USER MANUAL

MULTIPROTOCOL “KEY” GATEWAYS SERIES

PROFINET IO / ETHERNET/IP - MODBUS RTU&TCP GATEWAYS

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
16/12/2022	0	First revision for new dual core CPU Aligned with firmware 117 revision	MM
26/04/2023	1	New operating modes introduced with firmware 204 revision	MM
27/04/2023	2	Various fixes	MM
21/07/2023	3	Corrected the report on chapter 8: MODBUS DIAGNOSTICS	AZ
24/07/2023	5	Added support for E-series Gateway	MM
02/02/2024	6	Changes for firmware 228 support of P-series Gateways, VARIOUS FIXES	MM
24/02/2025	7	Rewritten common parts with KEY FLEX devices Added chapter on description of LEDs Added excel template also for -E version	MM
23/07/2025	8	Added info on new web page by copy/paste TAG.	MM

TABLE OF CONTENTS

1. DESCRIPTION 6
1.1. PROFINET IO (GATEWAY -P) PROTOCOL 6
1.2. ETHERNET/IP (GATEWAY -E) PROTOCOL 6
1.1. FEATURES OF THE "KEY" SERIES COMMUNICATION PORTS 7

2. DEVICE HARDWARE REVISION 7

3. FLEX TECHNOLOGY FOR PROTOCOL CHANGE 8
3.1. CHANGING PROTOCOLS WITH THE SENECA DISCOVERY DEVICE SOFTWARE 9

4. LED MEANING 10
4.1. Z-KEY-P (PROFINET IO)MODEL LED 10
4.2. Z-KEY- E (ETHERNET/IP) MODEL LED 11
4.3. R-KEY-LT-P (PROFINET IO)MODEL LED 12
4.4. R-KEY-LT- E (ETHERNET/IP) MODEL LED 13
4.5. Z-KEY-2ETH-P (PROFINET IO)MODEL LED 14
4.6. Z-KEY-2ETH- E (ETHERNET/IP) MODEL LED 15

5. ETHERNET PORT 16

6. FIRMWARE UPDATE 16

7. OPERATING MODE 17
7.1. "-P" VERSIONS 17
7.1.1. PROFINET IO DEVICE / MODBUS MASTER GATEWAY 17
7.1.2. PROFINET IO DEVICE / MODBUS SLAVE GATEWAY 18
7.2. "-E" VERSIONS 19
7.2.1. GATEWAY ETHERNET/IP ADAPTER / MODBUS MASTER 19

8. GATEWAY CONFIGURATION 20
8.1. "-P" GATEWAY CONFIGURATION VIA EASY SETUP 2 AND TIA PORTAL 20
8.1.1. "PROFINET IO - MODBUS MASTER GATEWAY" CONFIGURATION 20
8.1.2. "PROFINET IO - MODBUS SLAVE GATEWAY" CONFIGURATION 38
8.2. "-E" GATEWAY CONFIGURATION WITH WEBSERVER AND STUDIO 5000 LOGIX DESIGNER ® SOFTWARE 55

9. GATEWAY WEBSERVERS 63
9.1. "-P" GATEWAY WEBSERVER 63
9.1.1. WEBSERVER MODE AND PROFINET MODE 63
9.1.2. MANUAL PROCEDURE FOR SWITCHING FROM PROFINET MODE TO WEBSERVER MODE AND VICE VERSA 63
9.1.3. STEP BY STEP GUIDE FOR THE FIRST ACCESS TO THE WEBSERVER 64
9.1.4. WEBSERVER DEVICE CONFIGURATION 65
9.1.5. WEBSERVER SECTIONS 66

9.1.6. "STATUS" SECTION 67
9.1.7. "SETUP" SECTION 67
9.1.8. SAVING a configuration on a file 70
9.1.9. Importing a configuration from a file 71
9.1.10. "COMMANDS/TAGS" SECTION (ONLY FOR PROFINET IO / MODBUS MASTER GATEWAY MODE) 72
9.1.11. "I/O MAPPING" SECTION 73
9.1.12. "FIRMWARE UPDATE" SECTION 73
9.1.13. "DATABASE UPDATE" SECTION 73
9.1.14. SERIAL "SERIAL TRAFFIC MONITOR" 74
9.2. "-E" GATEWAY WEBSERVER 74
9.2.1. STEP BY STEP GUIDE FOR THE FIRST ACCESS TO THE WEBSERVER 74
9.2.2. WEBSERVER DEVICE CONFIGURATION 75
9.2.3. WEBSERVER SECTIONS 76
9.2.4. "STATUS" SECTION 77
9.2.5. "SETUP" SECTION 77
9.2.6. SAVING a configuration on a file 81
9.2.7. Importing a configuration from a file 81
9.2.8. "COMMANDS/TAGS" SECTION 82
9.2.9. "I/O MAPPING" SECTION 83
9.2.10. "FIRMWARE UPDATE" SECTION 83
9.2.11. "DATABASE UPDATE" SECTION 83
9.2.12. SERIAL "SERIAL TRAFFIC MONITOR" 83

10. RESETTING THE DEVICE TO ITS FACTORY CONFIGURATION 84

11. EXCEL TEMPLATE 85

12. SUPPORTED MODBUS COMMUNICATION PROTOCOLS 85
12.1. SUPPORTED MODBUS FUNCTION CODES 85

13. MODBUS DIAGNOSTICS 86

14. INFORMATION ABOUT MODBUS REGISTERS 87
14.1. NUMBERING OF "0-BASED" OR "1-BASED" MODBUS ADDRESSES 87
14.2. NUMBERING OF MODBUS ADDRESSES WITH "0-BASED" CONVENTION 88
14.3. NUMBERING OF MODBUS ADDRESSES WITH "1 BASED" CONVENTION (STANDARD) 88
14.4. BIT CONVENTION WITHIN A MODBUS HOLDING REGISTER 89
14.5. MSB and LSB BYTE CONVENTION WITHIN A MODBUS HOLDING REGISTER 89
14.6. REPRESENTATION OF A 32-BIT VALUE IN TWO CONSECUTIVE MODBUS HOLDING REGISTERS 90
14.7. TYPE OF 32-BIT FLOATING POINT DATA (IEEE 754) 91

1. DESCRIPTION

The Z-KEY-P, R-KEY-LT-P, Z-KEY-2ETH-P products allow to convert data coming from the Modbus serial bus or Modbus TCP-IP Ethernet into the Profinet IO bus or vice versa.

The Z-KEY- E, R-KEY-LT- E, Z-KEY-2ETH- E products allow to convert data coming from the Modbus serial bus or Modbus TCP-IP Ethernet into the Ethernet IP bus or vice versa.

1.1. PROFINET IO (GATEWAY -P) PROTOCOL

PROTOCOL	
Type of protocol	Profinet IO, Class A Device, Cyclic Real-time (RT) and Acyclic Data

MEMORY	
Memory size	In Gateway Master and Gateway Slave modes: 1200 bytes max in reading and 1200 bytes max in writing (-P versions) (20 slots max)

1.2. ETHERNET/IP (GATEWAY -E) PROTOCOL

PROTOCOL	
Type of protocol	ETHERNET/IP Adapter, 1 connection read/write

MEMORY	
Memory size	512 bytes max in reading and 512 bytes max in writing (-E versions)

1.1. FEATURES OF THE "KEY" SERIES COMMUNICATION PORTS

PRODUCT	*ETHERNET PORTS No.*	*NO. CONFIGURABLE RS232/RS485 SERIAL PORTS*	*SECOND RS485 SERIAL PORT*	*ISOLATED SERIAL PORTS*	*PROTOCOL*
Z-KEY-P	1	1	Yes	Yes, both ports	PROFINET-IO
R-KEY-LT-P	1	1	NO	NO	PROFINET-IO
Z-KEY-2ETH-P	2	1	Yes	Yes, both ports	PROFINET-IO
Z-KEY-E	1	1	Yes	Yes, both ports	ETHERNET/IP
R-KEY-LT-E	1	1	NO	NO	ETHERNET/IP
Z-KEY-2ETH-E	2	1	Yes	Yes, both ports	ETHERNET/IP

2. *DEVICE HARDWARE REVISION*

With a view to continuous improvement, Seneca updates and makes the hardware of its devices increasingly more sophisticated. It is possible to know the hardware revision of a product via the label on the side of the device.
An example of an R-KEY-LT product label is the following:

The label also shows the firmware revision present in the device (in this case 2.0.1.0) at the time of sale, the hardware revision (in this case) is E00.
To improve performance or extend functionality, Seneca recommends updating the firmware to the latest available version (see the section dedicated to the product on www.seneca.it).

An internal Webserver is also available for configuration and display of values in real time.

3. FLEX TECHNOLOGY FOR PROTOCOL CHANGE

Starting from the hardware revision indicated in the following table, the KEY series devices include Flex technology.

GATEWAY	FLEX TECHNOLOGY SUPPORTED BY HARDWARE REVISION
Z-KEY	"G00"
R-KEY-LT	"E00"
Z-KEY-2ETH	"C00"

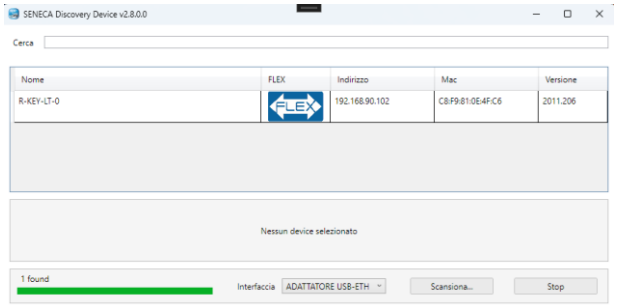
Flex allows you to change the combination of industrial communication protocols supported by the gateways at will from a list of available ones, the development is continuously updated, for a complete list refer to the page: https://www.seneca.it/flex/

Some examples of supported protocols are:

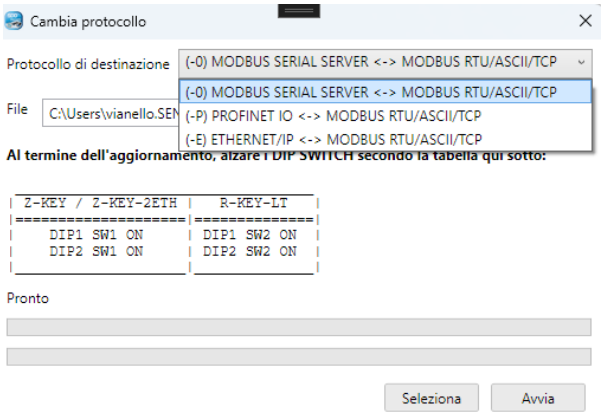
The gateway then becomes "universal" and compatible with Siemens or Rockwell or Schneider systems etc. without the need to purchase different hardware.

3.1. *CHANGING PROTOCOLS WITH THE SENECA DISCOVERY DEVICE SOFTWARE*

From revision 2.8 the Seneca Discovery Device software identifies the devices that support the “Flex” technology:



For example, in the case in the figure it is possible to press the “Change Protocol” button and select the destination protocol from those in the list:



At the end of the operation, bring (only at the first power-on) the dip switches 1 and 2 to “ON” to force the device to default (see also the chapter “RESETTING THE DEVICE TO ITS FACTORY CONFIGURATION”).

Always refer to the user manual of the communication protocol installed in the device by downloading it from the Seneca website.

4. LED MEANING

The devices are equipped with LEDs whose meaning is as follows:

4.1. Z-KEY-P (PROFINET IO)MODEL LED

LED	STATUS
PWR	***Steady on***: device powered and in Profinet IO mode ***Flashing:*** Device powered and in Webserver mode ***Off***: device not powered
COM	***Flashing:*** PLC communication active ***Off:*** PLC communication not active
TX1	***Flashing:*** data transmission on serial port #1 ***Off:*** no transmission on serial port #1
RX1	***Flashing:*** data reception on serial port #1 ***Steady on:*** check wiring on serial port #1 ***Off:*** no reception on serial port #1
TX2	***Flashing:*** data transmission on serial port #2 ***Off:*** no transmission on serial port #2
RX2	***Flashing:*** data reception on serial port #2 ***Steady on:*** check wiring on serial port #2 ***Off:*** no reception on serial port #2
ETH ACT (GREEN)	***Flashing:*** presence of data on ethernet port ***Steady on:*** ethernet port connected but no data present ***Off:*** check wiring of the ethernet port
ETH LNK (YELLOW)	***Steady on:***ethernet cable connected ***Off:*** check the wiring of the ethernet port

4.2. Z-KEY- E (ETHERNET/IP) MODEL *LED*

LED	*STATUS*
PWR	***Steady on***: device powered and IP address set ***Flashing***: IP address not yet set ***Off***: device not powered
COM	***Flashing:*** PLC communication active ***Off:*** PLC communication not active
TX1	***Flashing:*** data transmission on serial port #1 ***Off:*** no transmission on serial port #1
RX1	***Flashing:*** data reception on serial port #1 ***Steady on:*** check wiring on serial port #1 ***Off:*** no reception on serial port #1
TX2	***Flashing:*** data transmission on serial port #2 ***Off:*** no transmission on serial port #2
RX2	***Flashing:*** data reception on serial port #2 ***Steady on:*** check wiring on serial port #2 ***Off:*** no reception on serial port #2
ETH ACT (GREEN)	***Flashing:*** presence of data on ethernet port ***Steady on:*** ethernet port connected but no data present ***Off:*** check wiring of the ethernet port
ETH LNK (YELLOW)	***Steady on:***ethernet cable connected ***Off:*** check the wiring of the ethernet port

4.3. *R-KEY-LT-P (PROFINET IO)MODEL LED*

LED	*STATUS*
PWR	***Steady on***: device powered and in Profinet IO mode ***Flashing:*** Device powered and in Webserver mode ***Off***: device not powered
COM	***Flashing:*** PLC communication active ***Off:*** PLC communication not active
TX	***Flashing:*** data transmission on serial port ***Off:*** no transmission on serial port
RX	***Flashing:*** data reception on serial port ***Steady on:*** check wiring on serial port ***Off:*** no reception on serial port
ETH ACT (GREEN)	***Flashing:*** presence of data on ethernet port ***Steady on:*** ethernet port connected but no data present ***Off:*** check wiring of the ethernet port
ETH LNK (YELLOW)	***Steady on:***ethernet cable connected ***Off:*** check the wiring of the ethernet port

4.4. *R-KEY-LT- E (ETHERNET/IP) MODEL LED*

LED	***STATUS***
PWR	***Steady on***: device powered and IP address set ***Flashing***: IP address not yet set ***Off***: device not powered
COM	***Flashing:*** PLC communication active ***Off:*** PLC communication not active
TX	***Flashing:*** data transmission on serial port ***Off:*** no transmission on serial port
RX	***Flashing:*** data reception on serial port ***Steady on:*** check wiring on serial port ***Off:*** no reception on serial port
ETH ACT (GREEN)	***Flashing:*** presence of data on ethernet port ***Steady on:*** ethernet port connected but no data present ***Off:*** check wiring of the ethernet port
ETH LNK (YELLOW)	***Steady on:***ethernet cable connected ***Off:*** check the wiring of the ethernet port

4.5. *Z-KEY-2ETH-P (PROFINET IO)MODEL LED*

LED	*STATUS*
PWR	***Steady on***: device powered and in Profinet IO mode ***Flashing:*** Device powered and in Webserver mode ***Off***: device not powered
COM	***Flashing:*** PLC communication active ***Off:*** PLC communication not active
TX1	***Flashing:*** data transmission on serial port #1 ***Off:*** no transmission on serial port #1
RX1	***Flashing:*** data reception on serial port #1 ***Steady on:*** check wiring on serial port #1 ***Off:*** no reception on serial port #1
TX2	***Flashing:*** data transmission on serial port #2 ***Off:*** no transmission on serial port #2
RX2	***Flashing:*** data reception on serial port #2 ***Steady on:*** check wiring on serial port #2 ***Off:*** no reception on serial port #2
ET1	***Flashing:*** presence of data on ethernet port #1 ***Steady on:*** ethernet port #1 connected but no data present ***Off:*** check wiring of ethernet port #1
ET2	***Flashing:*** presence of data on ethernet port #2 ***Steady on:*** ethernet port #2 connected but no data present ***Off:*** check wiring of ethernet port #2

4.6. *Z-KEY-2ETH- E (ETHERNET/IP) MODEL LED*

LED	*STATUS*
PWR	***Steady on***: device powered and IP address set ***Flashing***: IP address not yet set ***Off***: device not powered
COM	***Flashing:*** PLC communication active ***Off:*** PLC communication not active
TX1	***Flashing:*** data transmission on serial port #1 ***Off:*** no transmission on serial port #1
RX1	***Flashing:*** data reception on serial port #1 ***Steady on:*** check wiring on serial port #1 ***Off:*** no reception on serial port #1
TX2	***Flashing:*** data transmission on serial port #2 ***Off:*** no transmission on serial port #2
RX2	***Flashing:*** data reception on serial port #2 ***Steady on:*** check wiring on serial port #2 ***Off:*** no reception on serial port #2
ET1	***Flashing:*** presence of data on ethernet port #1 ***Steady on:*** ethernet port #1 connected but no data present ***Off:*** check wiring of ethernet port #1
ET2	***Flashing:*** presence of data on ethernet port #2 ***Steady on:*** ethernet port #2 connected but no data present ***Off:*** check wiring of ethernet port #2

5. ETHERNET PORT

The factory configuration of the Ethernet port is:

STATIC IP: 192.168.90.101
SUBNET MASK: 255.255.255.0
GATEWAY: 192.168.90.1

Multiple devices must not be inserted on the same network with the same static IP.

⚠ ATTENTION!

DO NOT CONNECT 2 OR MORE FACTORY-CONFIGURED DEVICES ON THE SAME NETWORK, OR THE DEVICE WILL NOT WORK
(CONFLICT OF IP ADDRESSES 192.168.90.101)

6. FIRMWARE UPDATE

In order to improve, add or optimize the functions of the product, Seneca releases firmware updates on the device section on the www.seneca.it website

The firmware update is made using the appropriate command on the Easy Setup2 software or the webserver.

⚠ ATTENTION!

UPDATING THE FIRMWARE OF THE PROFINET IO DEVICES FROM A 1xx TO A 2xx REVISION WILL LOSE THE CONFIGURATION.
ON THE SENECA WEBSITE THERE IS AN EXCEL TEMPLATE THAT IMPORTS A CONFIGURATION OF THE TAGS CARRIED OUT WITH A 1xx FIRMWARE AND CONVERTS IT IN THE NEW MODE "GATEWAY PROFINET IO MODBUS MASTER" OF THE 2xx FIRMWARE REVISIONS
FOR MORE INFO REFER TO THE TEMPLATE ITSELF

⚠ ATTENTION!

NOT TO DAMAGE THE DEVICE DO NOT REMOVE THE POWER SUPPLY DURING THE FIRMWARE UPDATE OPERATION.

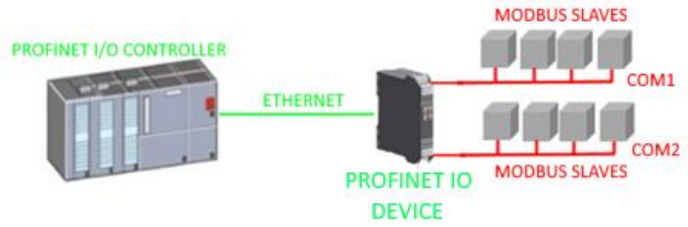
7. OPERATING MODE

7.1. *"-P" VERSIONS*

The Gateway allows you to operate in 3 different modes:
PROFINET IO DEVICE / MODBUS MASTER GATEWAY
PROFINET IO DEVICE / MODBUS SLAVE GATEWAY
GATEWAY WITH PORT#1 AND PORT#2 MASTER TAG.

7.1.1. PROFINET IO DEVICE / MODBUS MASTER GATEWAY

This operating mode is the most used and allows you to connect a Profinet IO PLC controller with Modbus RTU/ASCII Slave I/O devices:

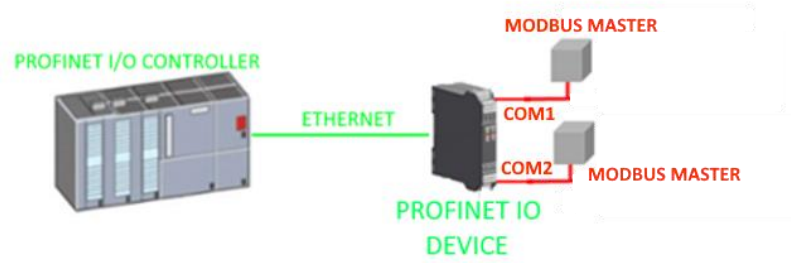


The Gateway, in the serial part, works as a Modbus master device and in the Ethernet part as a Profinet IO Device.
Modbus requests (read or write commands) are configured in the device and an GSDML file is automatically generated.
Once this file is imported into the PLC development software (e.g. TIA PORTAL) all configured IO will be accessible without any other configuration.

