2
2	2	Coils	DO3
3	3	Coils	DO4
4	4	Coils	DO5
5	5	Coils	DO6
40061-40062	60-61	Holding Registers	Counter 1
40063-40064	62-63	Holding Registers	Counter 2
40065-40066	64-65	Holding Registers	Counter 3
40067-40068	66-67	Holding Registers	Counter 4
40069-40070	68-69	Holding Registers	Counter 5
40071-40072	70-71	Holding Registers	Counter 6

24.6. COMMON RESOURCES MODBUS ADDRESSES

Data Type	***Register***	***Default address***
Holding Register	EPOCH TIME (UINT32) Time in seconds since 1/1/1970	45 (40046) – 46 (40047)
Holding Register	INTERNET ACCESS (UINT16) Defines through which network interface access to the Internet takes place (default gateway).	50 (40051)

	0: None 1: Ethernet (WAN) 2: Wi-Fi 3: Mobile (PPP)	
Holding Register	NETWORK REDUNDANCY PRIMARY STATUS (UINT16) Primary interface status 0: KO 1: OK	51 (40052)
Holding Register	NETWORK REDUNDANCY SECONDARY STATUS (UINT16) Secondary interface status 0: KO 1: OK	52 (40053)
Holding Register	NETWORK REDUNDANCY MODE (RW) (UINT16) Registry to configure the type of redundancy between: 0: OFF 1: WAN/Mobile 2: WAN/Wi-Fi 3: Mobile/WAN 4: Wi-Fi/WAN	53 (40054)
Holding Register	VPN ALLOWED INTERFACE (RW) (UINT16) Allows you to force the VPN to work on a specific interface 0: Auto 1: Mobile 2: WAN	54 (40055)

	3: Wi-Fi	

24.7. GNSS MODBUS ADDRESSES (ONLY FOR Z-PASS2-RT AND R-PASS WITH R-COMM OPTION)

Register	*Address*	*Data type*	*Description*	*Reading/Writing*
GPS_ERROR	40101	INT16	0: OK -1: GPS not fixed -2: GPS not available	RO
GPS_UTC_HH	40102	UINT16	UTC/hours	RO
GPS_UTC_MM	40103	UINT16	UTC/minutes	RO
GPS_UTC_SS	40104	UINT16	UTC/seconds	RO
GPS_DATE_DD	40105	UINT16	Date/day	RO
GPS_DATE_MM	40106	UINT16	Date/month	RO
GPS_DATE_YY	40107	UINT16	Date/year	RO
GPS_LATITUDE	40108	FLOAT64	Latitude	RO
GPS_LONGITUDE	40112	FLOAT64	Longitude	RO
GPS_ALTITUDE	40120	FLOAT64	Altitude	RO
GPS_COG	40124	FLOAT64	Ground heading	RO
GPS_SPKN	40132	FLOAT64	Speed in knots	RO

24.8. MODBUS ADDRESSES WITH ERROR COUNTERS

The error counters reset as soon as the error condition ends or can be reset with a writing (to the value 0).

Register	Address	Data type	Description	Reading/Writing
ERR_CNT_DL_SD	40151	UINT16	Error Counter for Data Logger SD protocol	RW
ERR_CNT_DL_FTP	40152	UINT16	Error Counter for Data Logger FTP protocol	RW
ERR_CNT_DL_EMAIL	40153	UINT16	Error Counter for Data Logger EMAIL protocol	RW
ERR_CNT_DL_HTTP	40154	UINT16	Error Counter for Data Logger HTTP protocol	RW
ERR_CNT_DL_MQTT	40155	UINT16	Error Counter for Data Logger MQTT protocol	RW

ERR_CNT_ALR_SMS	40156	UINT16	Error Counter for SMS alarms	RW
ERR_CNT_ALR_EMAIL	40157	UINT16	Error Counter for EMAIL alarms	RW
ERR_CNT_ALR_HTTP	40158	UINT16	Error Counter for HTTP alarms	RW
ERR_CNT_ALR_MQTT	40159	UINT16	Error Counter for MQTT alarms	RW
ERR_CNT_ALR_AUDIO	40160	UINT16	Error Counter for audio call alarms	RW
ERR_CNT_PPP	40161	UINT16	Error Counter for PPP (mobile data connection)	RW
ERR_CNT_VPN_SRV	40162	UINT16	Error Counter for VPN Box service channel	RW
ERR_CNT_VPN_CONN	40163	UINT16	Error Counter for VPN Box or OpenVPN connection	RW
ERR_CNT_MBRD_COM1	40164	UINT16	Error Counter for Modbus read transactions on COM1 port	RW
ERR_CNT_MBWR_COM1	40165	UINT16	Error Counter for Modbus write transactions on COM1 port	RW
ERR_CNT_MBRD_COM2	40166	UINT16	Error Counter for Modbus read transactions on COM2 port	RW
ERR_CNT_MBWR_COM2	40167	UINT16	Error Counter for Modbus write transactions on COM2 port	RW
ERR_CNT_MBRD_COM4	40168	UINT16	Error Counter for Modbus read transactions on COM4 port	RW
ERR_CNT_MBWR_COM4	40169	UINT16	Error Counter for Modbus write	RW

			transactions on COM4 port	
ERR_CNT_MBRD_TCP1	40170	UINT16	Error Counter for Modbus read transactions with TCP Server 1	RW
ERR_CNT_MBWR_TCP1	40171	UINT16	Error Counter for Modbus write transactions with TCP Server 1	RW
ERR_CNT_MBRD_TCP2	40172	UINT16	Error Counter for Modbus read transactions with TCP Server 2	RW
ERR_CNT_MBWR_TCP2	40173	UINT16	Error Counter for Modbus write transactions with TCP Server 2	RW
……	…..	…..	…..	…..
ERR_CNT_MBRD_TCP5	40218	UINT16	Error Counter for Modbus read transactions with TCP Server 25	RW
ERR_CNT_MBWR_TCP5	40219	UINT16	Error Counter for Modbus write transactions with TCP Server 25	RW

24.9. MODBUS ADDRESSES RELATING TO THE MOBILE MODEM

Below is the list of modbus registers that have information relating to the mobile modem, available only in models that support it.