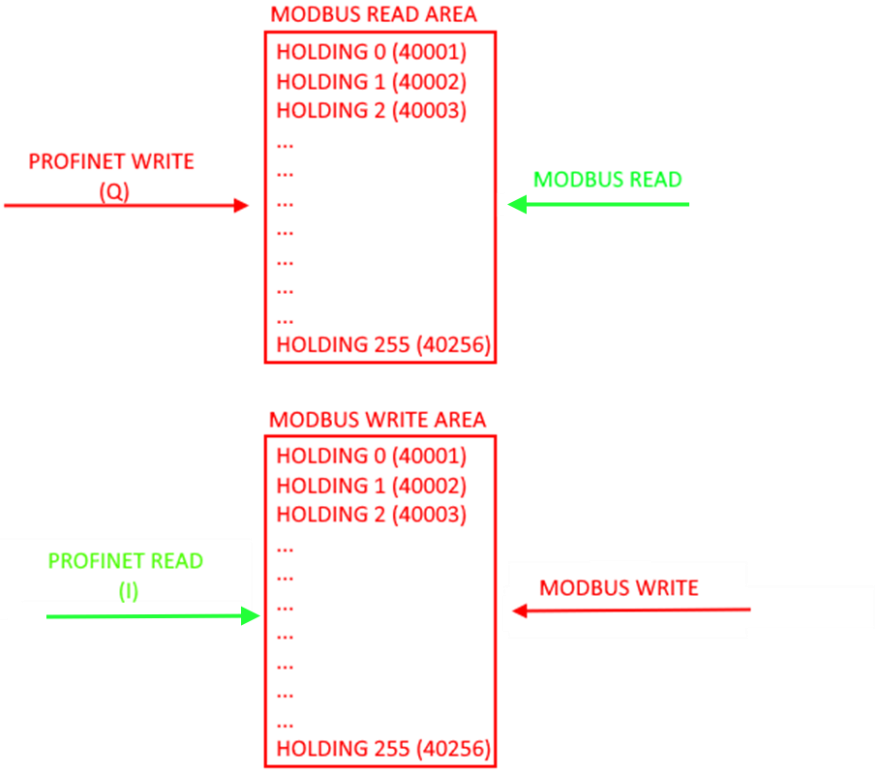
In addition to serial devices it is also possible to connect up to 3 Modbus TCP-IP servers.

7.1.2. PROFINET IO DEVICE / MODBUS SLAVE GATEWAY

This operating mode allows you to connect a Profinet IO PLC controller with a maximum of 1 or 2 devices (based on the number of serial ports available in the gateway) of the Modbus RTU/ASCII Master type (typically of the PLCs):



The gateway provides two different areas of 512 bytes for reading and 512 bytes for writing.
Bytes are available from Modbus Holding Register 0 to Holding Register 255 inclusive.
The “Modbus Read Area” is only readable by Modbus and only writable by Profinet.
The "Modbus Write Area" is only writable by Modbus and only readable by Profinet.



⚠ ATTENTION!

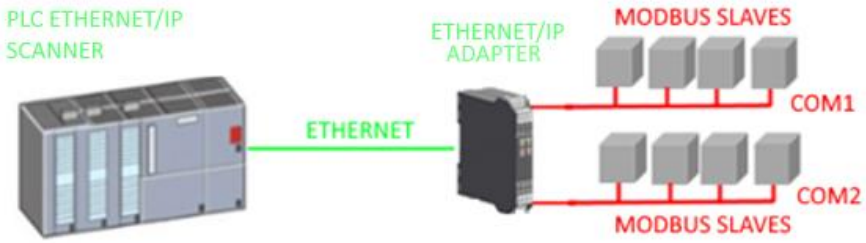
THE GATEWAY CREATES TWO DIFFERENT MODBUS AREAS, ONE FOR READING AND ONE FOR WRITING.
FOR EXAMPLE IF YOU WRITE BYTES FROM MODBUS THESE WILL END UP IN THE WRITING AREA AND THEN THEY WILL NOT BE READABLE BY THE MODBUS ITSELF

7.2. "-E" VERSIONS

The Gateway allows you to operate in the following mode:
GATEWAY ETHERNET/IP ADAPTER / MODBUS MASTER

7.2.1. GATEWAY ETHERNET/IP ADAPTER / MODBUS MASTER

This operating mode allows you to connect an ETHERNET/IP PLC scanner with Modbus RTU/ASCII Slave I/O devices



The Gateway, in the serial part, works as a Modbus master device and in the Ethernet part as a Ethernet /IP Adapter.
Modbus requests (read or write commands) are configured in the device and an EDS file is automatically generated.
Once this file is imported into the PLC development software (e.g. Rockwell STUDIO 5000) all configured IO will be accessible without any other configuration.

In addition to serial devices it is also possible to connect up to 3 Modbus TCP-IP servers.

8. GATEWAY CONFIGURATION

8.1. "-P" GATEWAY CONFIGURATION VIA EASY SETUP 2 AND TIA PORTAL

The easiest method to configure the gateway is through the Easy Setup2 software.
For more information, refer to the help in the software.

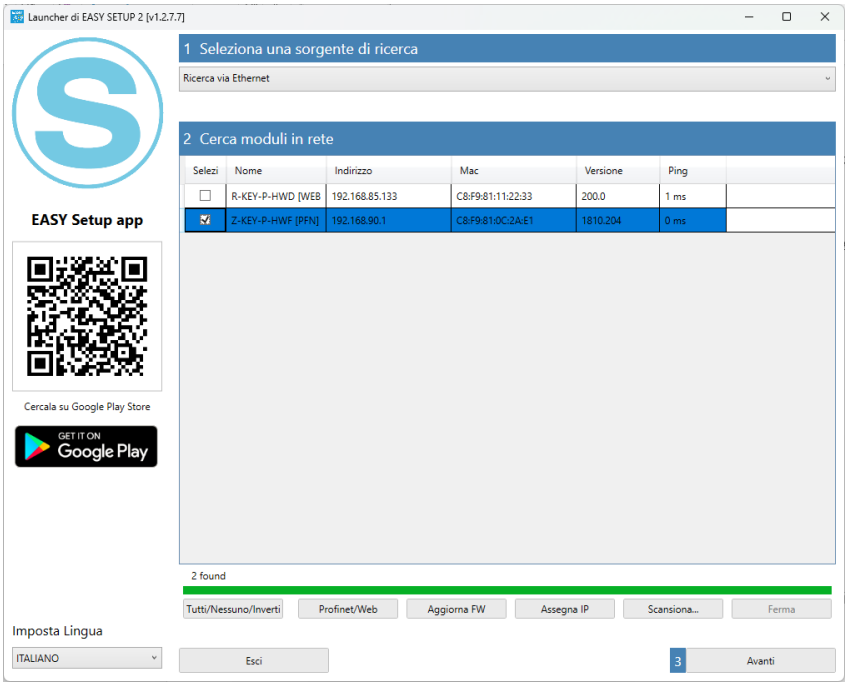
8.1.1. "PROFINET IO - MODBUS MASTER GATEWAY" CONFIGURATION

You want to connect a Siemens™ PLC to two Seneca Modbus RTU slave devices:
Z-10-D-IN (SLAVE STATION ADDRESS 1)
Z-10-D-OUT (SLAVE STATION ADDRESS 2).

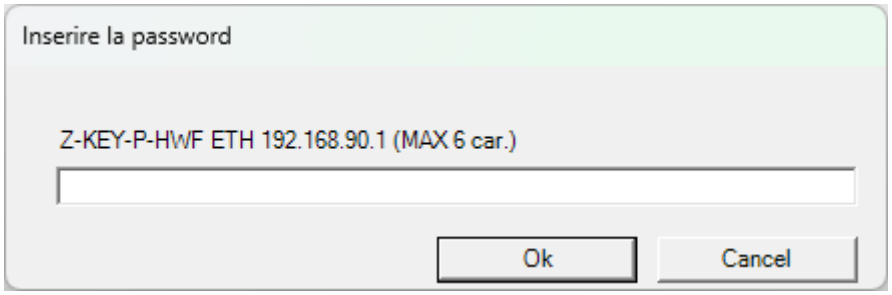
In the example we will use the Z-KEY-P product (the steps are exactly the same for the other R-KEY-LT-P and Z-KEY-2ETH-P devices).

The 10 digital inputs of the Z-10-D-IN are from coil address 1 to coil address 10 of station address #1
The 10 digital outputs of the Z-10-D-OUT are from coil address 1 to coil 10 of Station Address #2

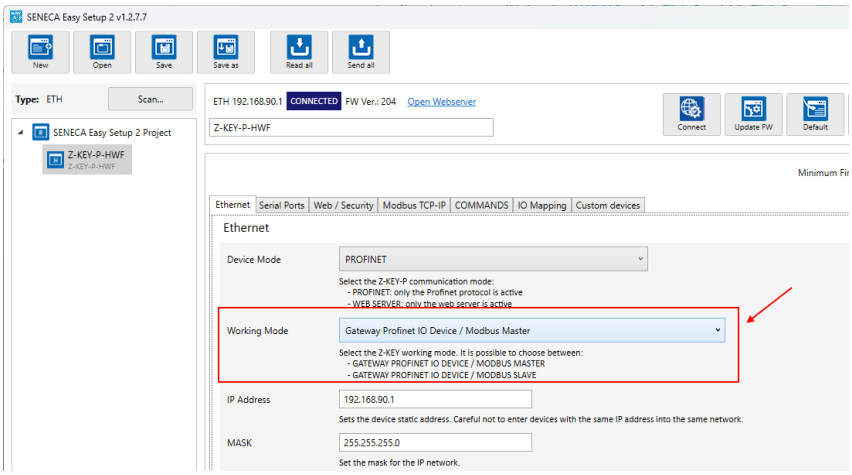
First we disconnect the PLC from the Ethernet network. Now we use the Easy Setup 2 software selecting the Z-KEY-P product (with SCAN or in manual entry):



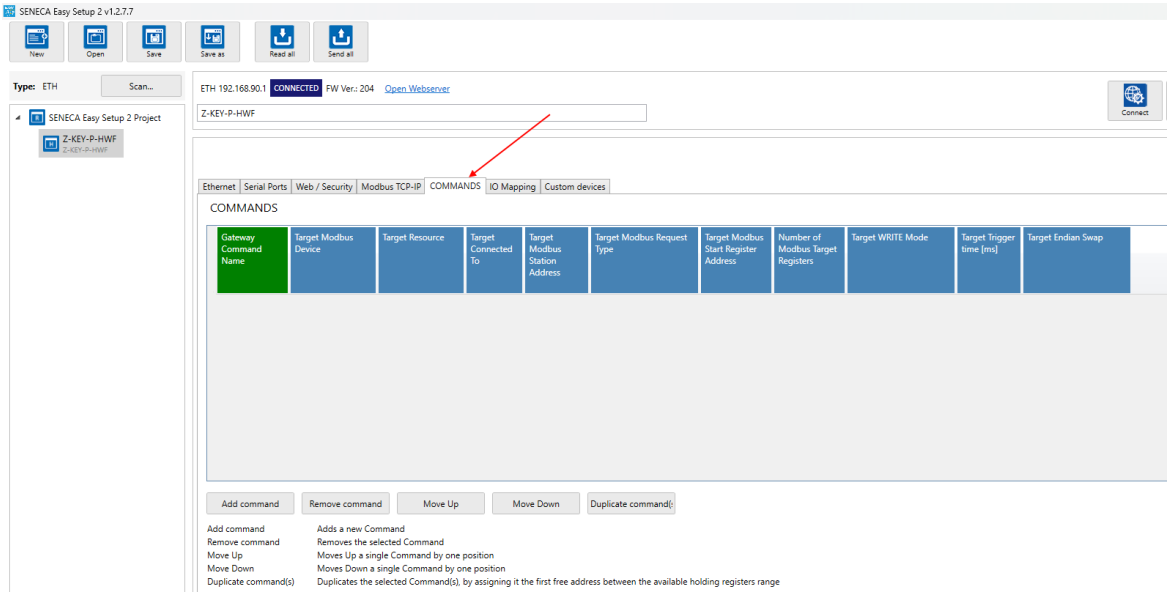
At this point the device access password is requested (default: admin):



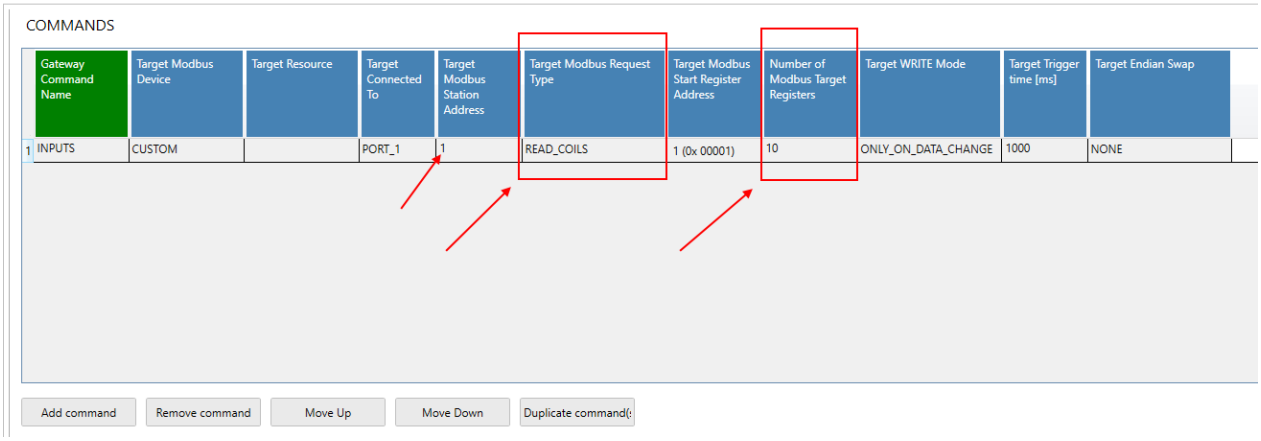
Once the password has been entered, select the Profinet IO Device / Modbus Master mode Gateway:



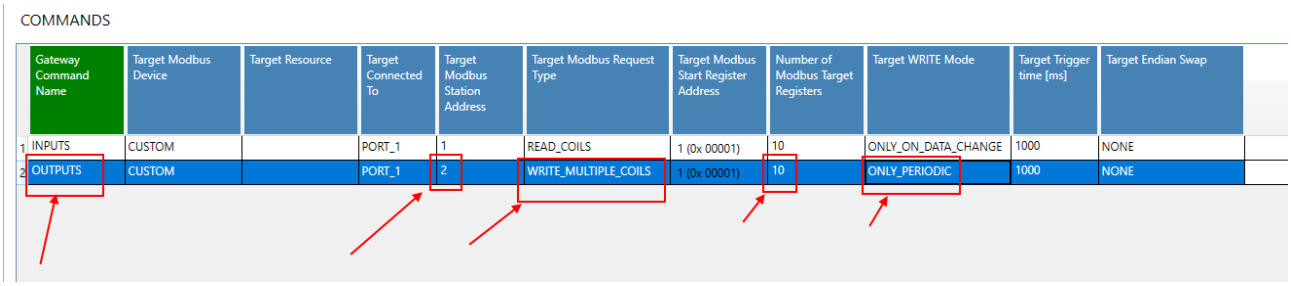
Now let's add the Modbus commands to acquire the inputs and write the outputs, select the COMMANDS section:



We add the reading of 10 coil registers relating to the 10 digital inputs of Z-10-D-IN:

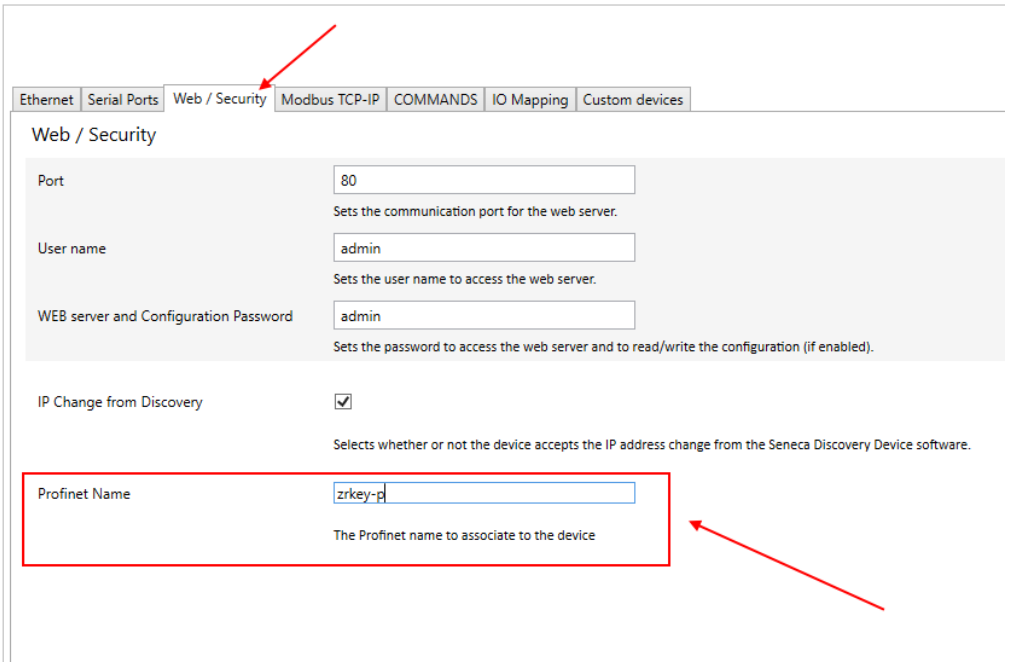


Now let's add the writing of 10 coil registers related to the 10 digital outputs of Z-10-D-OUT:

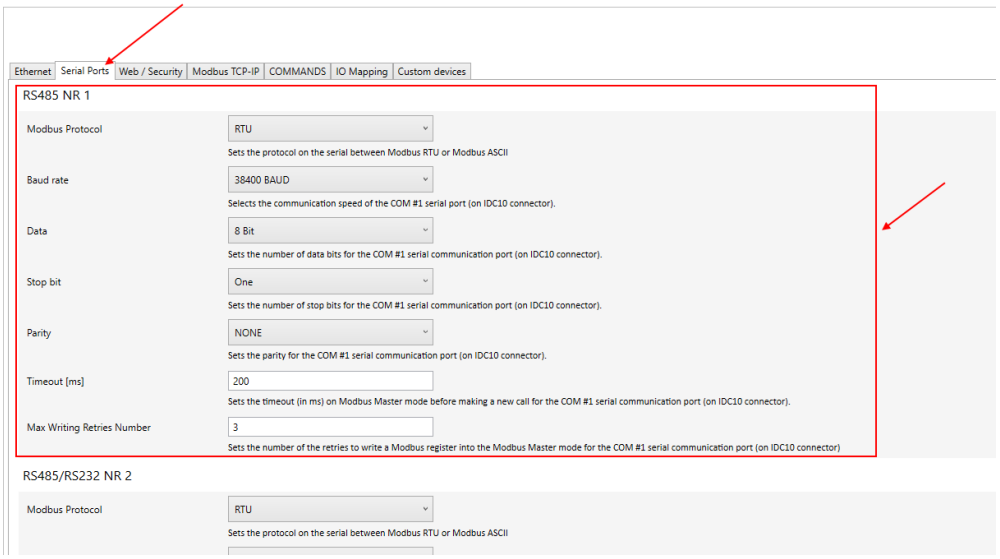


We set the writings in "Only Periodic" so they will always be performed every 1000 ms.

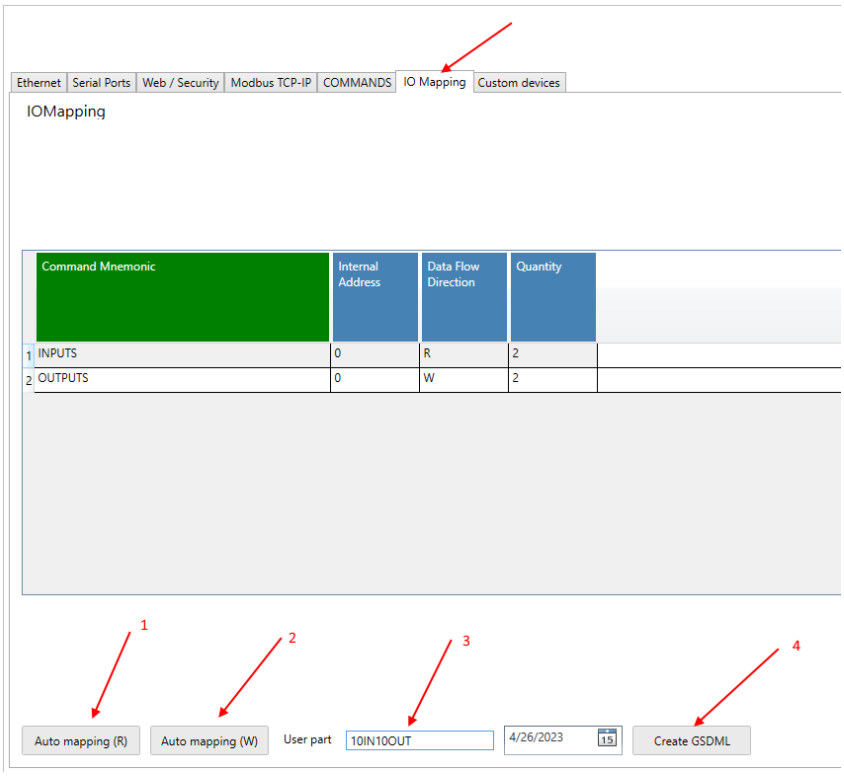
Enter the profinet name of the device:



Let's verify that serial port 1 is configured correctly for slave devices:



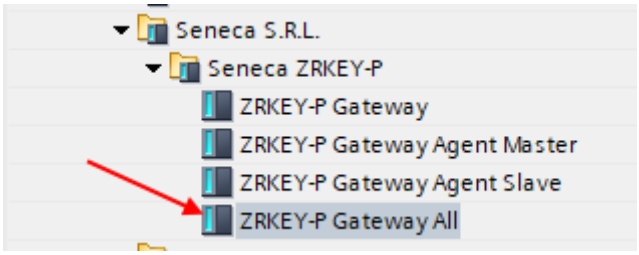
At this point we export the GSDML file from the "IO Mapping" section:



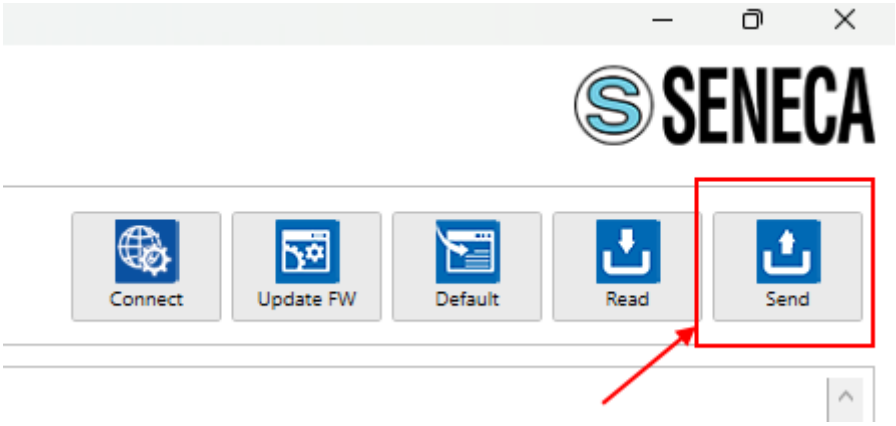
1 We press the button to calculate the offsets of the readings
2 We press the button to calculate the write offsets
3 We enter a name to recognize the GSDML file
4 We export the GSDML file

⚠ ATTENTION!