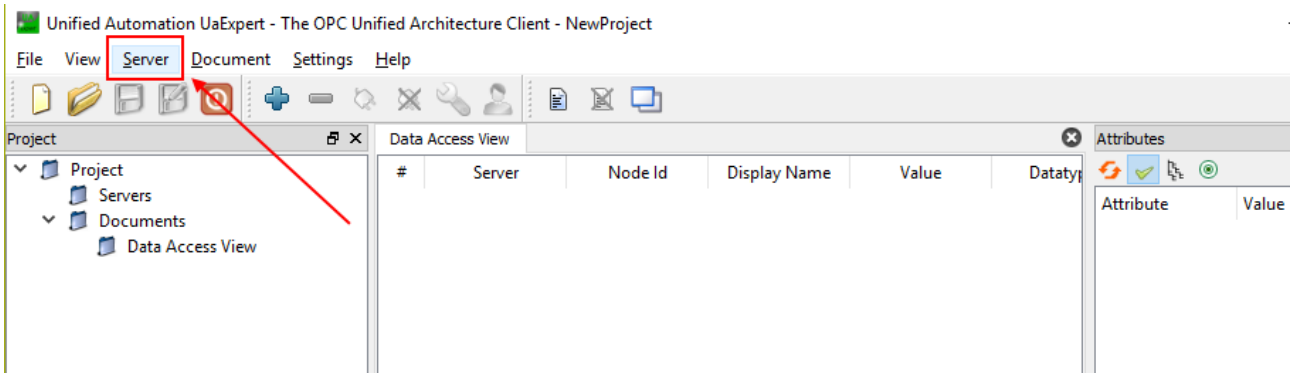
Register	Address	Data type	Description	Reading/Writing
SIGNAL_LEVEL	43001	INT16	signal level in dBm; 255 = invalid	RO
ACCESS_TECHNOLOGY	43002	UINT16	0: invalid 2: 2G 3: 3G 4: 4G	RO
OPERATOR	43003	UINT32	MCC-MNC of the selected operator (e.g.: 22210)	RO
ICCID	43005	INT64	ICCID of the SIM	RO

			NOTE: the first 4 digits "8939" are omitted	
IMEI	43009	INT64	IMEI of the modem	RO
CONN_TS	43013	UINT32	Connection activation timestamp ("epoch" time)	RO
GW_MODE	43015	UINT16	RESERVED	RO
SIGNAL_SCALE	43016	UINT16	0: signal absent [1..5] signal level	RO
APN	43021-43041	STRING (40)	APN (Access Point Name)	RO
OPER_STRING	43042-43052	STRING (20)	operator as string	RO
SIGNAL_RSSI	43053	UINT16	signal level in RSSI	RO
LAC	43054	UINT16	LAC (Location Area Code)	RO
CELL_ID	43055	UINT32	Cell Identity	RO

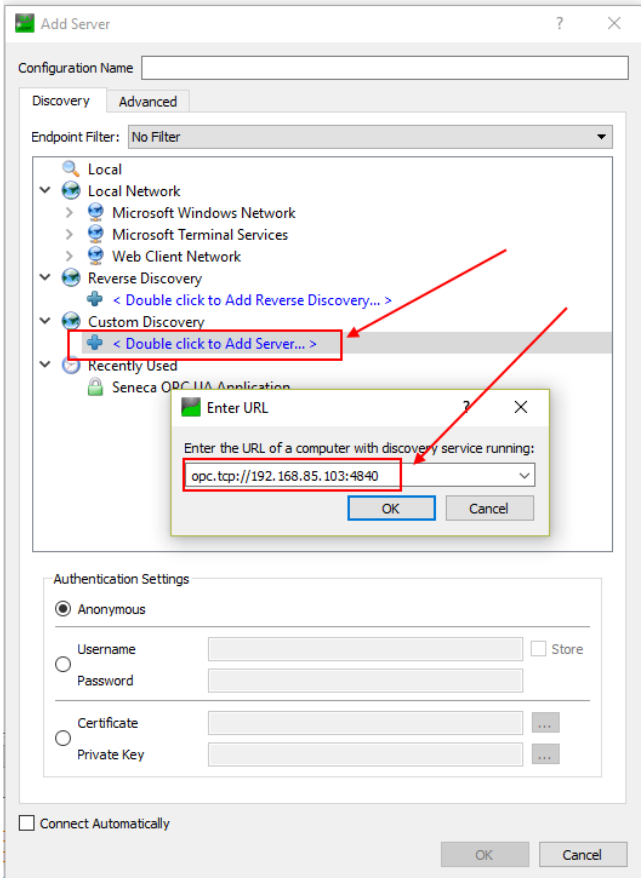
25. "UA EXPERT" CLIENT CONFIGURATION

This chapter will provide the steps to configure the connection and the correct security policy with the "UA Expert" client software

Click Server-> Add

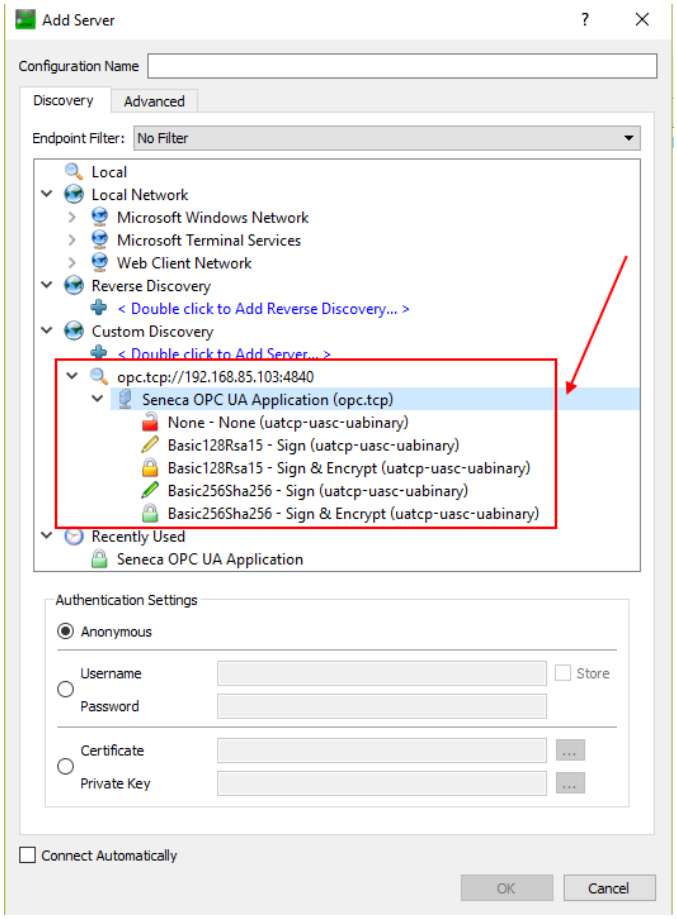


In "Custom Discovery" enter the url for the OPC-UA server:



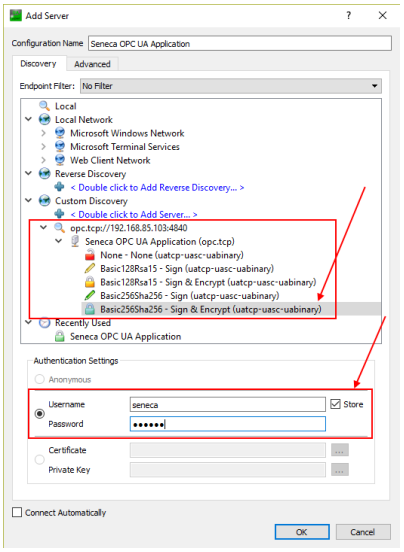
Press OK.

Supported security policies are now displayed:



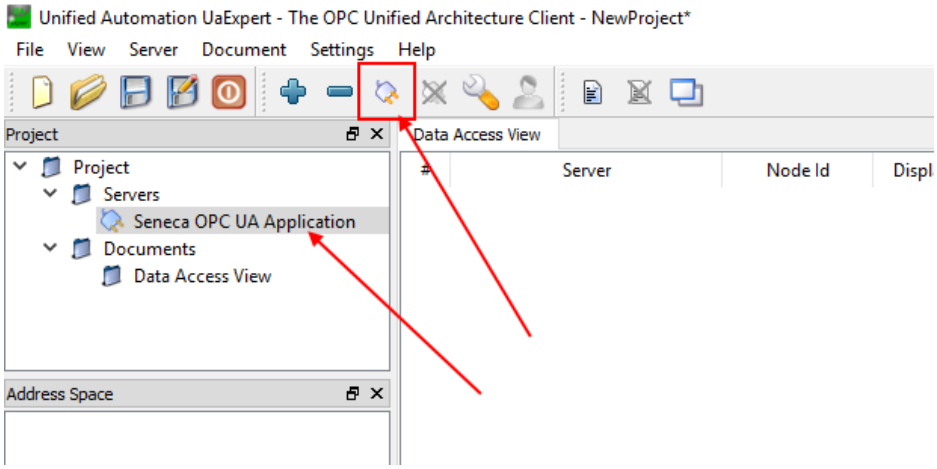
Select the one to use.

Then go to Authentication settings and enter the username and password configured in the OPC-UA server:

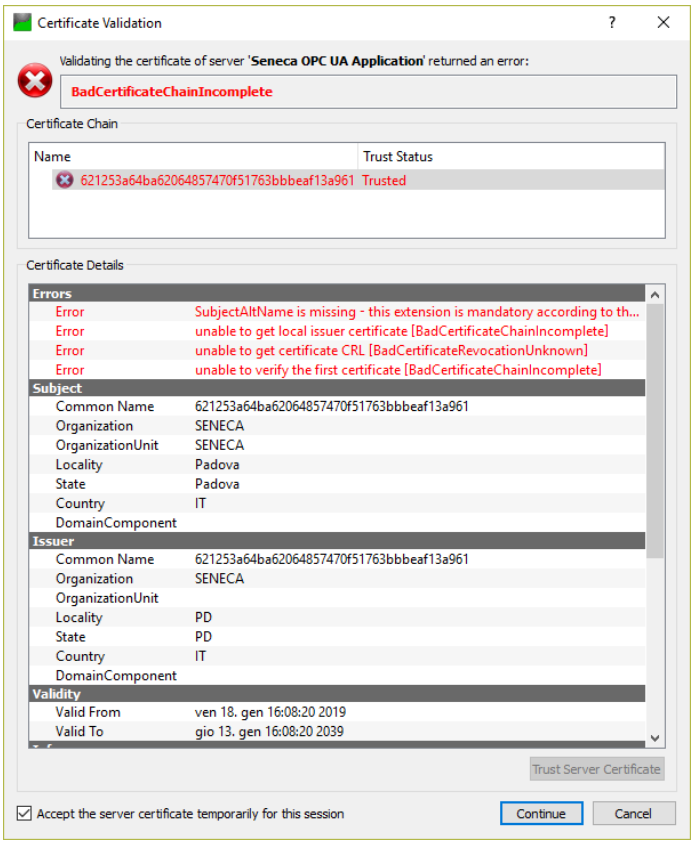


Press OK:

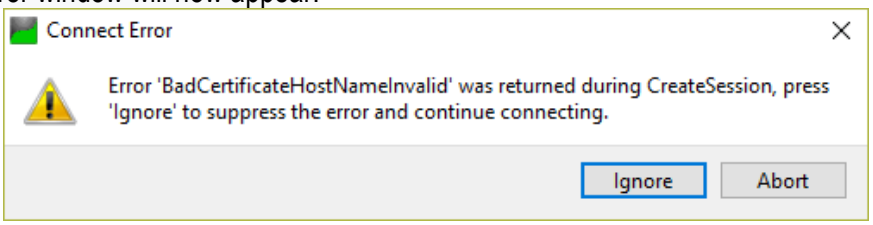
Now it is possible to connect to the server using the appropriate icon:



A new server certificate validation dialog will open. After reviewing the certificate, select Trust Server Certificate to permanently add the certificate to UaExpert's trust list. It is also possible to check the appropriate box to temporarily accept the server certificate for this session and choose Continue to not save the certificate to the trusted list or select Cancel to reject the certificate.