IT IS ALSO POSSIBLE TO DOWNLOAD A GSDML FILE (Gateway All) FROM THE WWW.SENECA.IT WEBSITE (IN THE SECTION RELATED TO PROFINET GATEWAYS) AND COMPOSE YOUR OWN CONFIGURATION FROM TIA PORTAL WITHOUT HAVING TO IMPORT THE FILE EACH TIME.

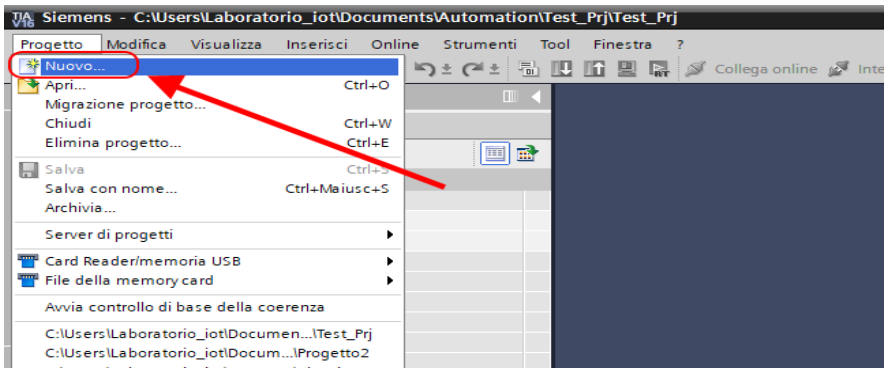


Now let's send the configuration to the device with the "send" button:

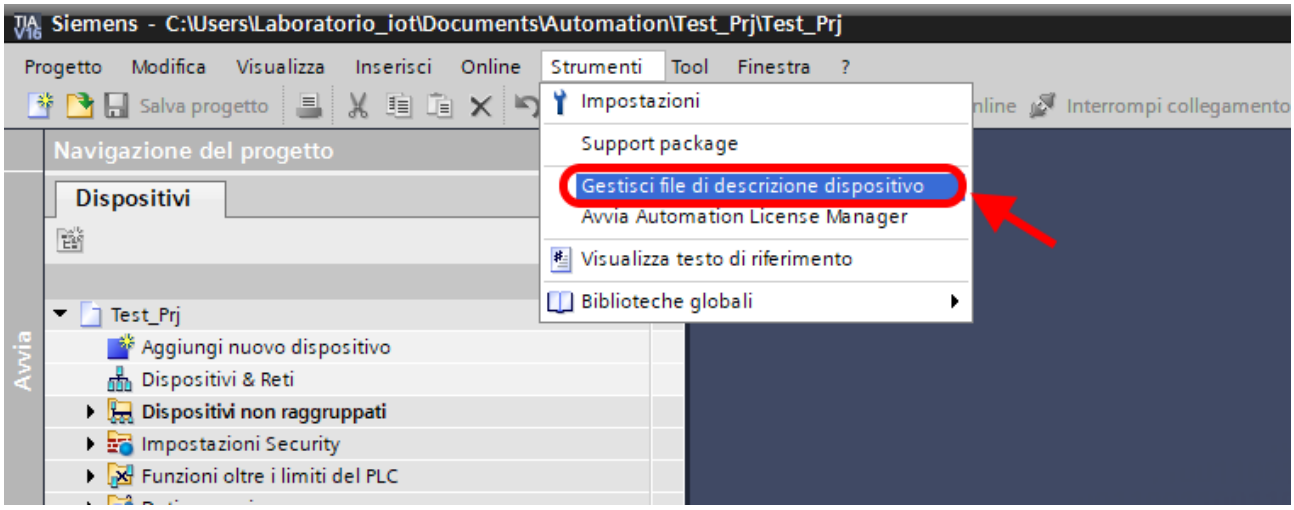


We can now move on to configuring the PLC via Tia Portal™:

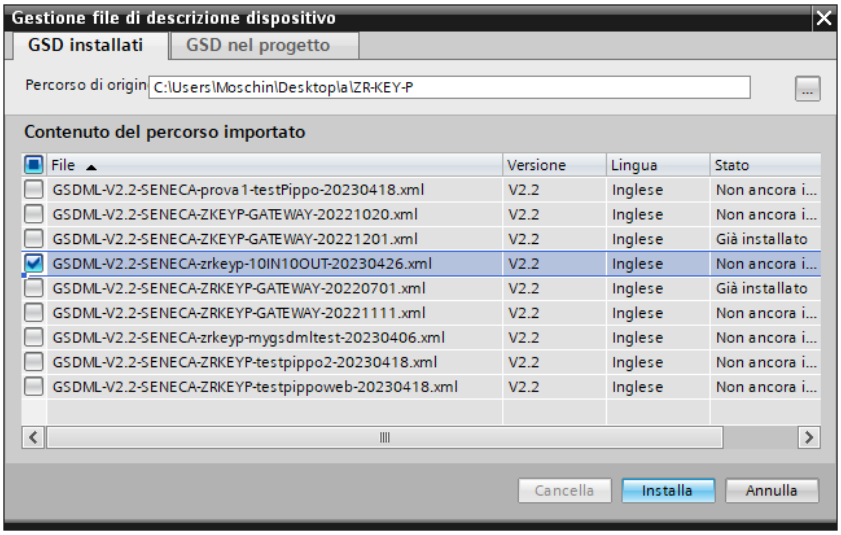
Creating a new project:



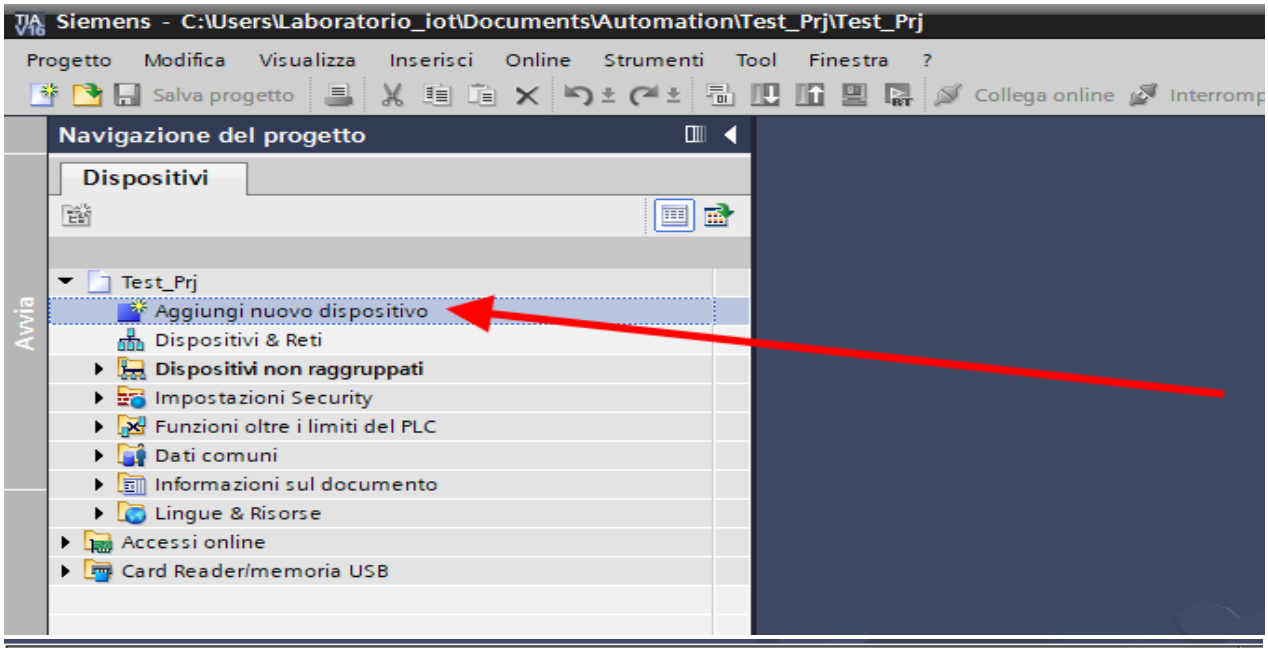
Install the GSD file of the Seneca product:

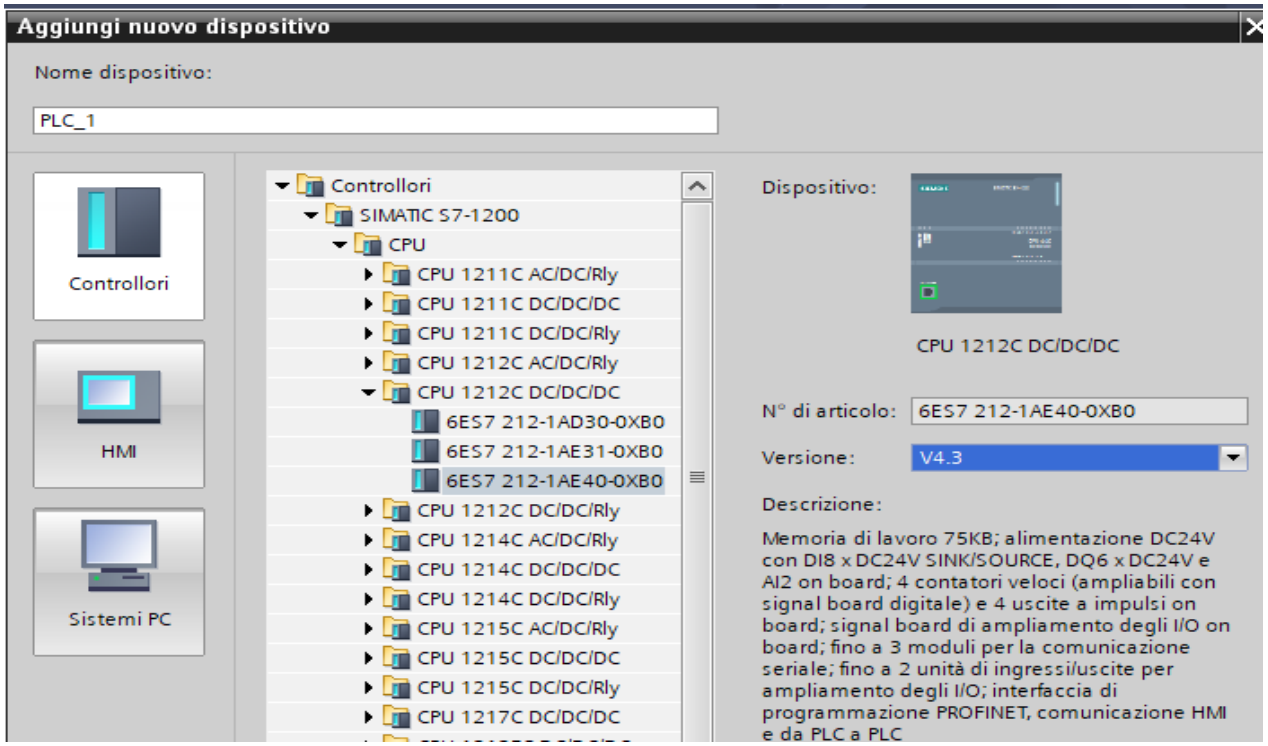


Point to the directory where we previously saved the GSDML file and press INSTALL:

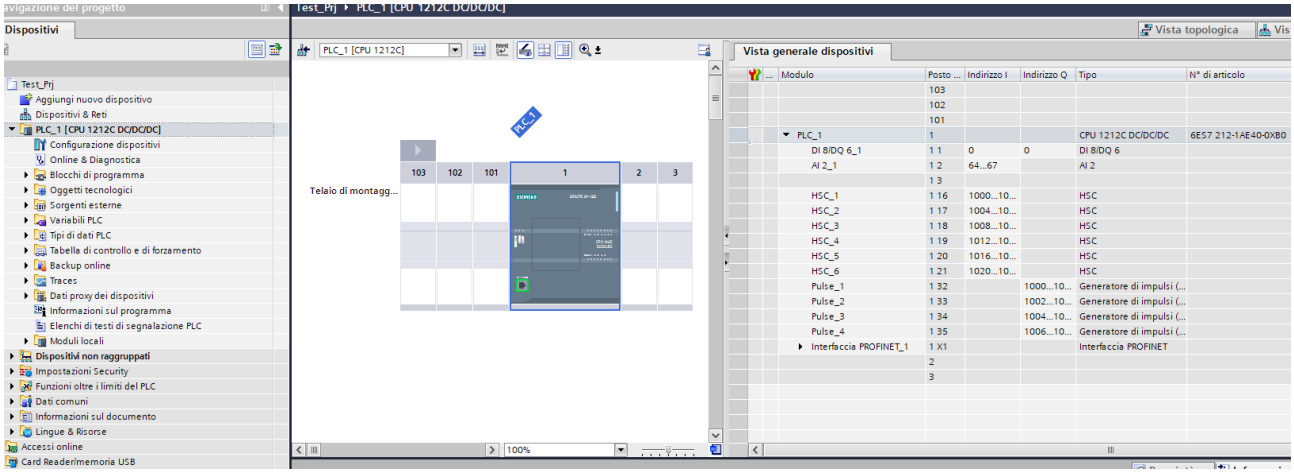


Now insert the Siemens PLC (in our example a SIEMATIC S7 1200), click on "Add new device":

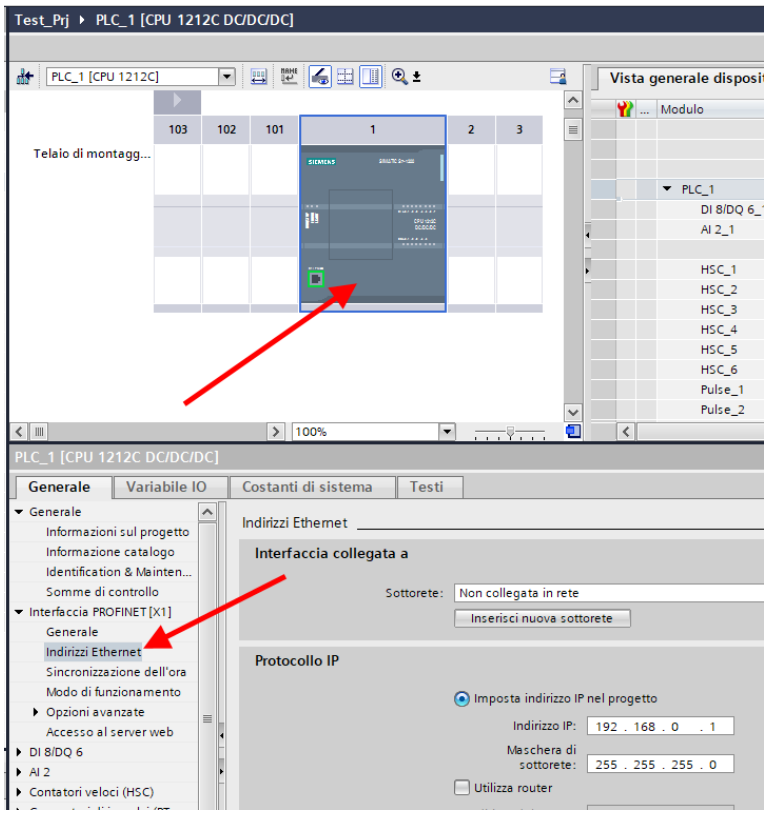




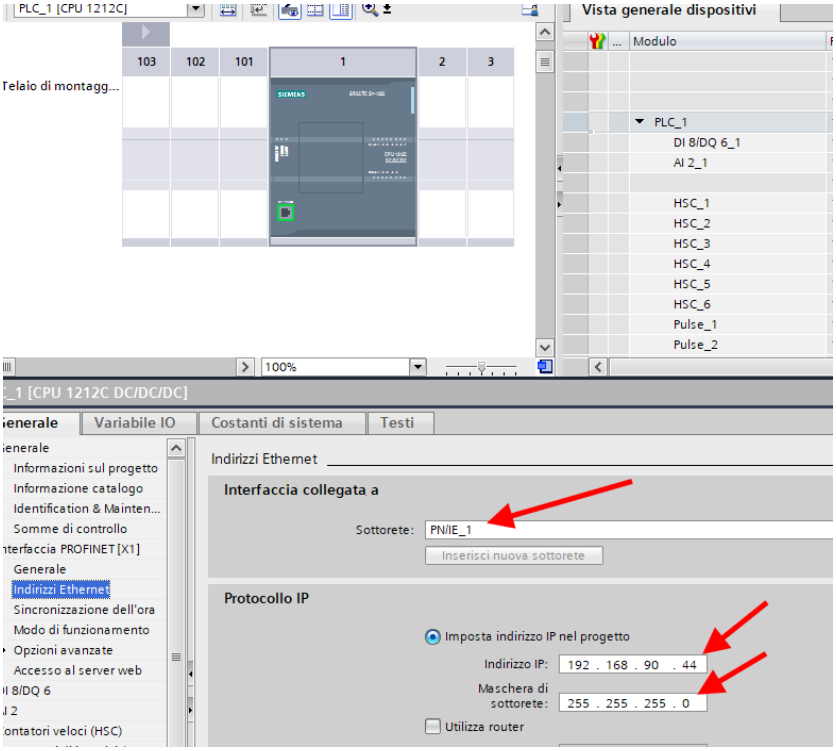
Confirm and the PLC will be added to the rack:



Now click on the PLC and select Profinet interface -> Ethernet addresses

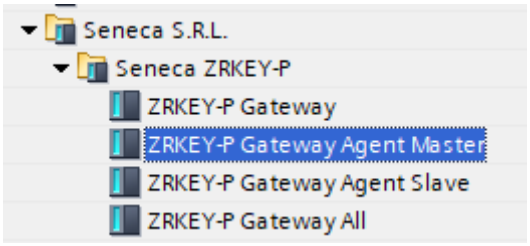


Set the IP you want for the PLC (in this case 192.168.90.44) and the PLC subnet:

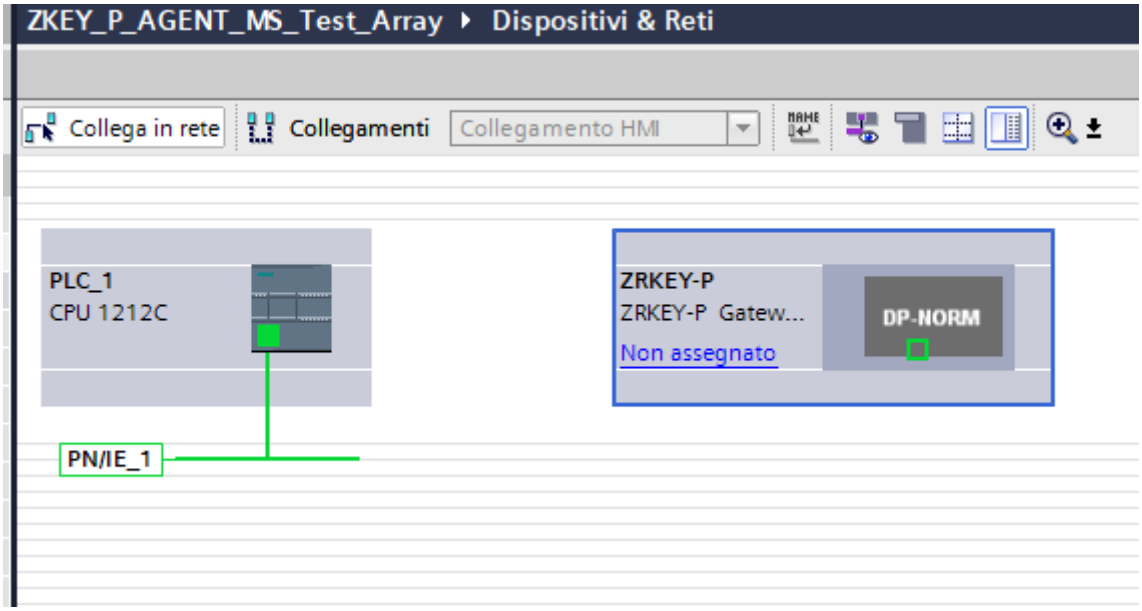


Move on to “devices and network” view:

On the right select "Hardware Catalogue" and then under "Additional Field Equipment" -> PROFINET IO -> GATEWAY -> Seneca SRL - ZR-KEY-P Gateway-> ZRKEY-P Gateway Agent Master