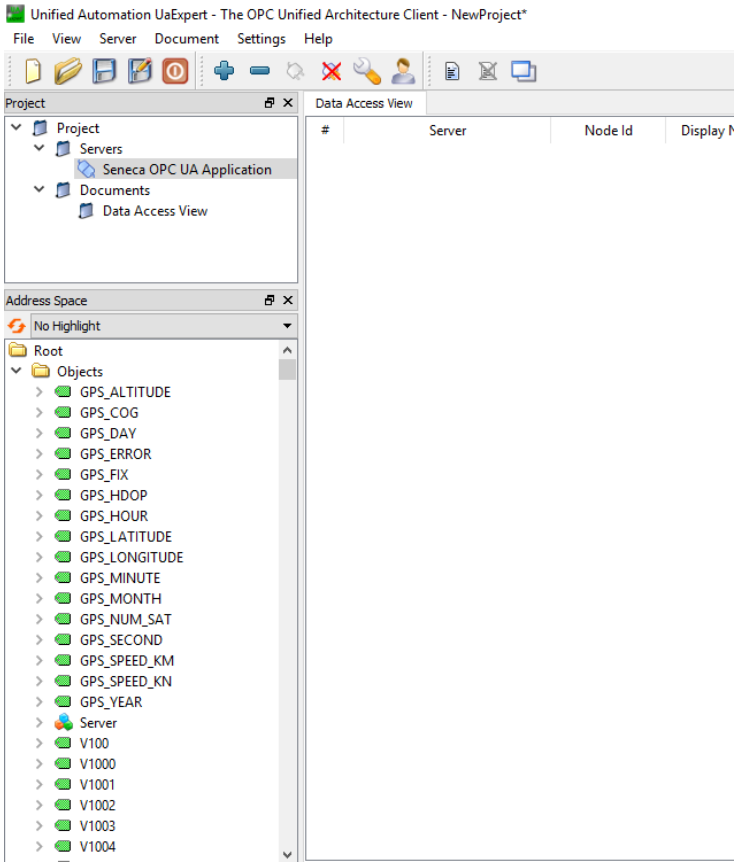


The Certificate Error window will now appear:

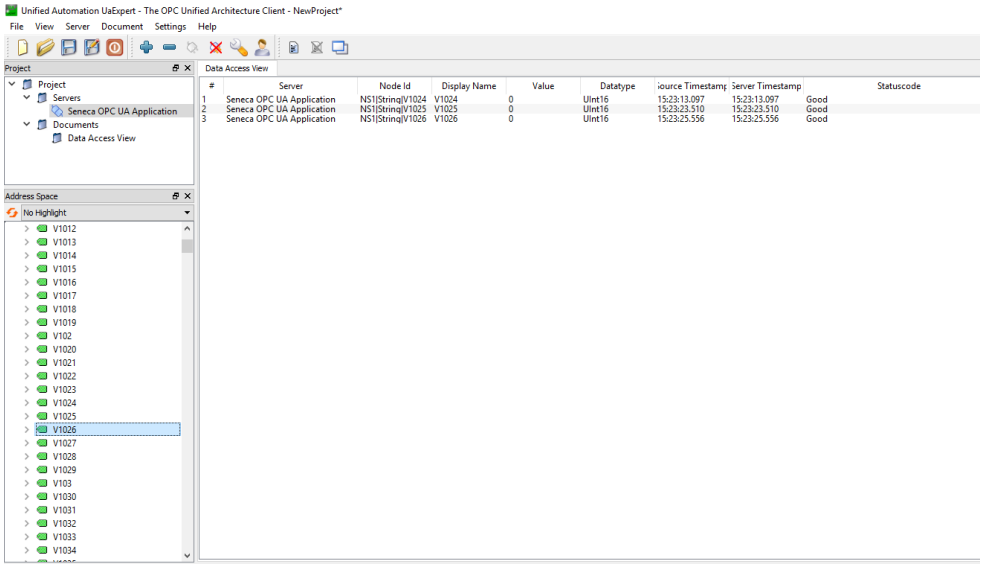


Click on "Ignore" to continue.

Now the connection is established, you can read/write the value of the tags



To update the tags in real time, drag and drop what you want to display:



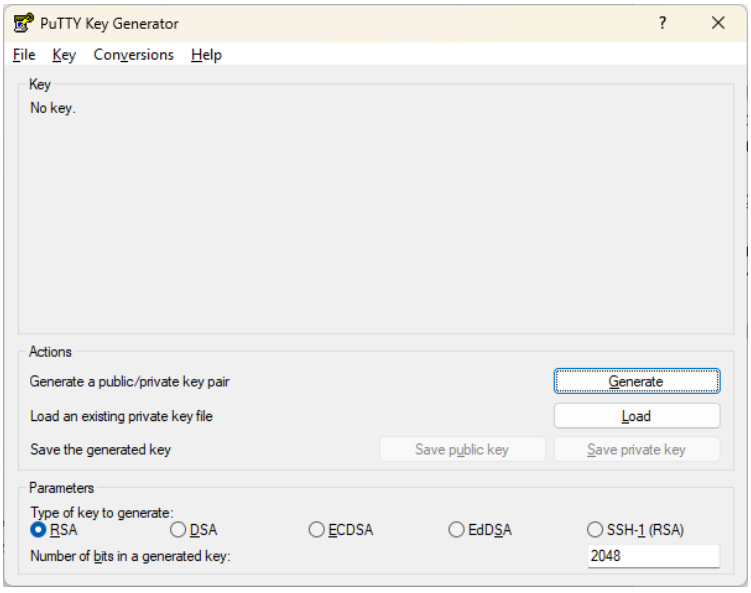
26. KEYS CREATION FOR SSH CONNECTION

The following chapter describes the procedure for creating public and private keys for accessing the device via ssh.
The creation of the keys requires the use of the putty software, which can be downloaded from:

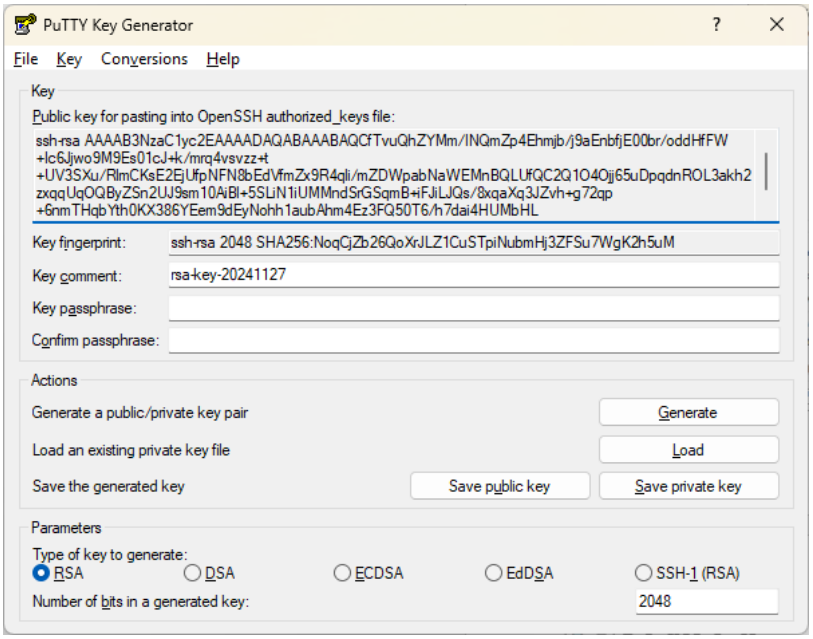
https://www.putty.org/

To create and use SSH keys on Windows, it is necessary to install PuTTY. This software also installs other tools that are necessary for our purpose.

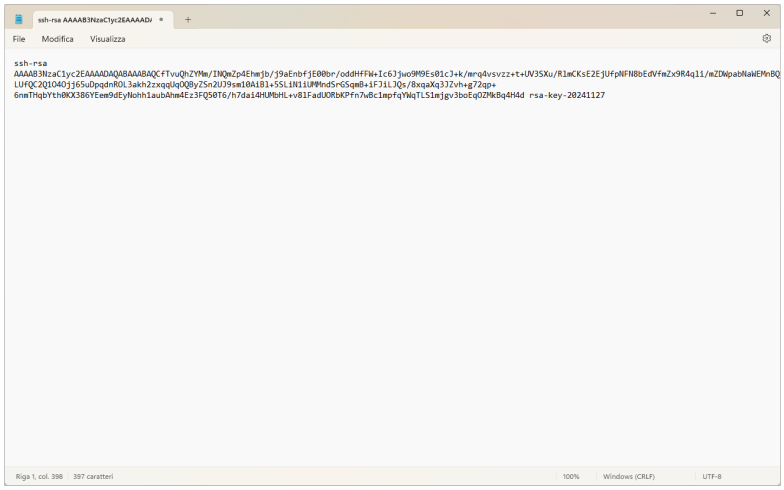
After installing putty, open the PuTTYgen program:



Now press the “Generate” button:



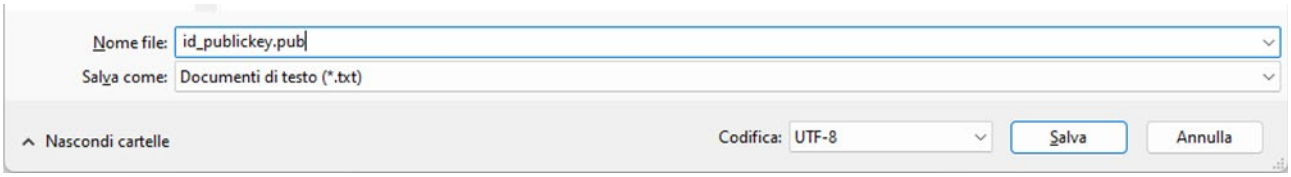
The public key appears in the textbox, which must be copied to the device; do not save the key by pressing the button but copy/paste it into a new file, making sure to select the ENTIRE key:



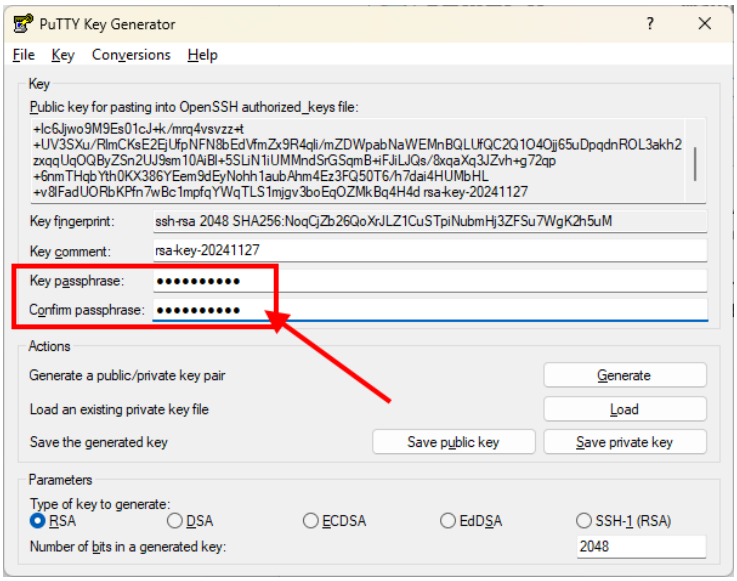
To be loaded into the device, the file type must be:
“id_*.pub “

For example, rename the file as:

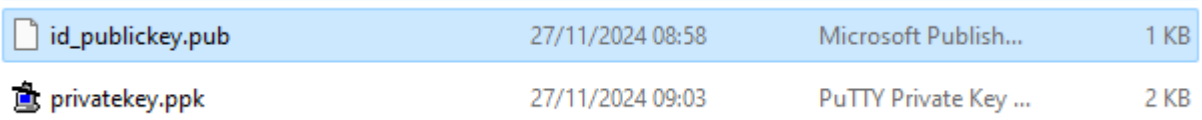
“id_publickey.pub”:



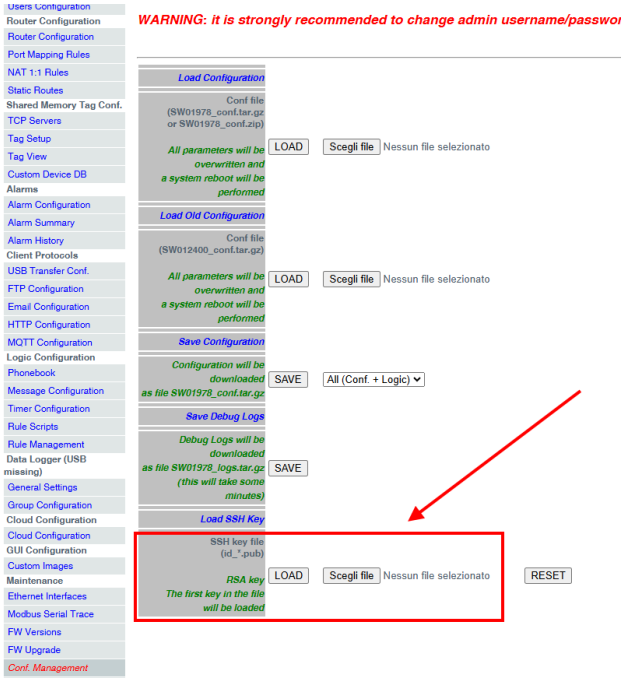
It is now possible to save the private key. To do so, enter a password:



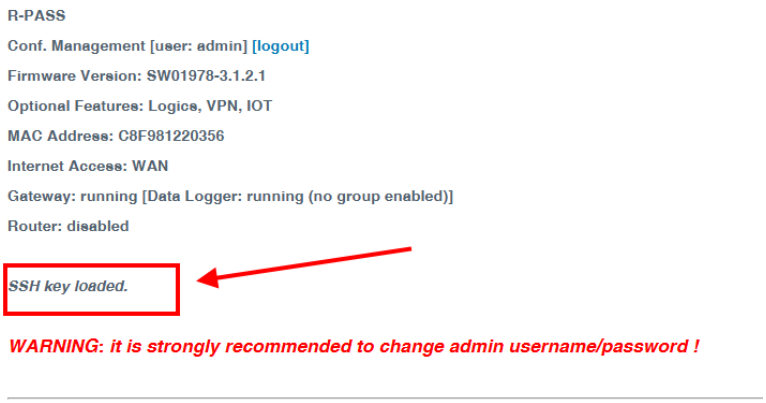
Once finished, click the Save Private Key button and select a safe place to store it. You can name your key as you like. The ".ppk" extension will be added automatically.
We now have the 2 key files, public and private:



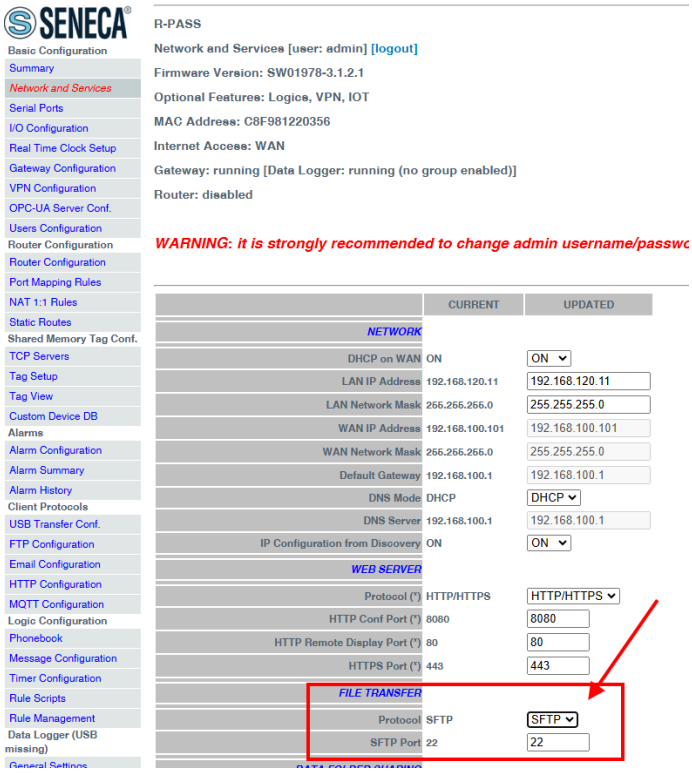
We can now load the public key “id_publickey.pub” into the edge device from the “conf_management” page:



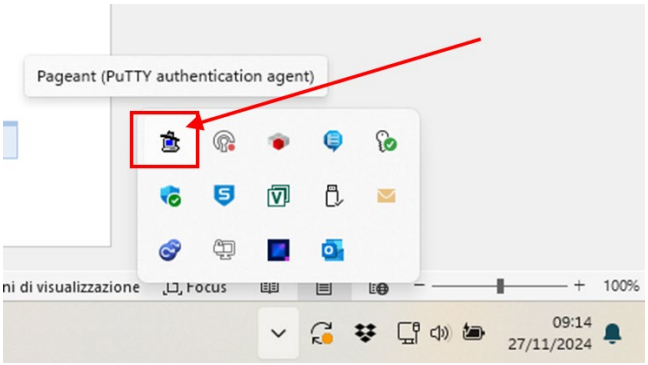
Press the “LOAD” button to load the selected file. The following screen appears:



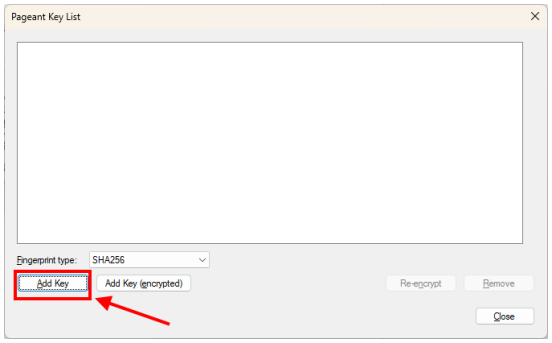
It is now possible to activate the sftp/ssh service in the edge device:



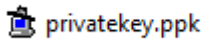
Let's now run the software on pc windows pageant (always part of the Putty installation). At the end of the procedure it can be found here:



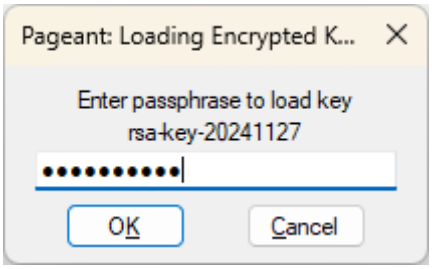
Double-click on the icon and select “Add Key”:



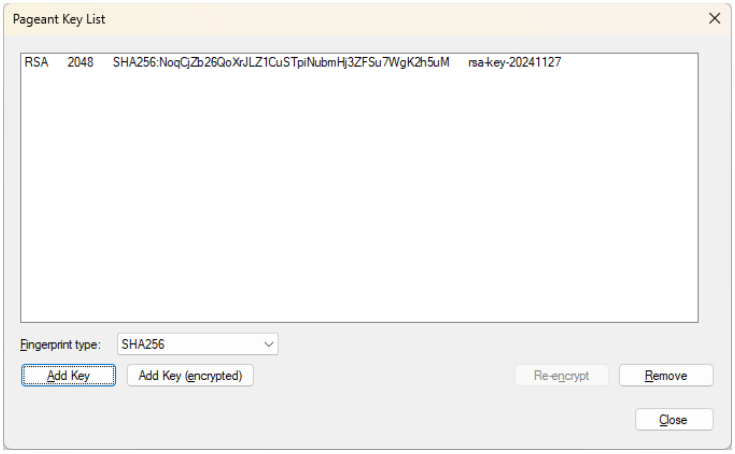
And select the newly generated private key:



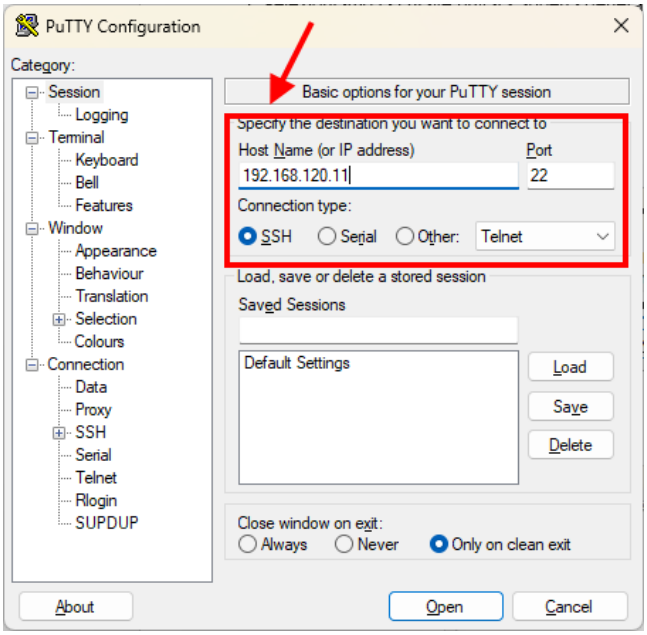
We will be prompted to enter the previously set password:



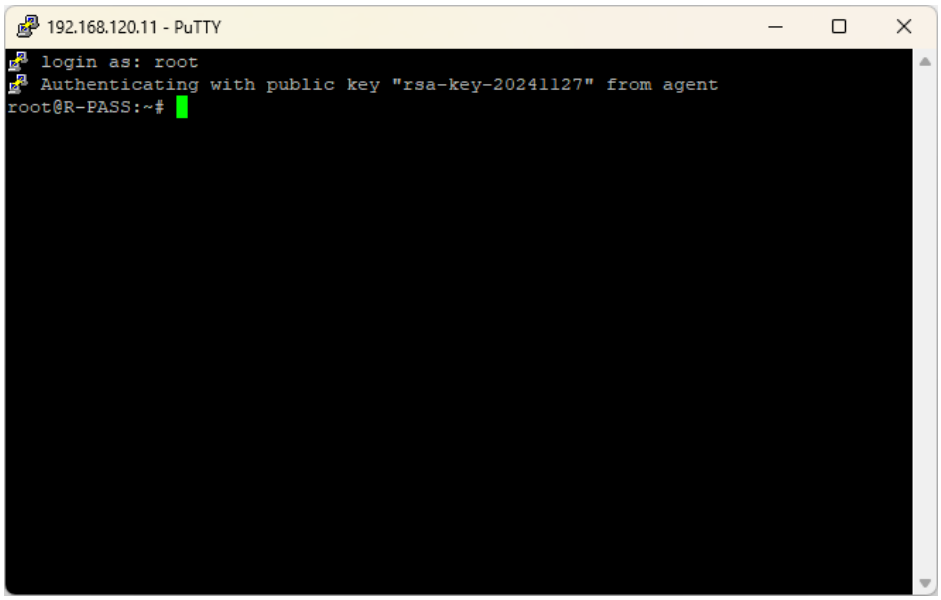
And confirm with OK:



The private key is now installed in Putty. We can press “Close” and connect with Putty:



We can now access as root:



ATTENTION!

Each time the PC is restarted, it will be necessary to reload the private key with the pageant software

ATTENTION!

Activating the sFTP/SSH service may lead to a decrease in the defences of the edge device against external attacks (potential cybersecurity problems). Once maintenance via ssh has been completed, Seneca suggests disabling the service.

27. NUMBERING OF "0-BASED" OR "1-BASED" MODBUS ADDRESSES

According to the Modbus standard the Holding Registers are addressable from 0 to 65535, there are 2 different conventions for numbering the addresses: "0-BASED" and "1-BASED".
For greater clarity, Seneca shows its register tables in both conventions.

ATTENTION!