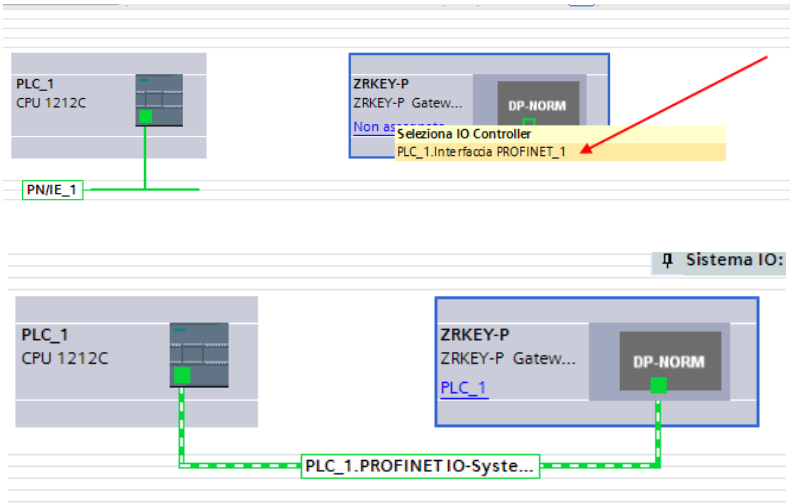


Drag the device to the network view:

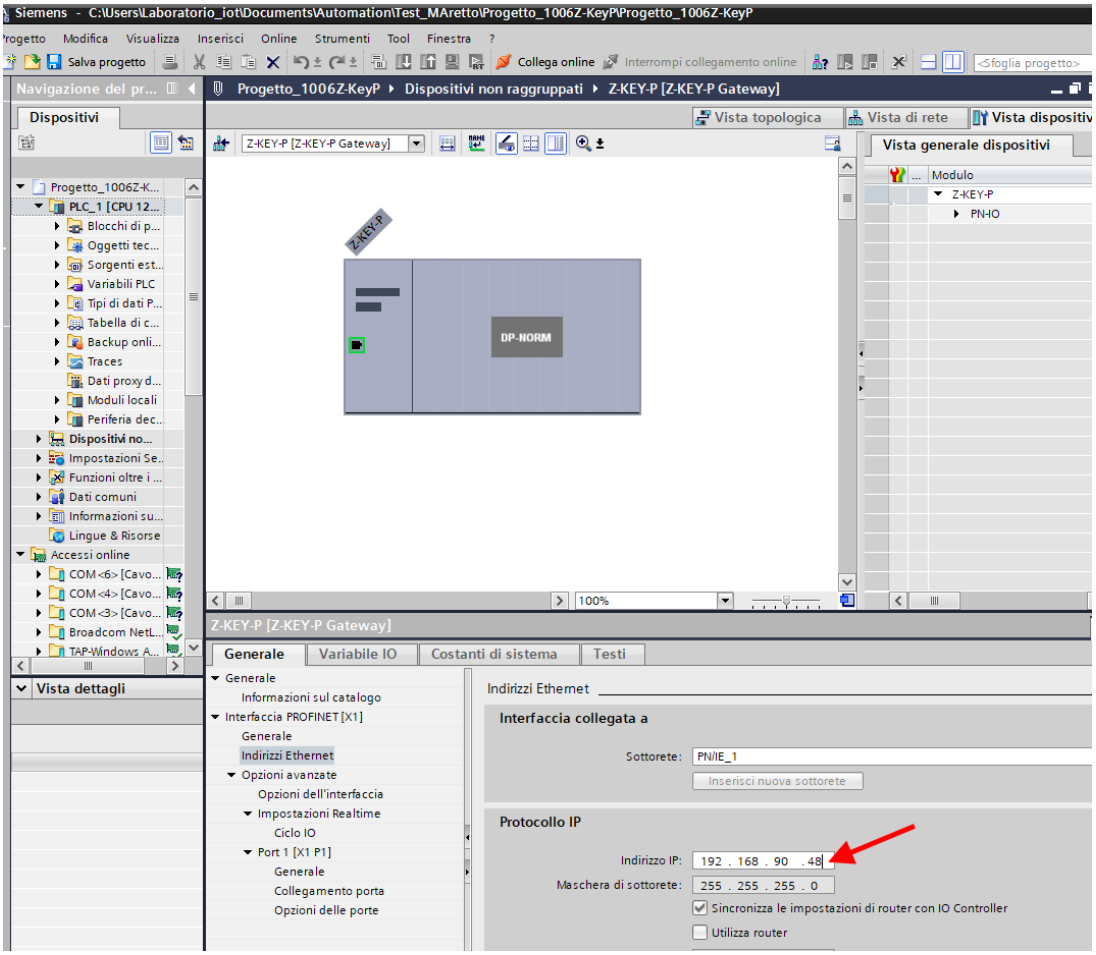


Now associate it with the PLC:

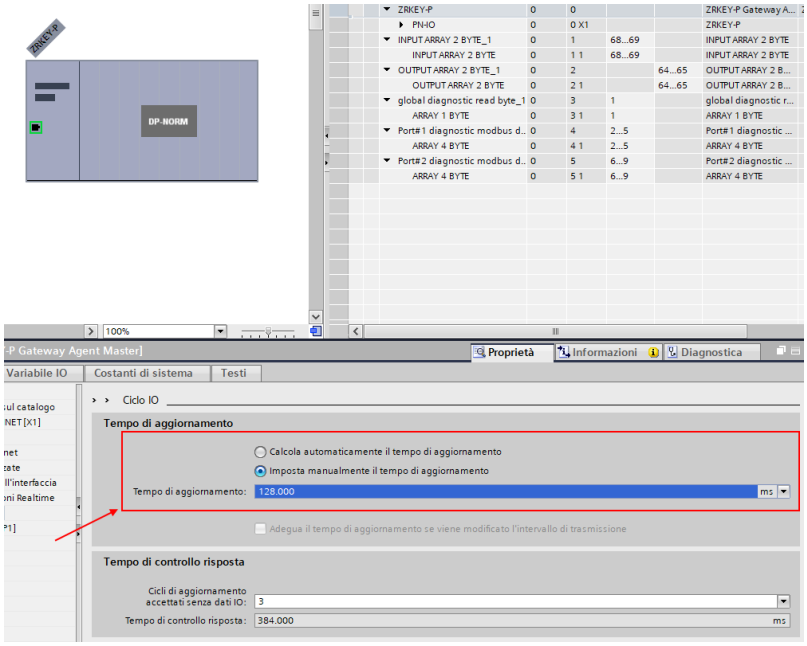
Click with the left mouse button on "Not assigned" and then select the PLC:



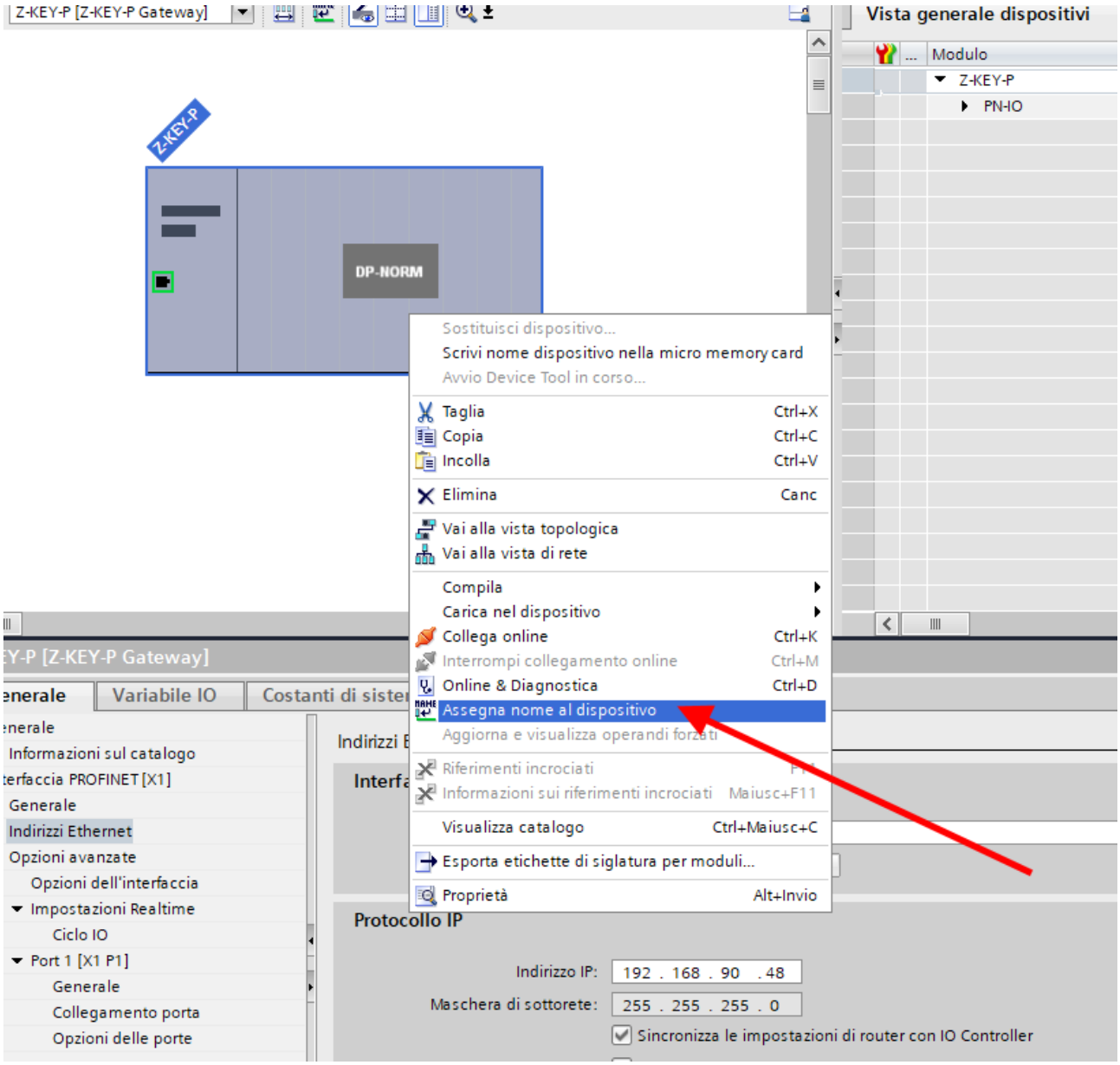
Click twice on the Seneca device and configure the IP address here too (for example 192.168.90.48) and the timing:



Depending on the project it is necessary to set the cycle time (typically 128 ms):

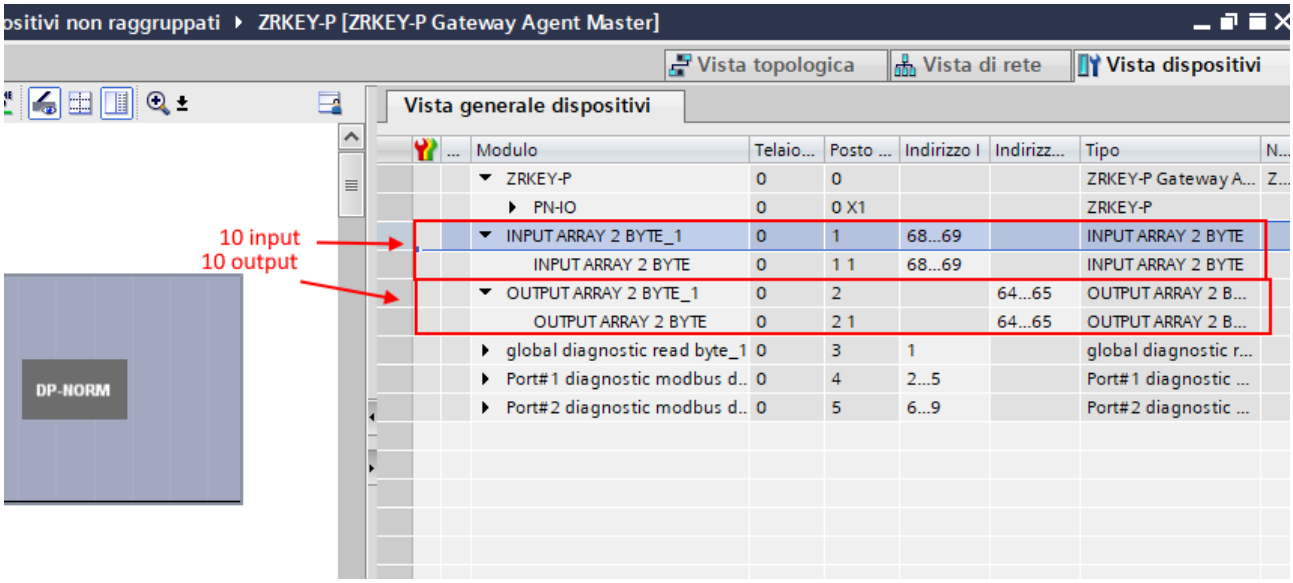


In Profinet the devices are identified by their name, so right click on the Seneca device and select "Assign device name"



Scan the network with "Update list" and set (if necessary) the device name with "Assign name".

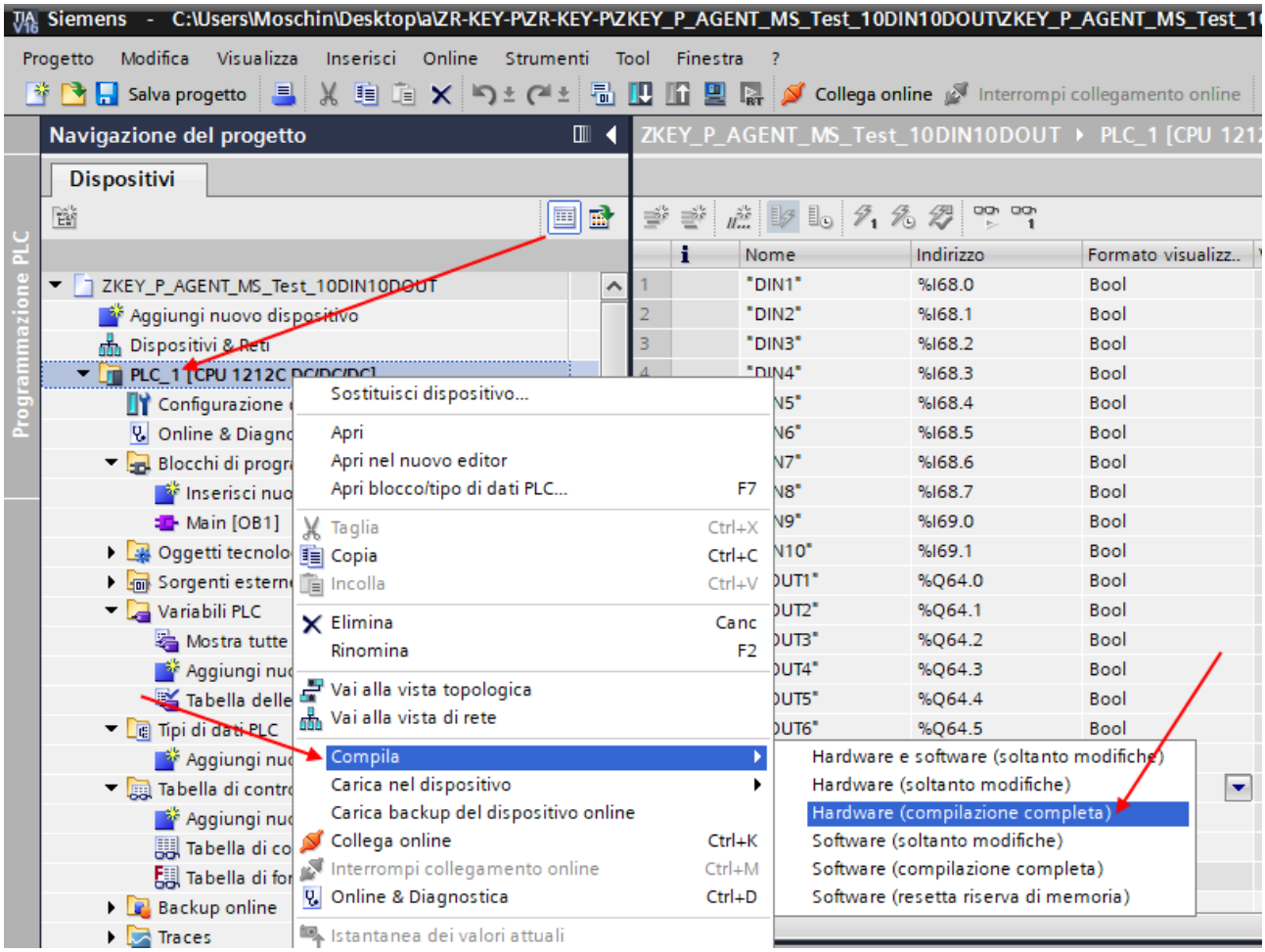
The IO configuration has already been prepared having imported the GSDML project (otherwise if you have imported the generic GSDML file "Gateway All" you must drag the correct number of read/write bytes):



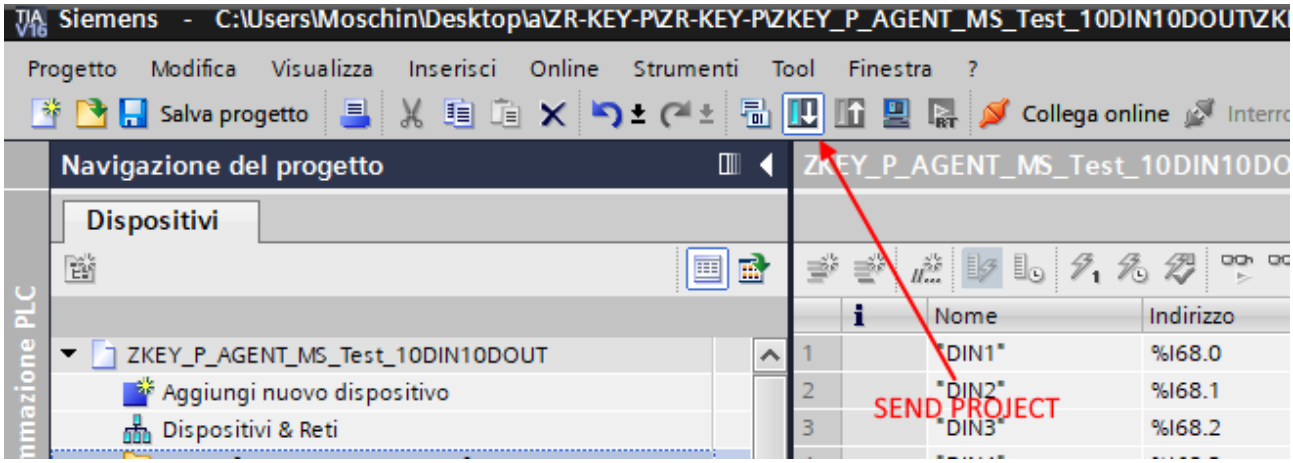
In particular, the 10 inputs are available at addresses I68 and I69 while the outputs are located at addresses Q64 and Q65.

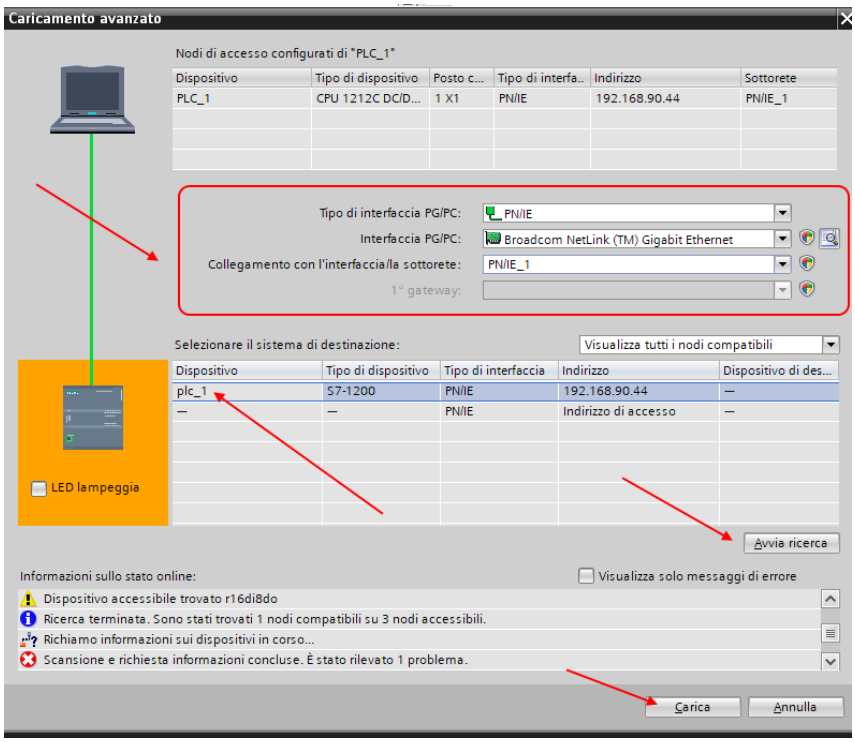
Now the devices are configured, all that remains is to compile and send the configuration to the PLC.