CAREFULLY READ THE DOCUMENTATION OF THE MODBUS MASTER DEVICE IN ORDER TO UNDERSTAND WHICH OF THE TWO CONVENTIONS THE MANUFACTURER HAS DECIDED TO USE
SENECA USES THE "1 BASED" CONVENTION FOR ITS PRODUCTS

27.1. NUMBERING OF MODBUS ADDRESSES WITH "0-BASED" CONVENTION

The numbering is:

HOLDING REGISTER MODBUS ADDRESS (OFFSET)	*MEANING*
0	FIRST REGISTER
1	SECOND REGISTER
2	THIRD REGISTER
3	FOURTH REGISTER
4	FIFTH REGISTER

Therefore, the first register is at address 0.
In the following tables, this convention is indicated with ***"ADDRESS OFFSET".***

27.2. NUMBERING OF MODBUS ADDRESSES WITH "1 BASED" CONVENTION (STANDARD)

The numbering is that established by the Modbus consortium and is of the type:

HOLDING REGISTER MODBUS ADDRESS 4x	***MEANING***
40001	FIRST REGISTER
40002	SECOND REGISTER
40003	THIRD REGISTER
40004	FOURTH REGISTER
40005	FIFTH REGISTER

This convention is indicated with ***"ADDRESS 4x"*** since a 40000 is added to the address so that the first Modbus register is 40001.
A further convention is also possible where the number 4 is omitted in front of the register address:

HOLDING MODBUS ADDRESS WITHOUT 4x	***MEANING***
1	FIRST REGISTER
2	SECOND REGISTER
3	THIRD REGISTER
4	FOURTH REGISTER
5	FIFTH REGISTER

27.3. BIT CONVENTION WITHIN A MODBUS HOLDING REGISTER

A Modbus Holding Register consists of 16 bits with the following convention:

BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0

For instance, if the value of the register in decimal is
12300
the value 12300 in hexadecimal is:
0x300C

the hexadecimal 0x300C in binary value is:
11 0000 0000 1100

So, using the above convention, we get:

BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
0	0	1	1	0	0	0	0	0	0	0	0	1	1	0	0

27.4. MSB and LSB BYTE CONVENTION WITHIN A MODBUS HOLDING REGISTER

A Modbus Holding Register consists of 16 bits with the following convention:

BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0

LSB Byte (Least Significant Byte) defines the 8 bits ranging from Bit 0 to Bit 7 included, we define MSB Byte (Most Significant Byte) the 8 bits ranging from Bit 8 to Bit 15 inclusive:

BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
BYTE MSB								BYTE LSB							

27.5. REPRESENTATION OF A 32-BIT VALUE IN TWO CONSECUTIVE MODBUS HOLDING REGISTERS

The representation of a 32-bit value in the Modbus Holding Registers is made using 2 consecutive Holding Registers (a Holding Register is a 16-bit register). To obtain the 32-bit value it is therefore necessary to read two consecutive registers:
For example, if register 40064 contains the 16 most significant bits (MSW) while register 40065 contains the least significant 16 bits (LSW), the 32-bit value is obtained by composing the 2 registers:

BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
40064 MOST SIGNIFICANT WORD															

BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
40065 LEAST SIGNIFICANT WORD															

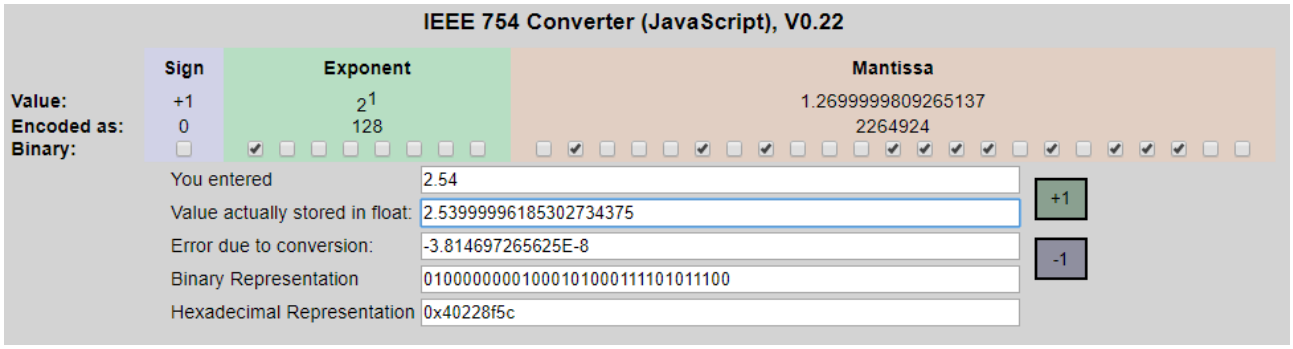
$$Value_{32bit} = Register_{LSW} + (Register_{MSW} * 65536)$$

In the reading registers it is possible to swap the most significant word with the least significant word, therefore it is possible to obtain 40064 as LSW and 40065 as MSW.

27.6. TYPE OF 32-BIT FLOATING POINT DATA (IEEE 754)

The IEEE 754 standard (https://en.wikipedia.org/wiki/IEEE_754) defines the format for representing floating point numbers.
As already mentioned, since it is a 32-bit data type, its representation occupies two 16-bit holding registers.
To obtain a binary/hexadecimal conversion of a floating point value it is possible to refer to an online converter at this address:

http://www.h-schmidt.net/FloatConverter/IEEE754.html



Using the last representation the value 2.54 is represented at 32 bits as:

0x4022 8F5C

Since we have 16-bit registers available, the value must be divided into MSW and LSW:

0x4022 (16418 decimal) are the 16 most significant bits (MSW) while 0x8F5C (36700 decimal) are the 16 least significant bits (LSW).

27.7. TYPE OF STRING DATA

The representation of a string in the Modbus Holding Registers is made using N consecutive Holding Registers (a Holding Register is a 16-bit register). To read/write the string, it is necessary to read/write multiple consecutive registers based on the size of the string itself.
The structure of the string in the registers is represented here:

- the first register contains the actual length of the string in bytes and therefore in characters (maximum length = 255)
- the subsequent registers contain 2 ASCII characters, with "Big Endian" convention the first character is in the MSB and the second in the LSB
- in case of an odd number of characters, the least significant byte of the register contains 0x00
- the excess registers (without characters) are at 0x0000

For example, the string tag whose size has been defined as 40 characters:

mobile.vodafone.it

actually occupies 21 registers (i.e. 40 characters + 1 register for the actual length of the string), note that the actual string size is 18 characters so the first register will be 0x0012 = 18 decimal:

reg[0]=0x0012
reg[1]=0x6D6F
reg[2]=0x6269
reg[3]=0x6C65
reg[4]=0x2E76
reg[5]=0x6F64
reg[6]=0x6166
reg[7]=0x6F6E
reg[8]=0x652E
reg[9]=0x6974
reg[10]=0x0000
reg[11]=0x0000
...
reg[19]=0x0000
reg[20]=0x0000