To compile we select the complete hardware compilation:

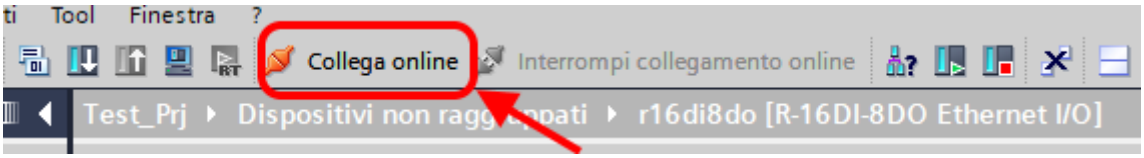


Then press icon to send the project to the PLC:

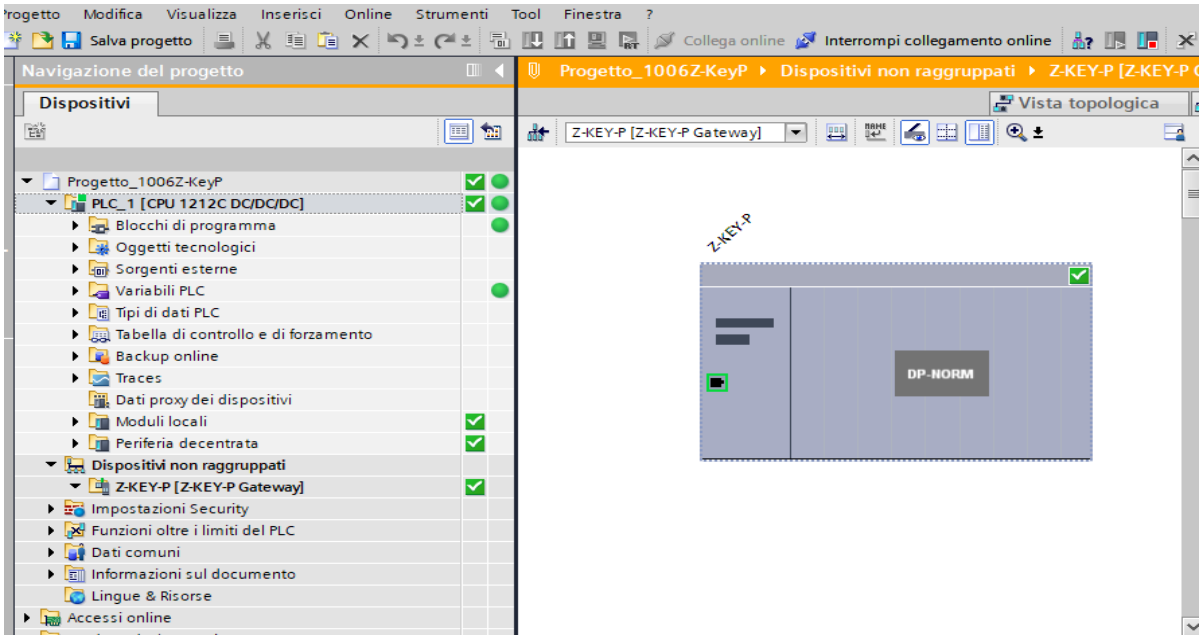




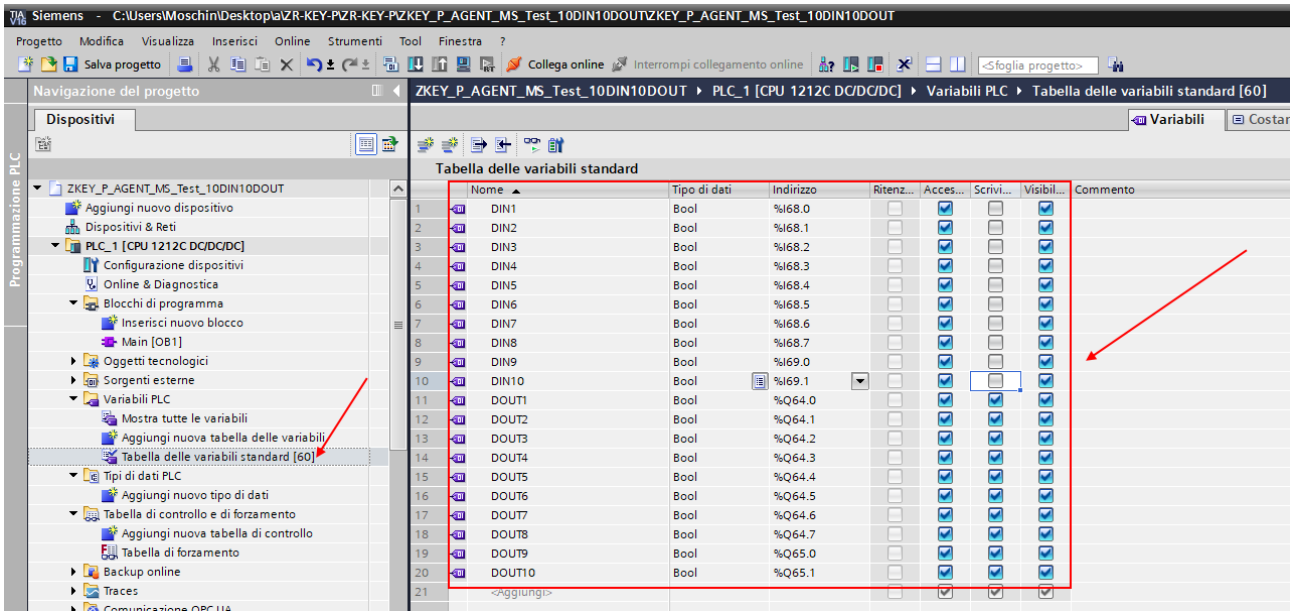
Let's go online to check if there are any errors:



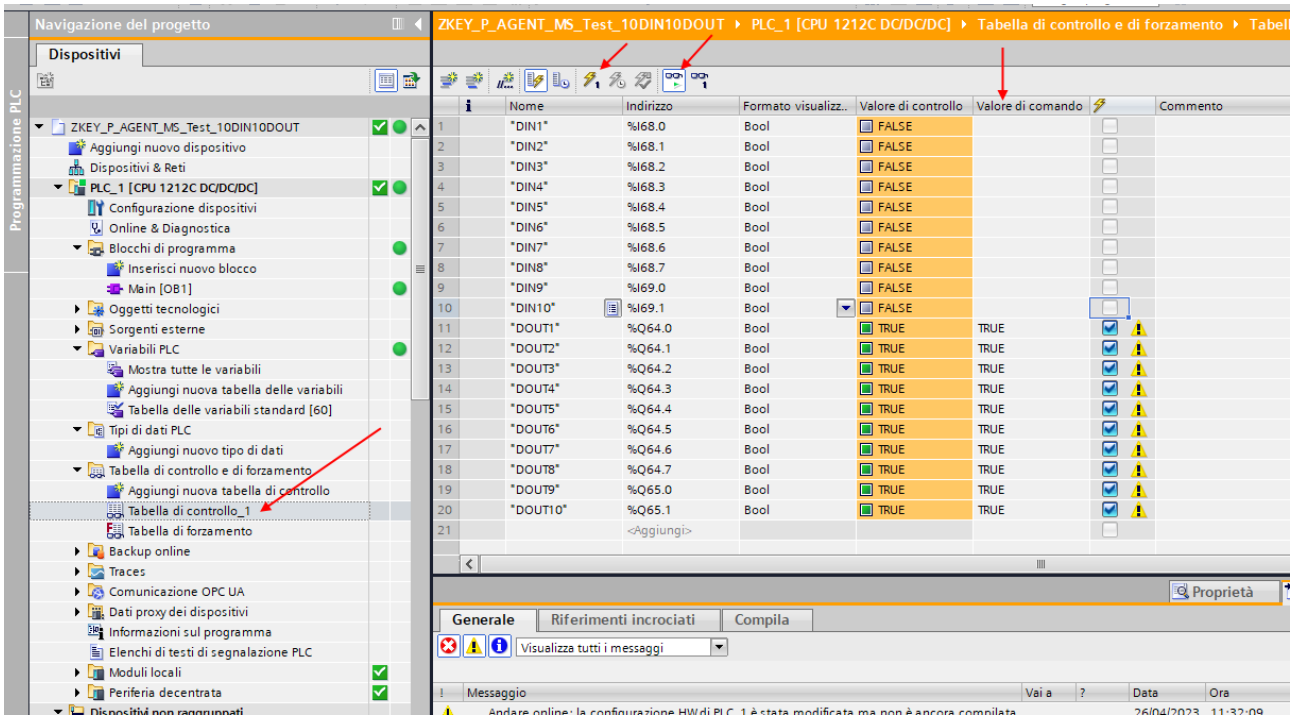
If everything is correct you will get a green icon next to the Seneca device:



It is also possible to read and write the IO (for debugging purposes) directly from the TIA portal.
Then define the variables for the PLC referring to the above addresses:



And then define a control table:



Here it is now possible to read inputs and force write outputs.

8.1.2. "PROFINET IO - MODBUS SLAVE GATEWAY" CONFIGURATION

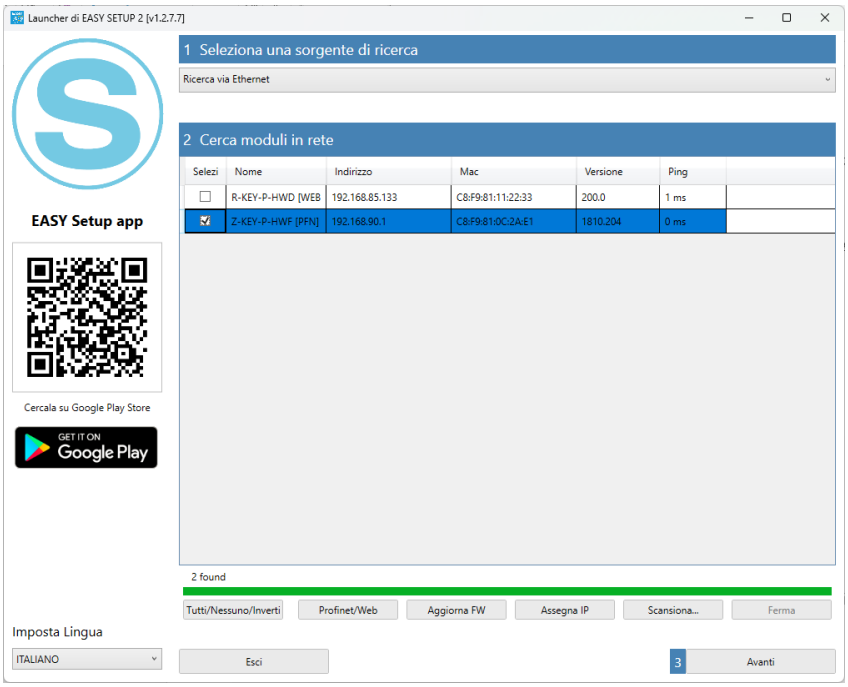
You want to connect a Siemens™ PLC to another PLC connected to serial port 1. The serial PLC supports the Modbus Master protocol.

In the example we will use the Z-KEY-P product (the steps are exactly the same for the other R-KEY-LT-P and Z-KEY-2ETH-P devices).

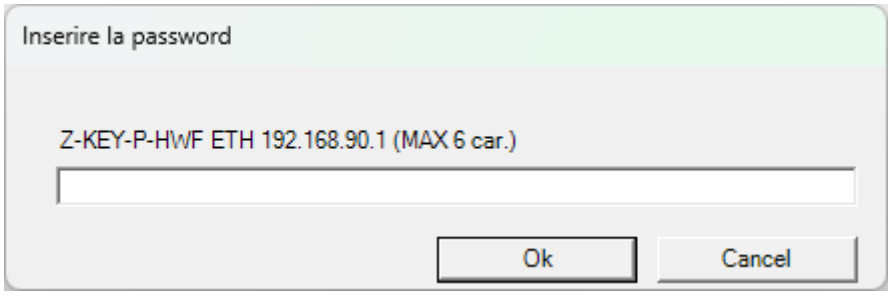
Suppose you want to exchange 10 bytes from the serial PLC to the Siemens PLC and 5 bytes from the Siemens PLC to the serial PLC.

First we disconnect the PLC from the Ethernet network.

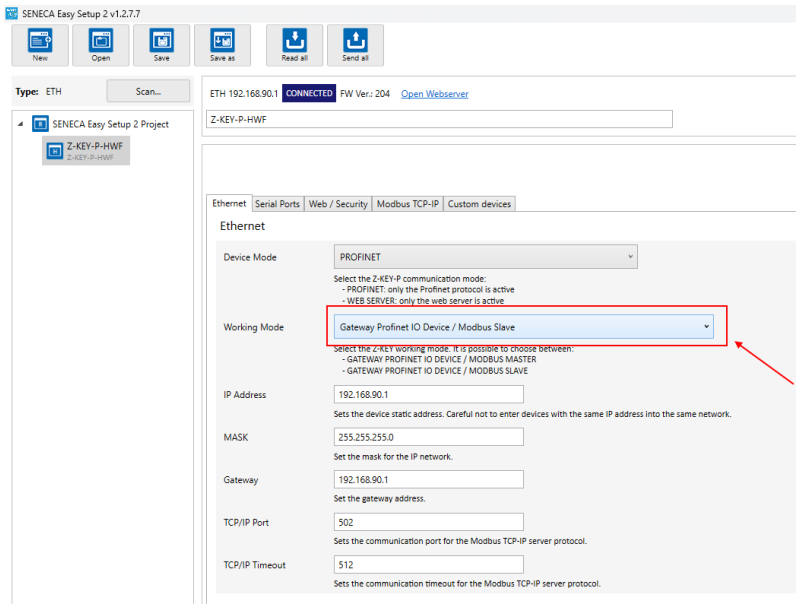
Now we use the Easy Setup 2 software selecting the Z-KEY-P product (with SCAN or in manual entry):



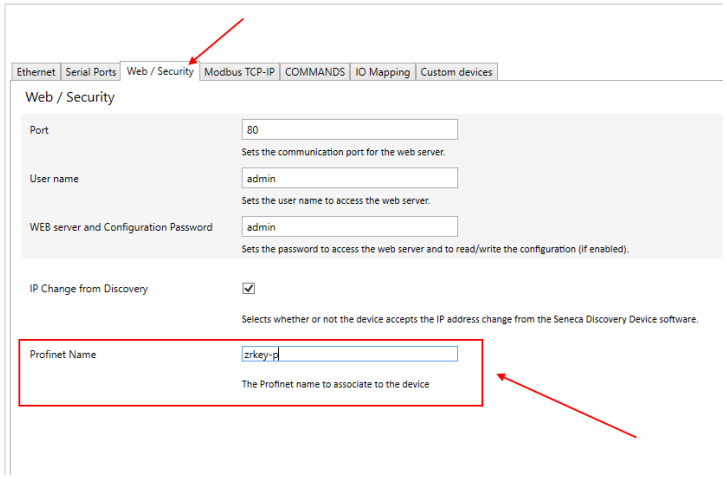
At this point the device access password is requested (default: admin):



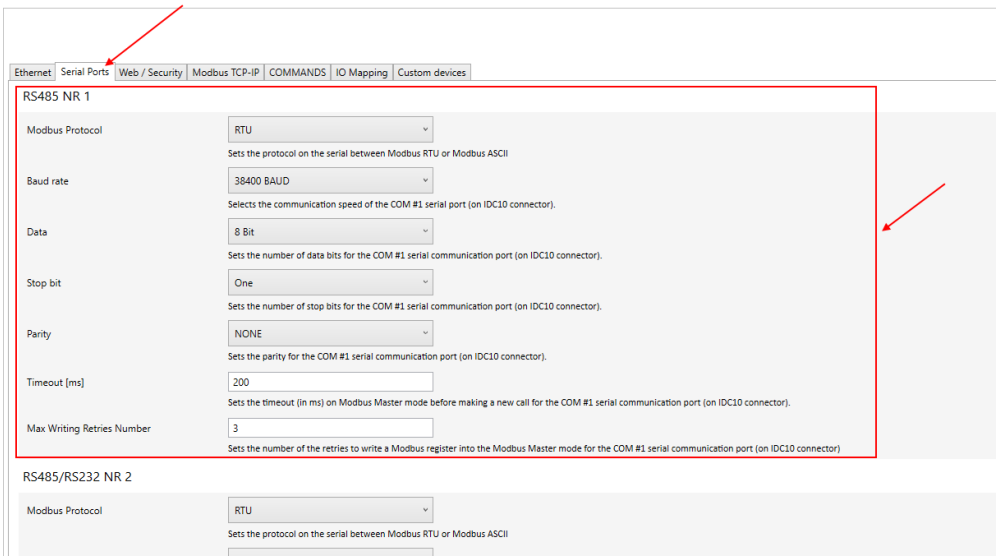
Once the password has been entered, select the Profinet IO Device / Master slave Gateway mode:



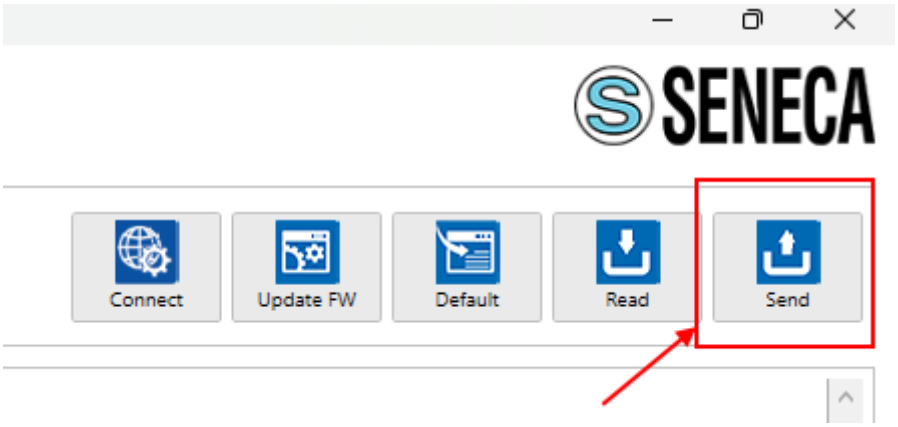
Enter the profinet name of the device:



Check that serial port 1 is configured correctly for the serial PLC:

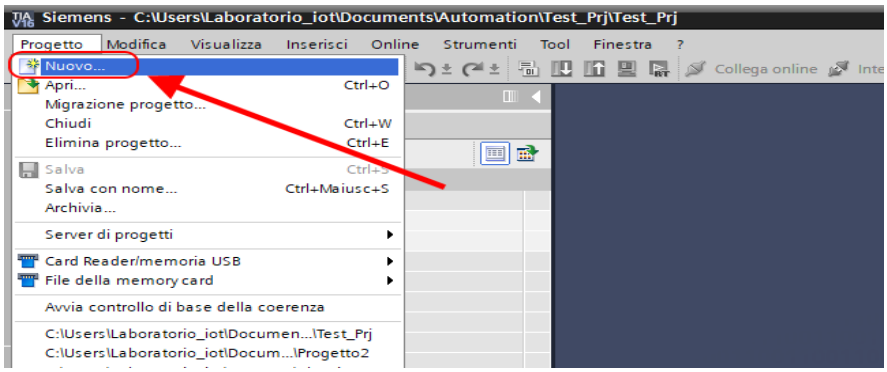


Now let's send the configuration to the device with the "send" button:

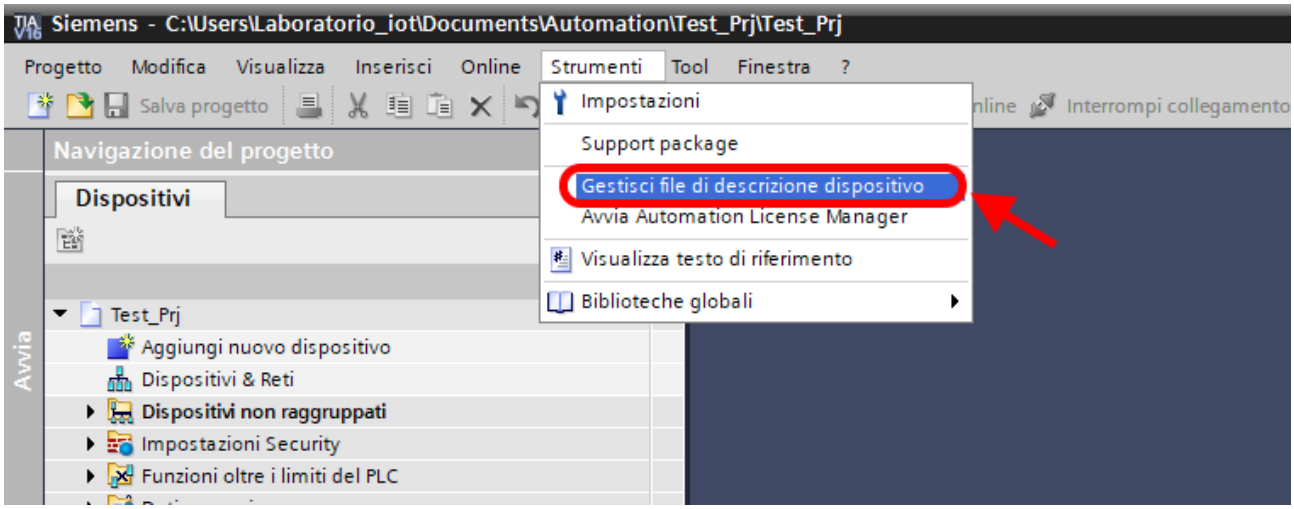


We can now move on to configuring the PLC via Tia Portal™:

Creating a new project:

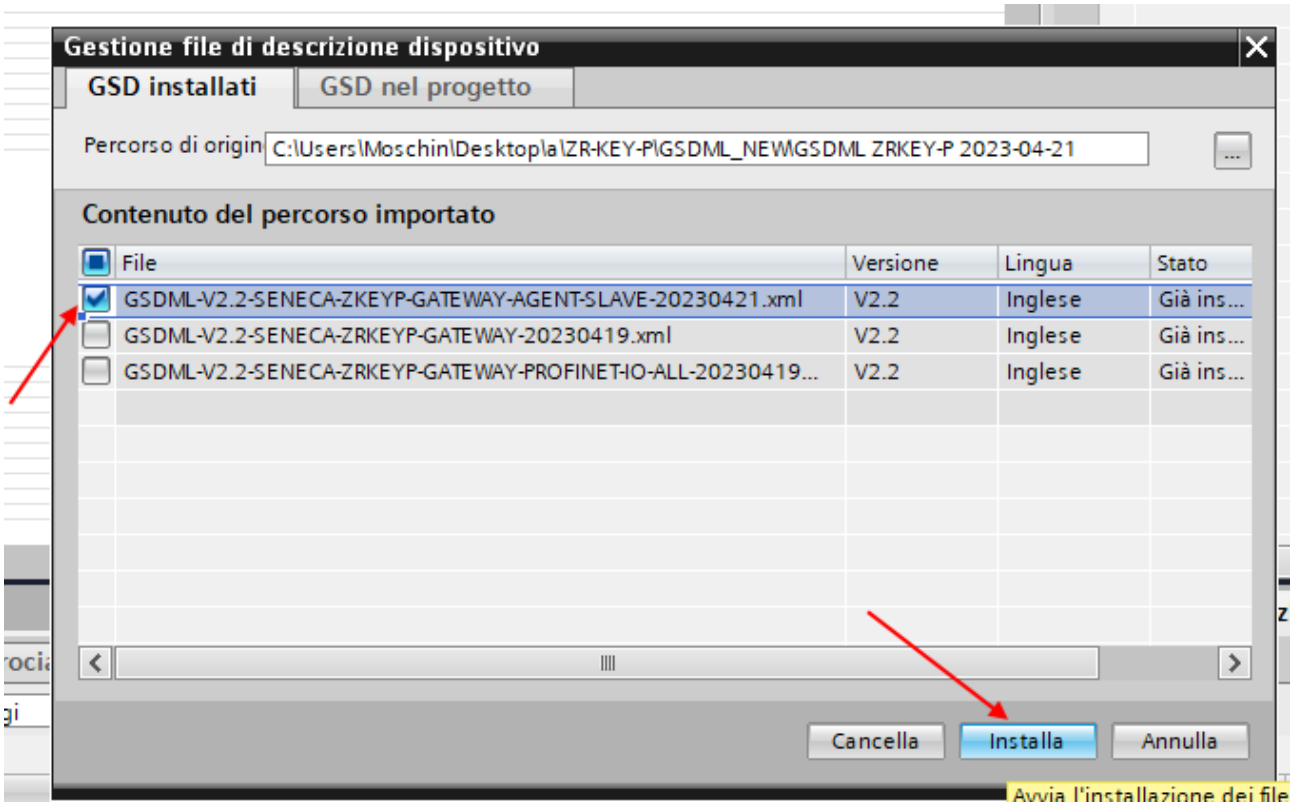


Install the GSD file of the Seneca product:

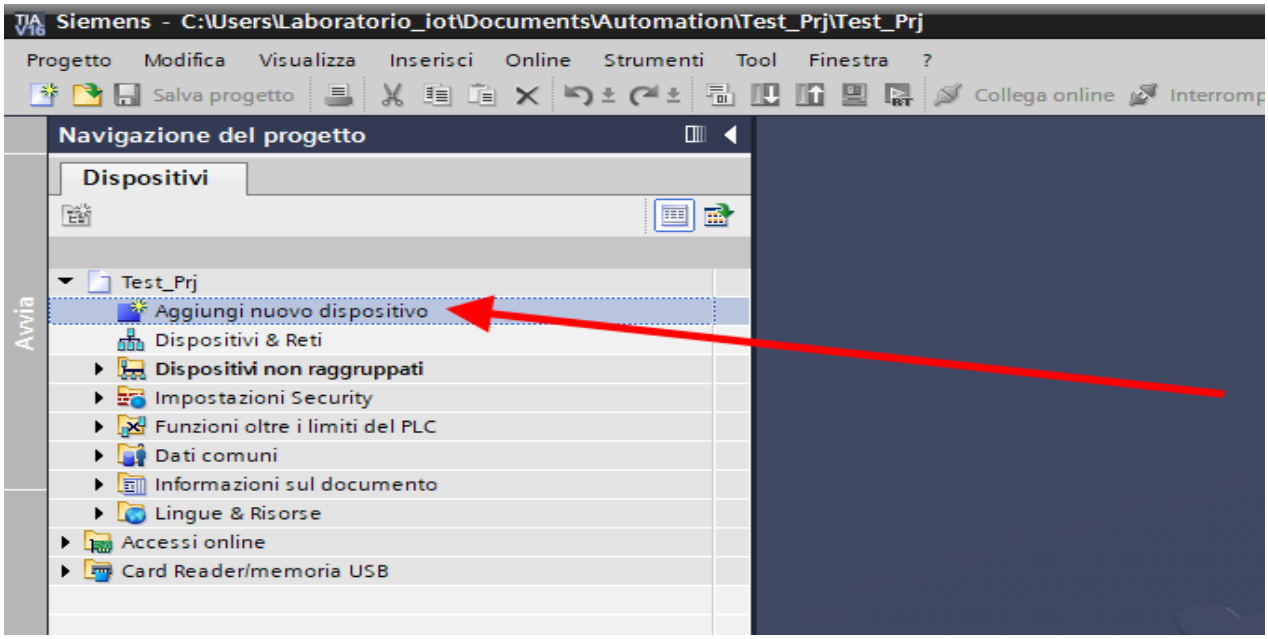


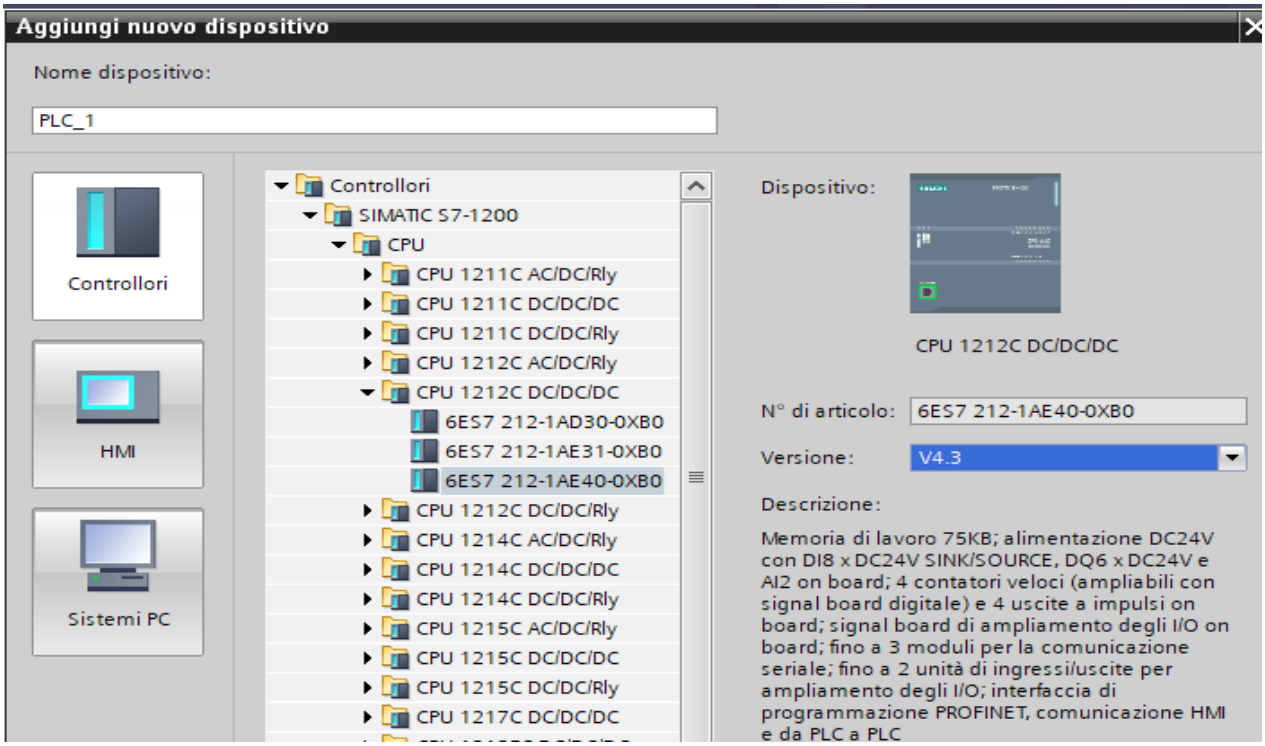
For the Modbus Slave mode, the GSDML file is generic and can be downloaded from the www.seneca.it website in the gateway section of the key-p series.

Point to the directory where you saved the GSDML file and press INSTALL.

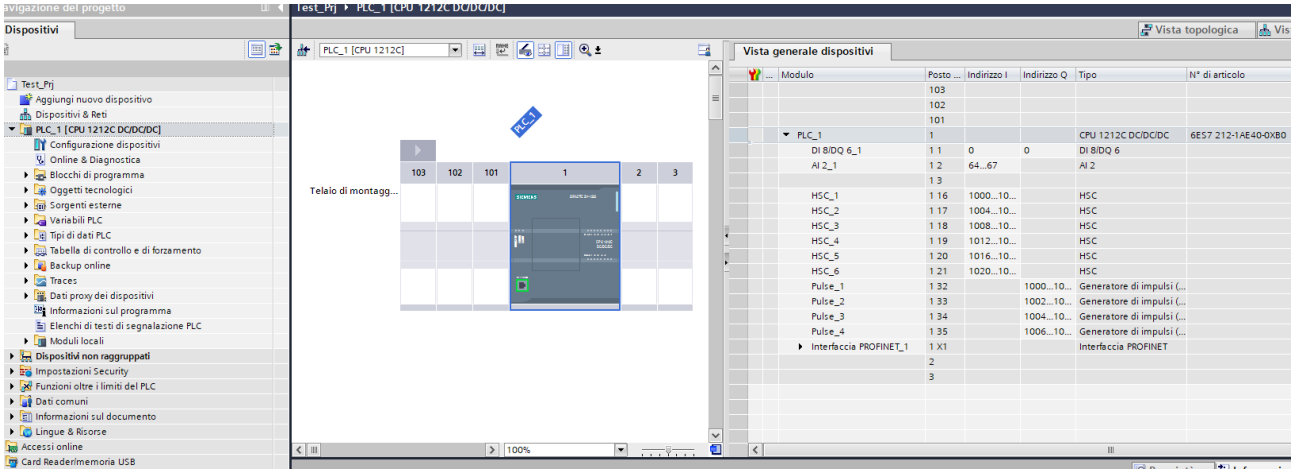


Now insert the Siemens PLC (in our example a SIEMATIC S7 1200), click on "Add new device ...":

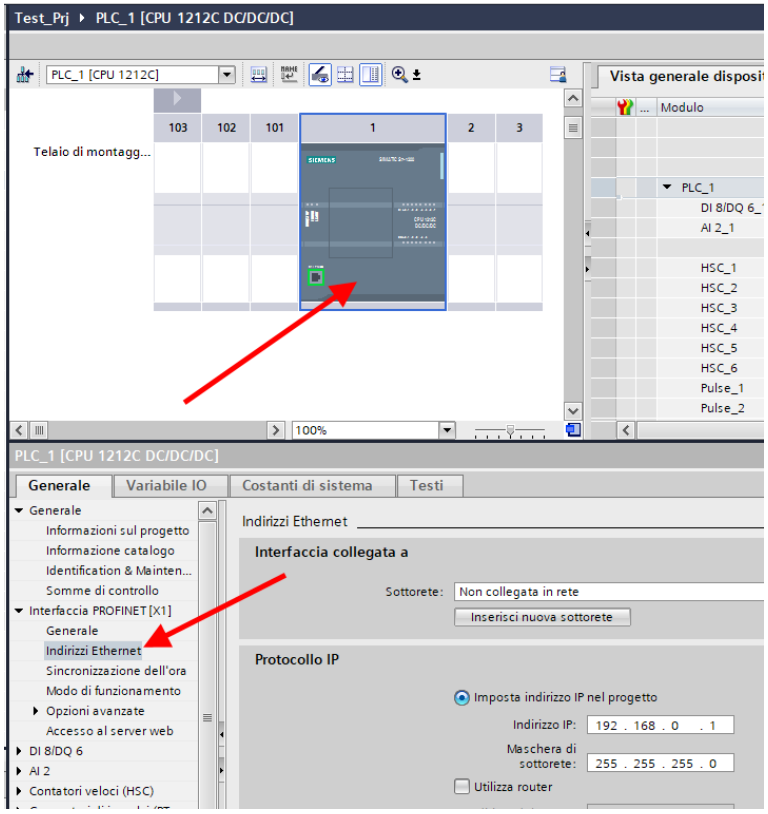




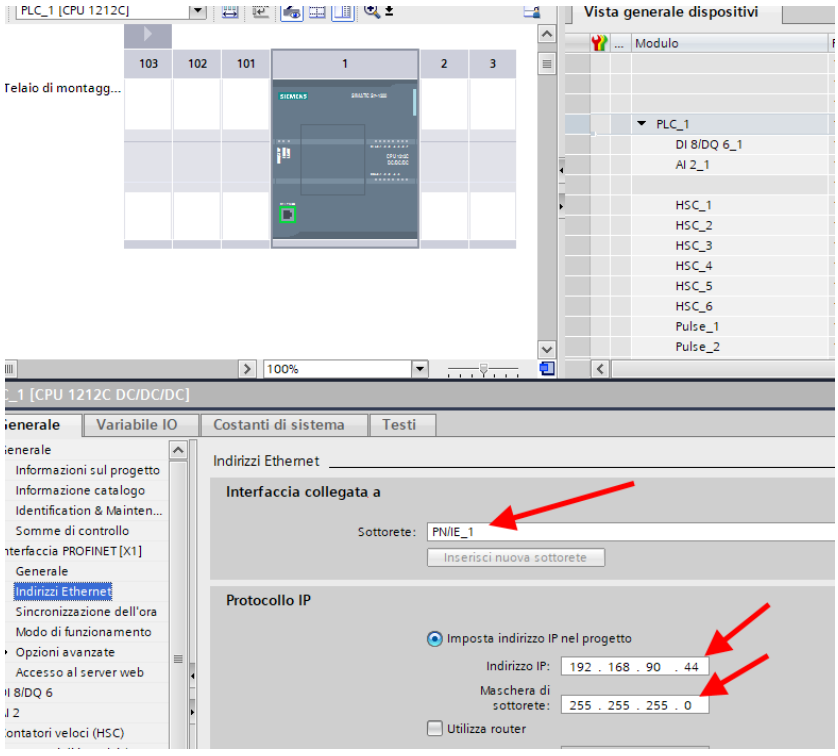
Confirm and the PLC will be added to the rack:



Now click on the PLC and select Profinet interface -> Ethernet addresses

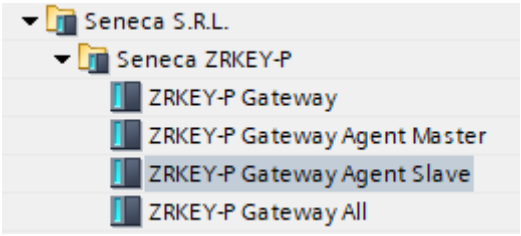


Set the IP you want for the PLC (in this case 192.168.90.44) and the PLC subnet:

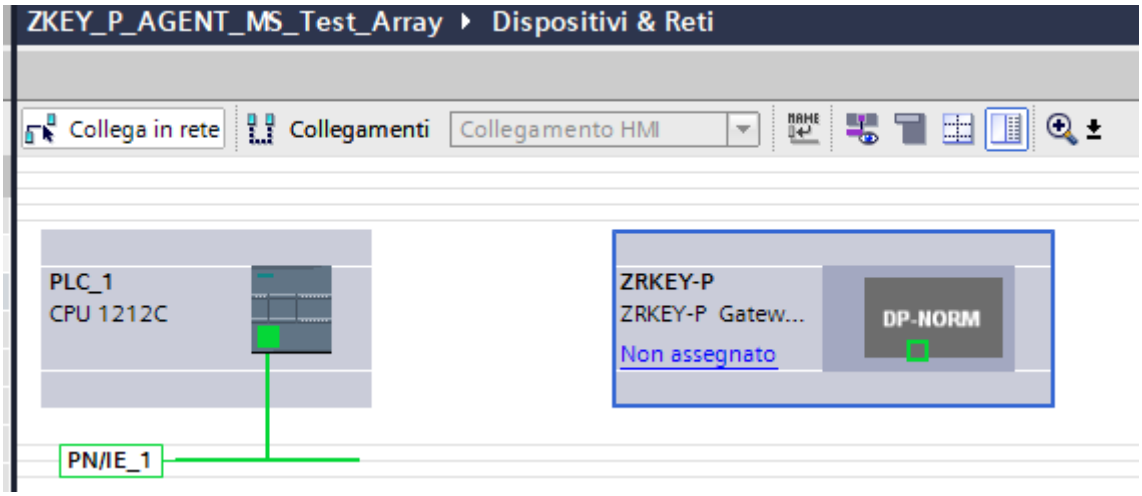


Move on to “devices and network” view:

On the right select "Hardware Catalogue" and then under "Additional Field Equipment" -> PROFINET IO -> GATEWAY -> Seneca SRL -> ZR-KEY-P Gateway -> ZRKEY-P Gateway Agent Slave

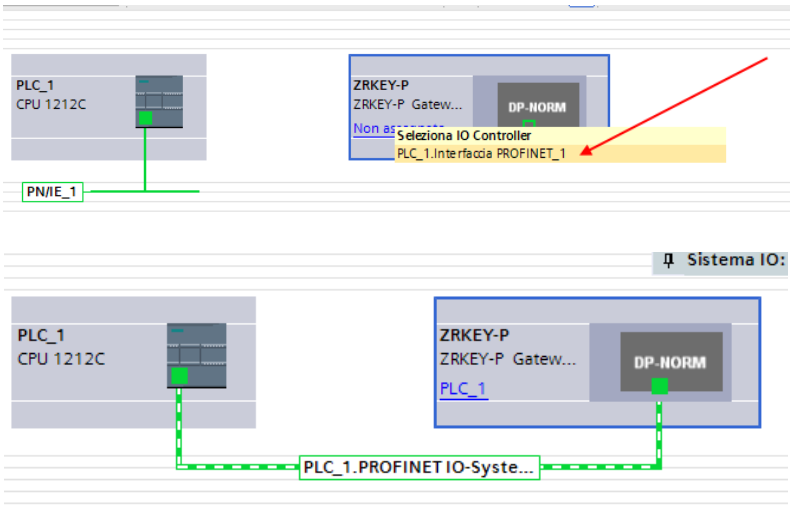


Drag the device to the network view:

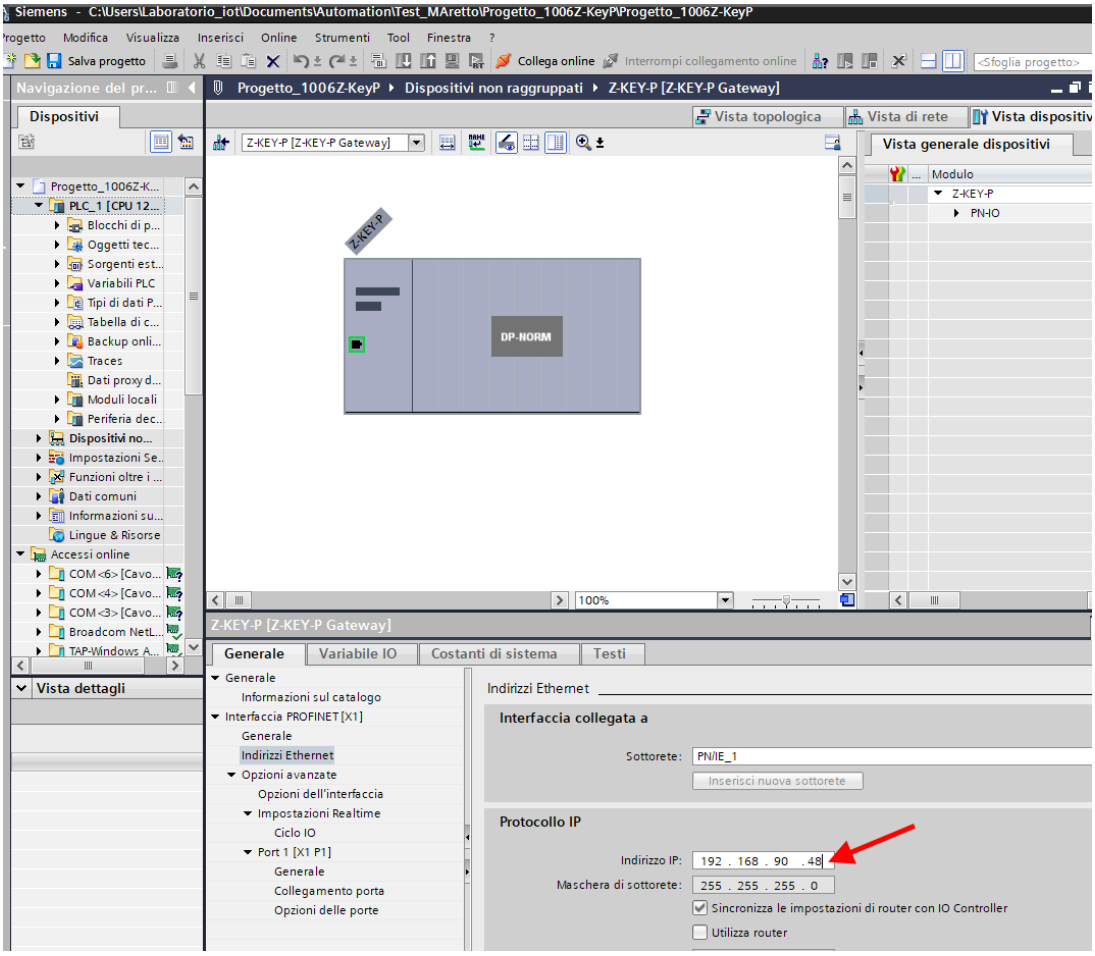


Now associate it with the PLC:

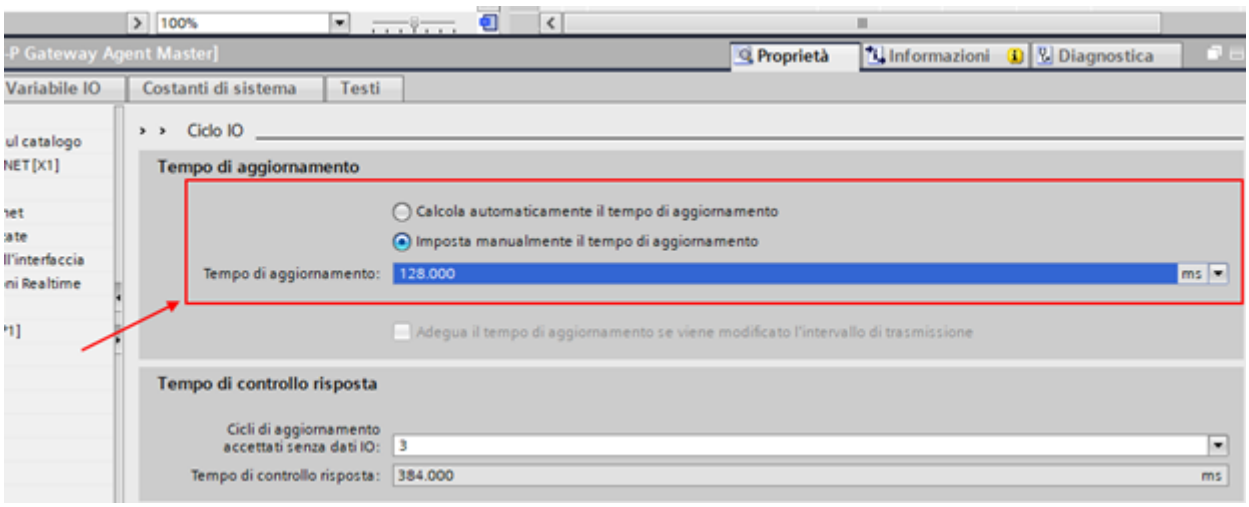
Click with the left mouse button on "Not assigned" and then select the PLC:



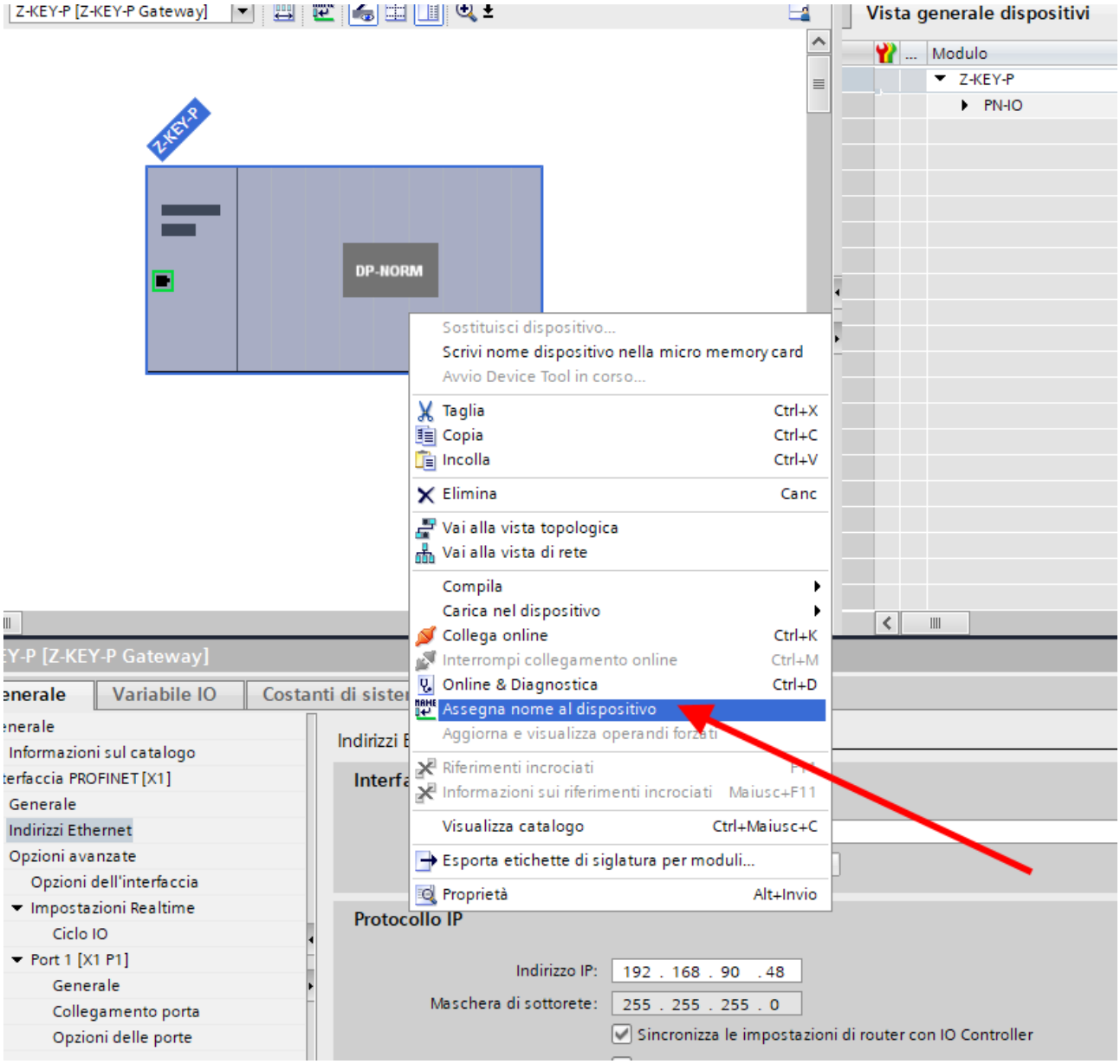
Click twice on the Seneca device and configure the IP address here too (for example 192.168.90.48) and the timing:



Depending on the project it is necessary to set the cycle time (typically 128 ms):



In Profinet the devices are identified by their name, so right click on the Seneca device and select "Assign device name"

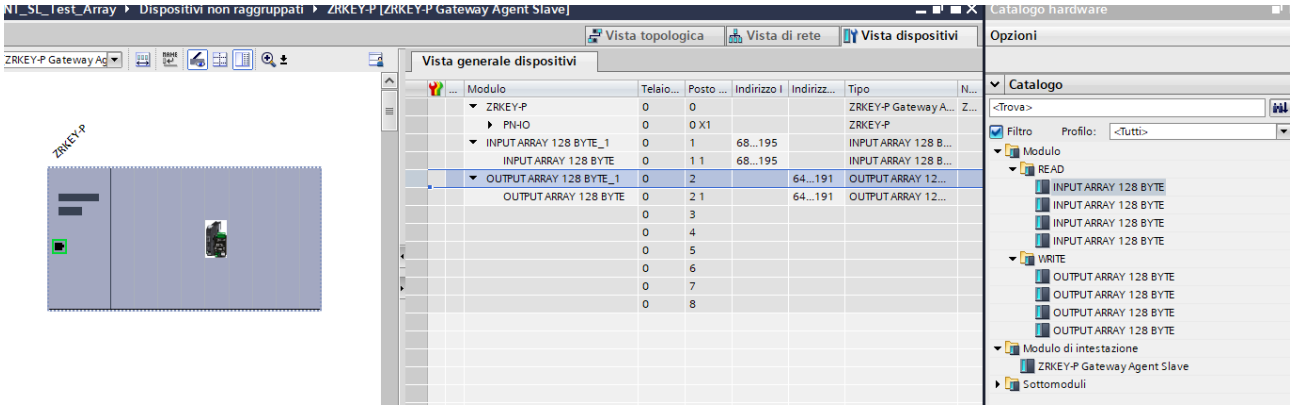


Scan the network with "Update list" and set (if necessary) the device name with "Assign name".

You said that you want to get the following map:

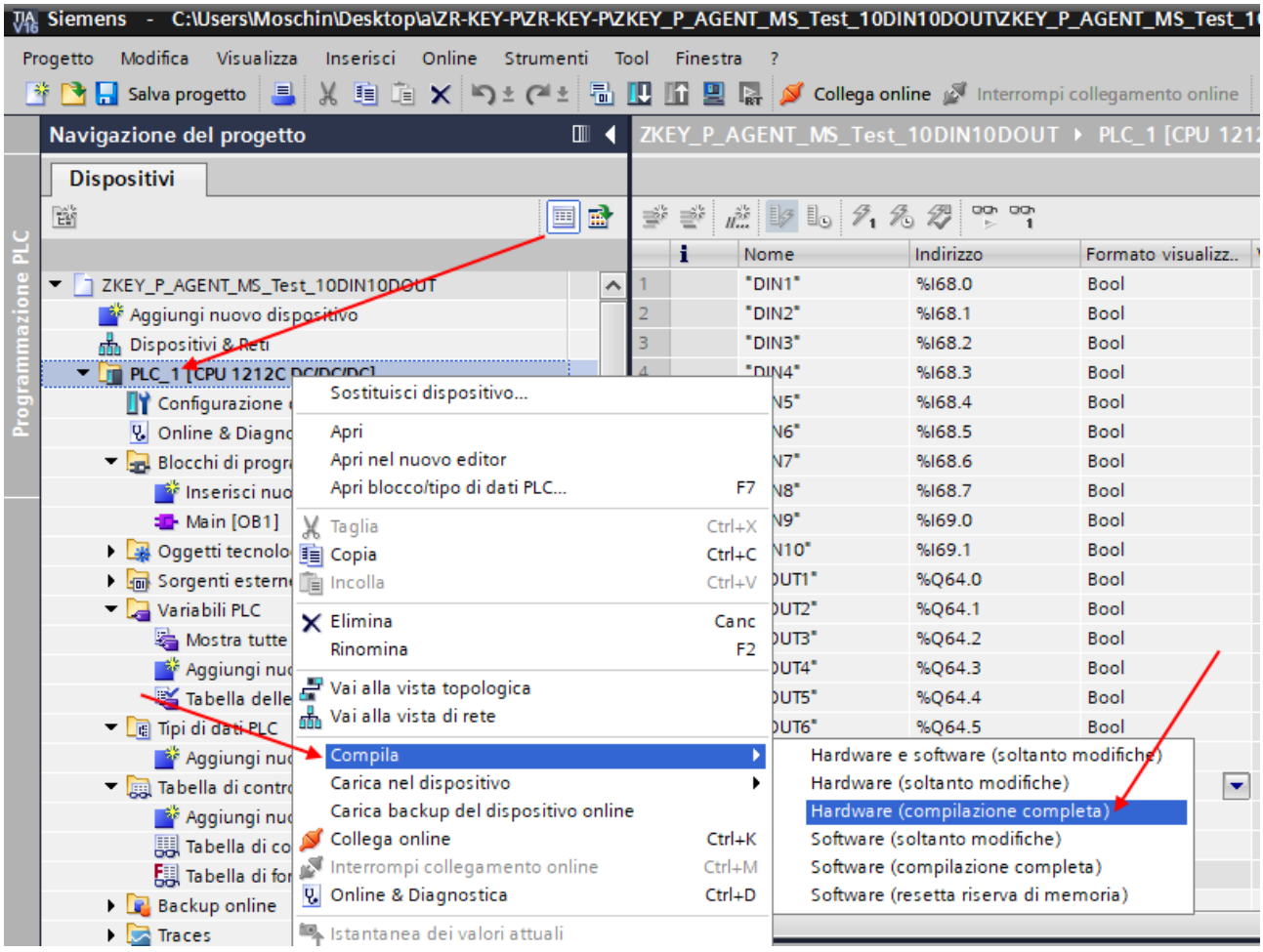
SERIAL PLC -> Writes 10 Byte on Modbus -> SIEMENS PLC Reads 10 Byte from Profinet
SIEMENS PLC -> Writes 5 Bytes on Profinet -> SERIAL PLC Reads 5 Bytes from Modbus

The IO configuration must therefore be prepared:

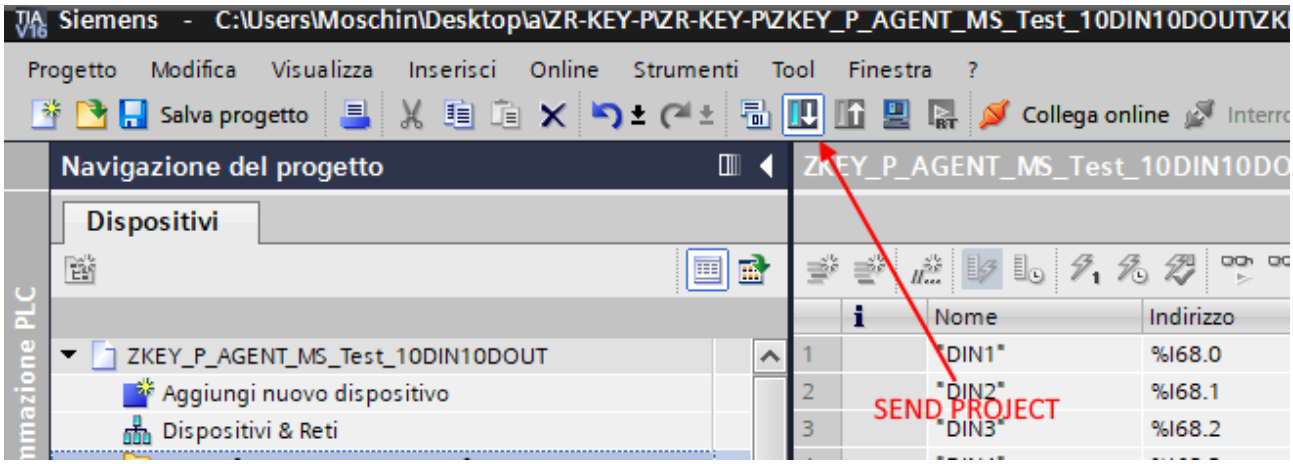


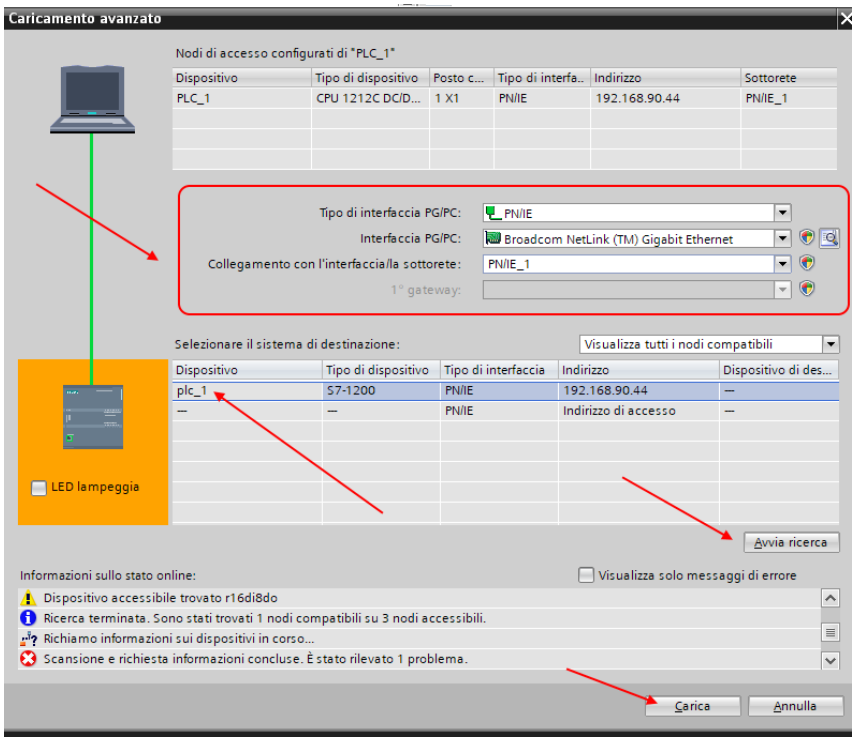
Move one 128-byte array for inputs and another 128-byte array for outputs.
You will only need 10 bytes for writing and 5 bytes for reading.

Now the devices are configured, all that remains is to compile and send the configuration to the PLC.
To compile we select the complete hardware compilation:

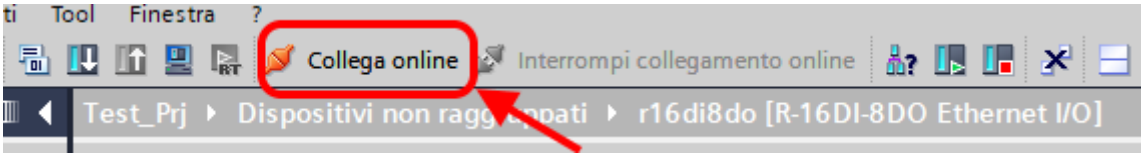


Then press icon to send the project to the PLC:

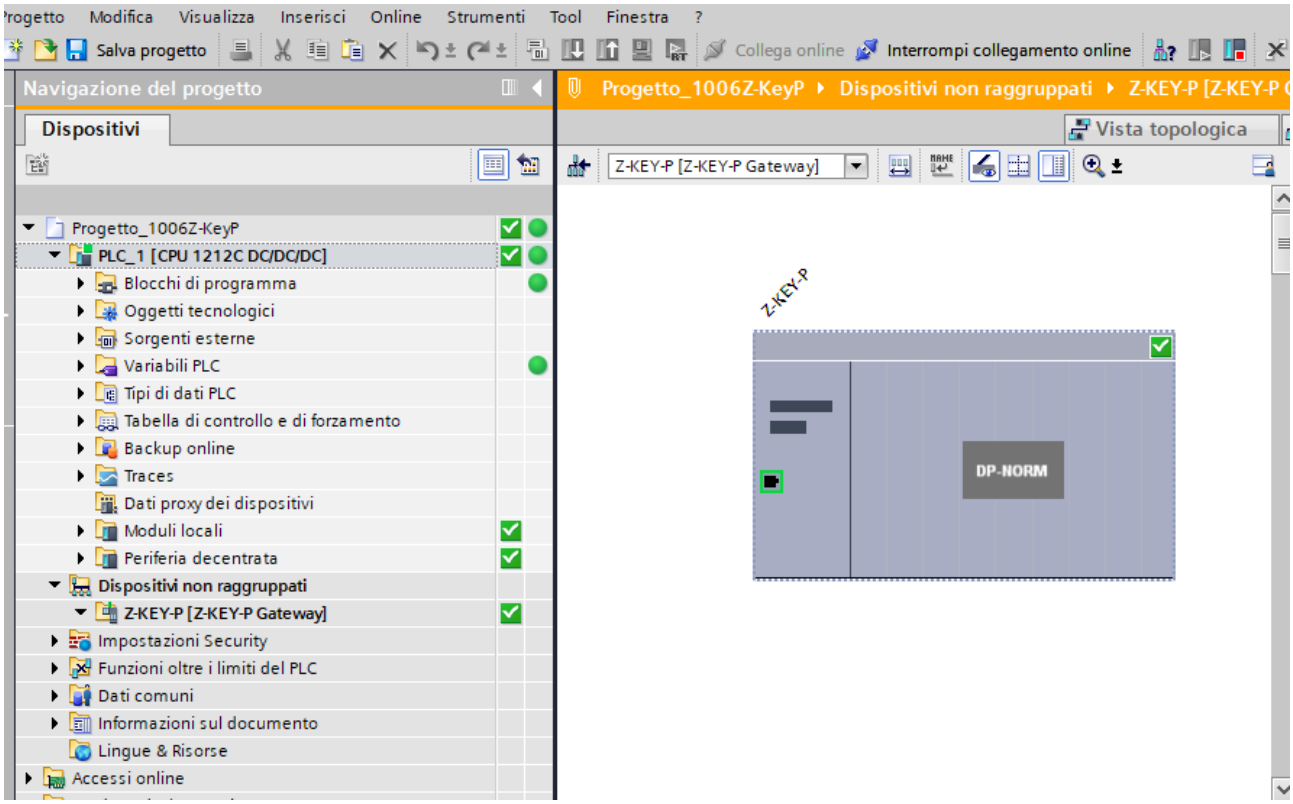




Let's go online to check if there are any errors:



If everything is correct you will get a green icon next to the Seneca device:



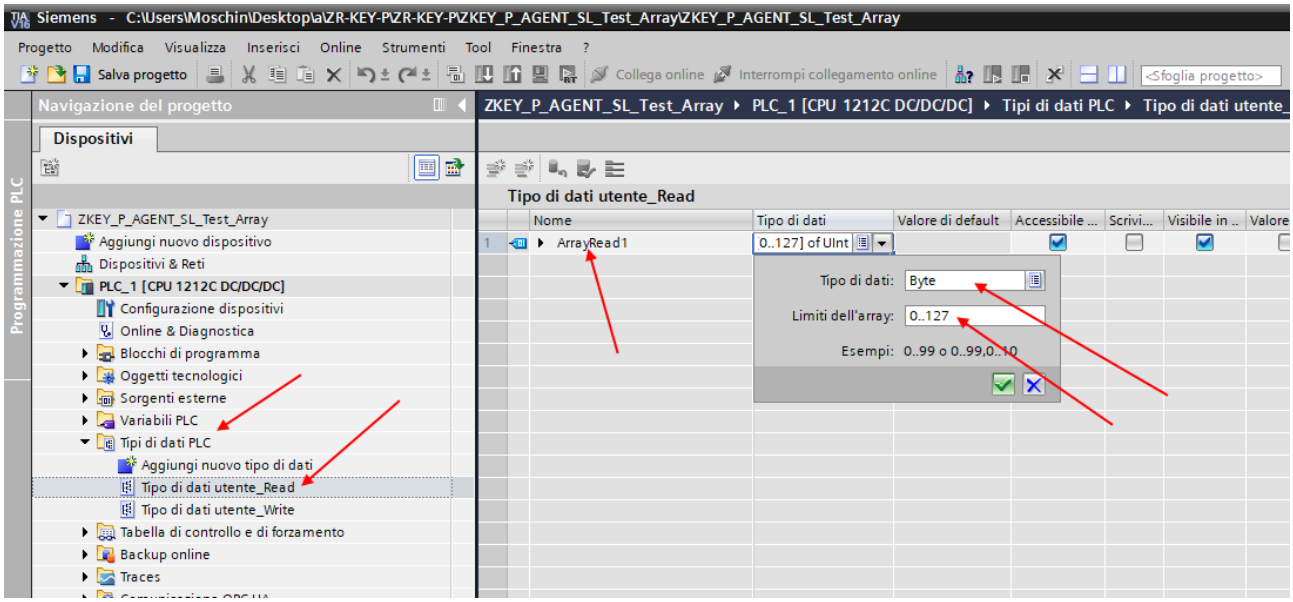
It is also possible to read and write the IO (for debugging purposes) directly from the TIA portal.

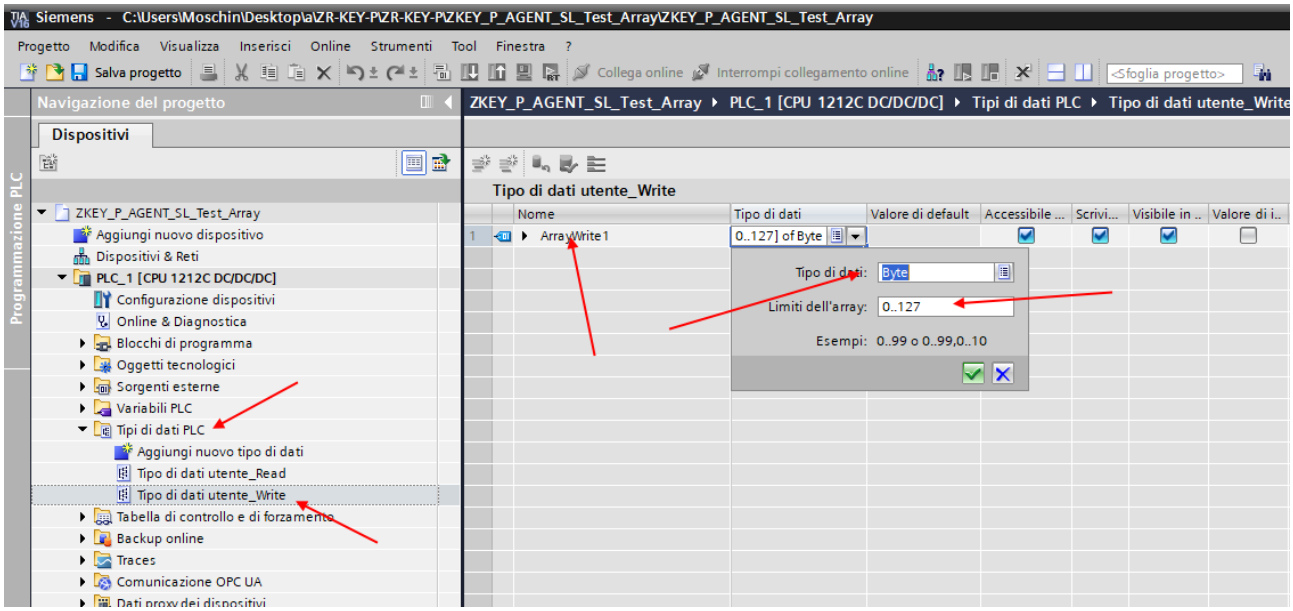
Important:
Registers written in Modbus cannot be read by Modbus but only by Profinet
Registers read by Modbus cannot be written by Modbus but only by Profinet

Then define the variables for the PLC.

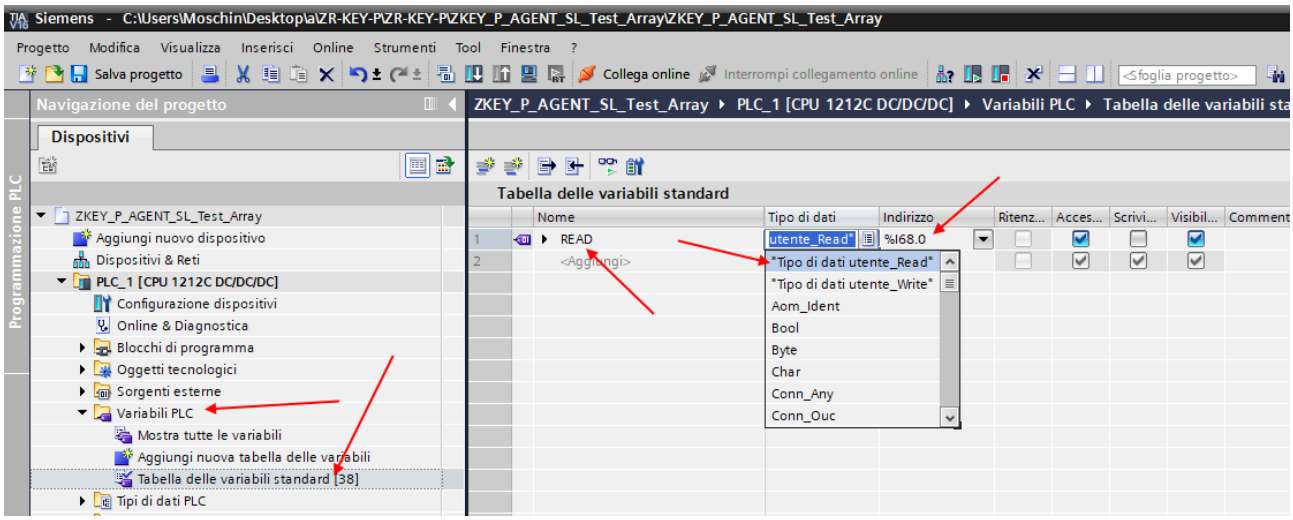
It is convenient to define data types to manage arrays:

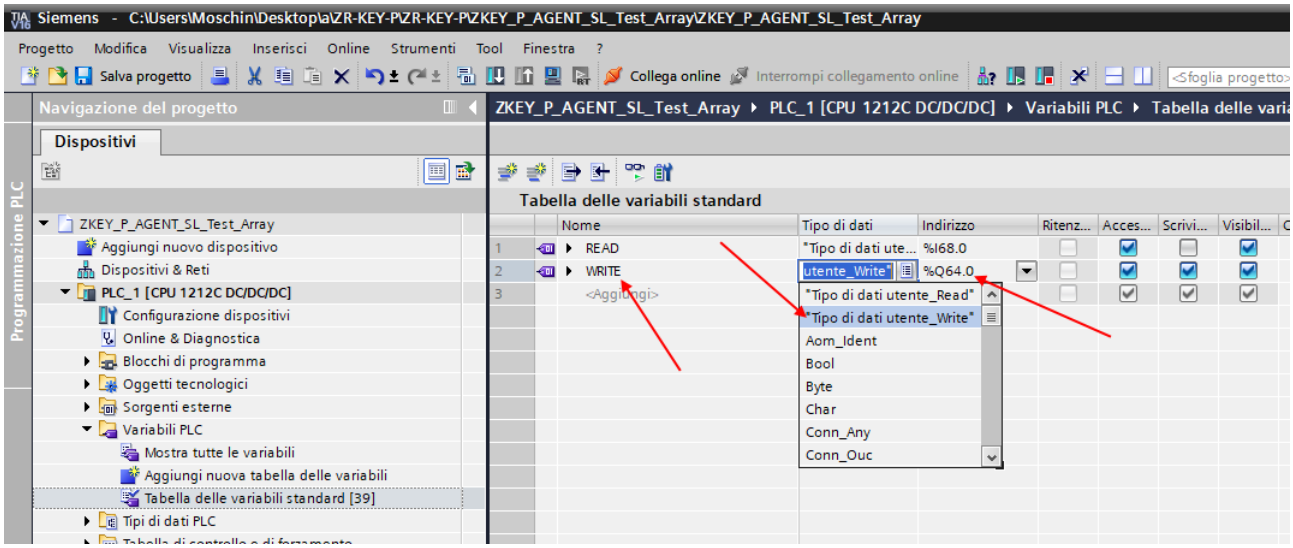
Create two data types, one for read and one for write, each of 128 bytes:



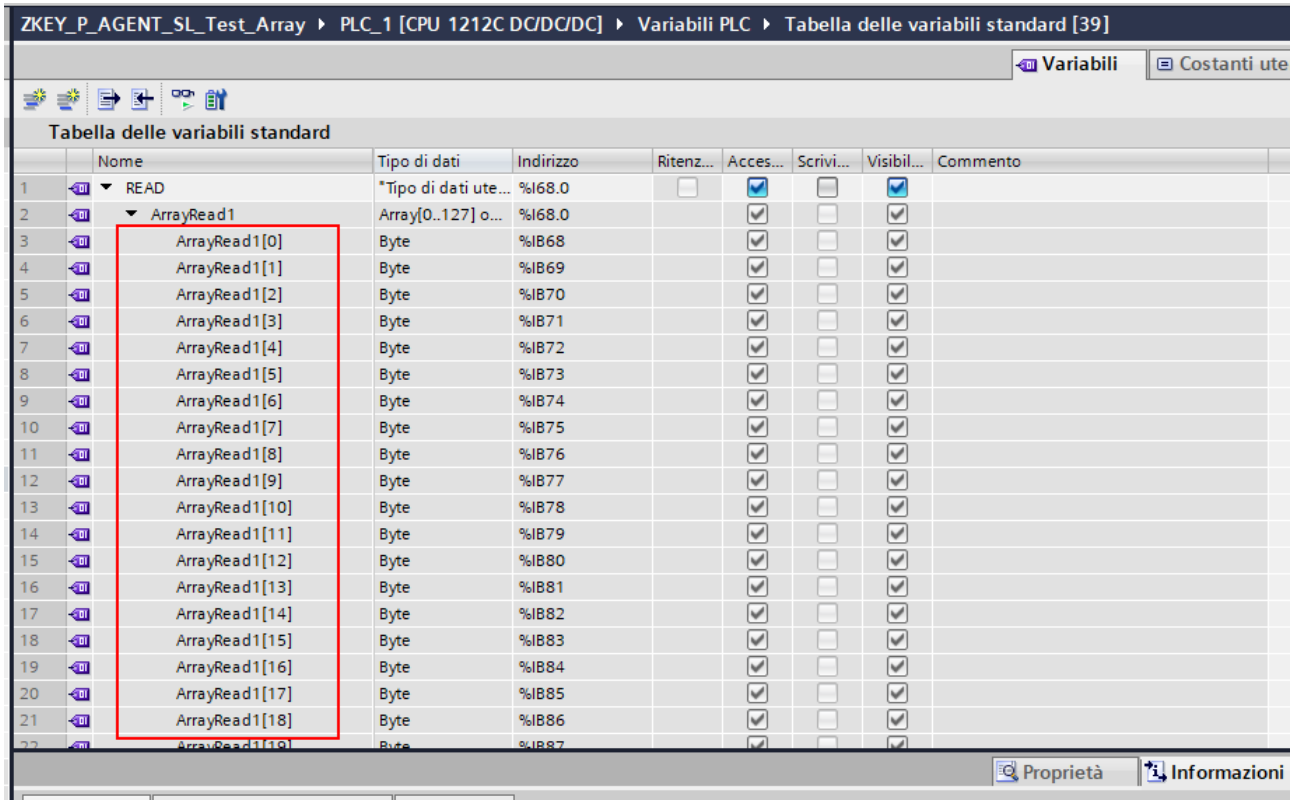


Now define the PLC variables using the newly created ones as data type:

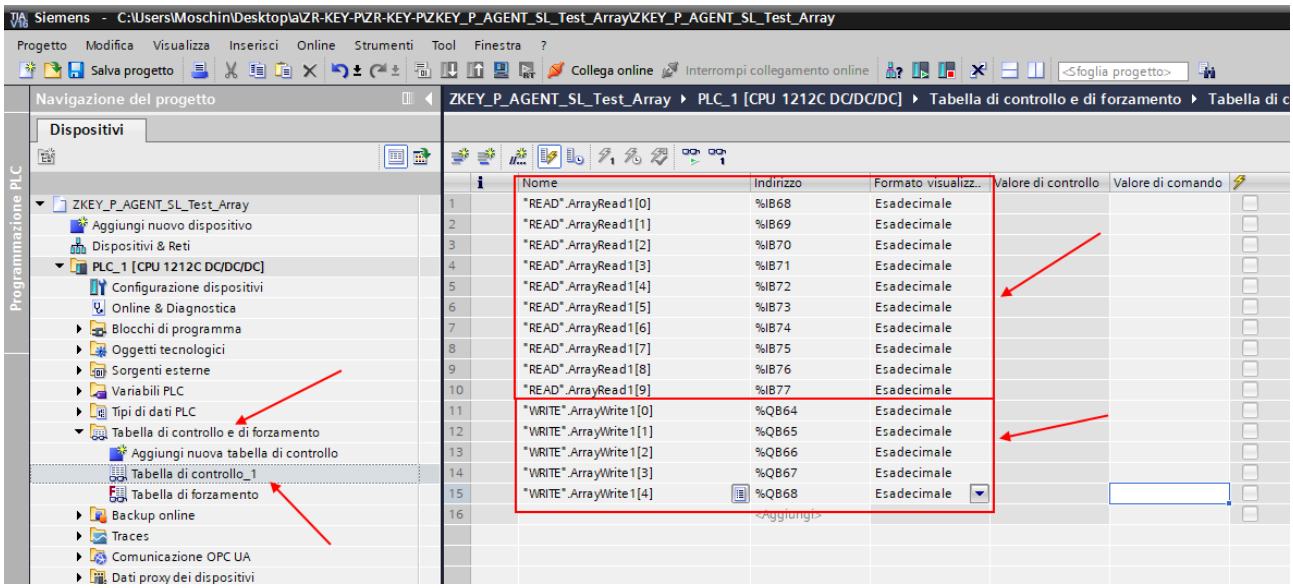




This way you created the arrays (albeit beyond our needs):
For example the array of reads:



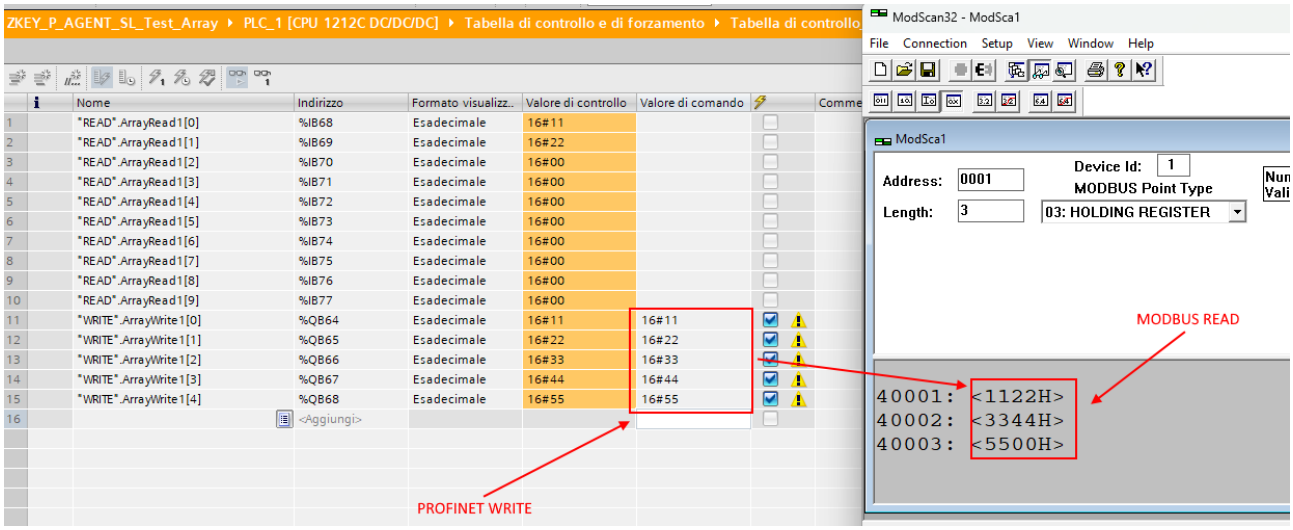
And then define a control table using the following notation:



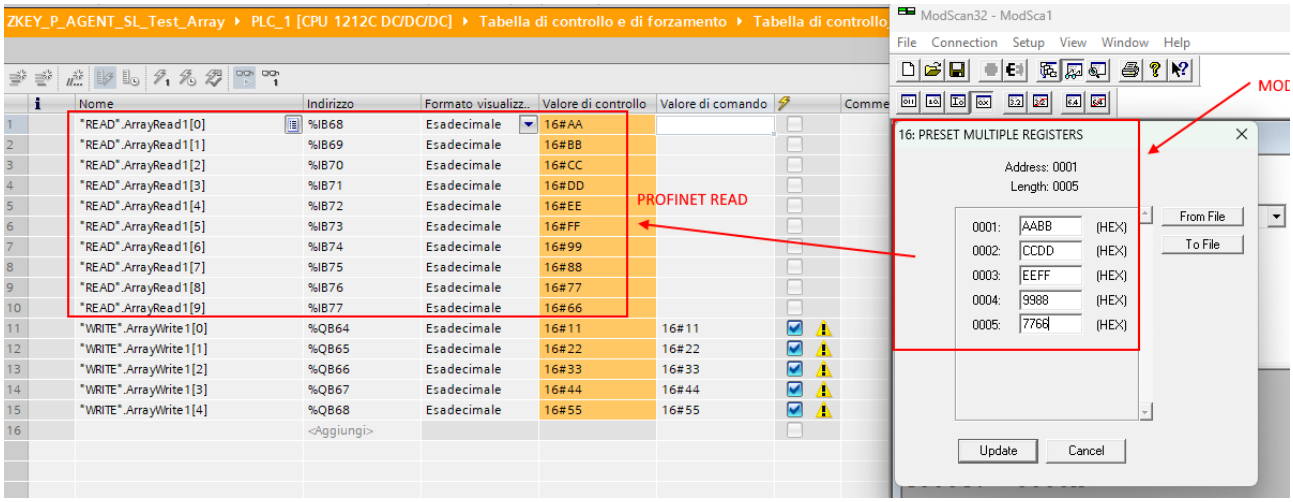
The 5 bytes of Profinet writing are the 5 bytes of reading from modbus (3 Modbus Bytes = 6 bytes).
The 10 bytes of Profinet readings are the 10 bytes of modbus writing (5 Modbus registers = 10 bytes).

Here it is now possible to read inputs and force write outputs.

Profinet writes in the “Write” arrays are read by Modbus like this:



Writes from modbus are read by profinet like this:

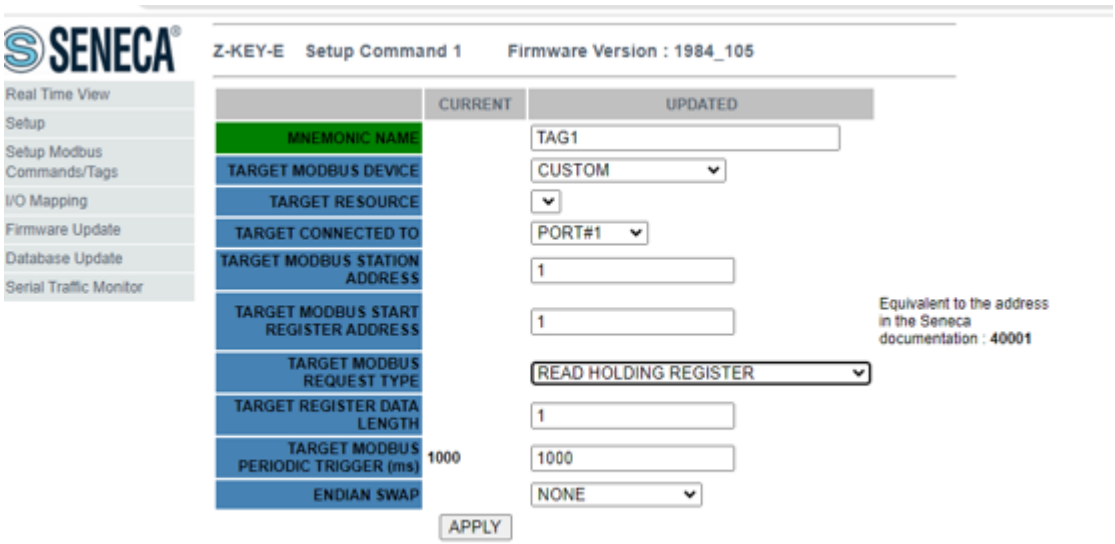


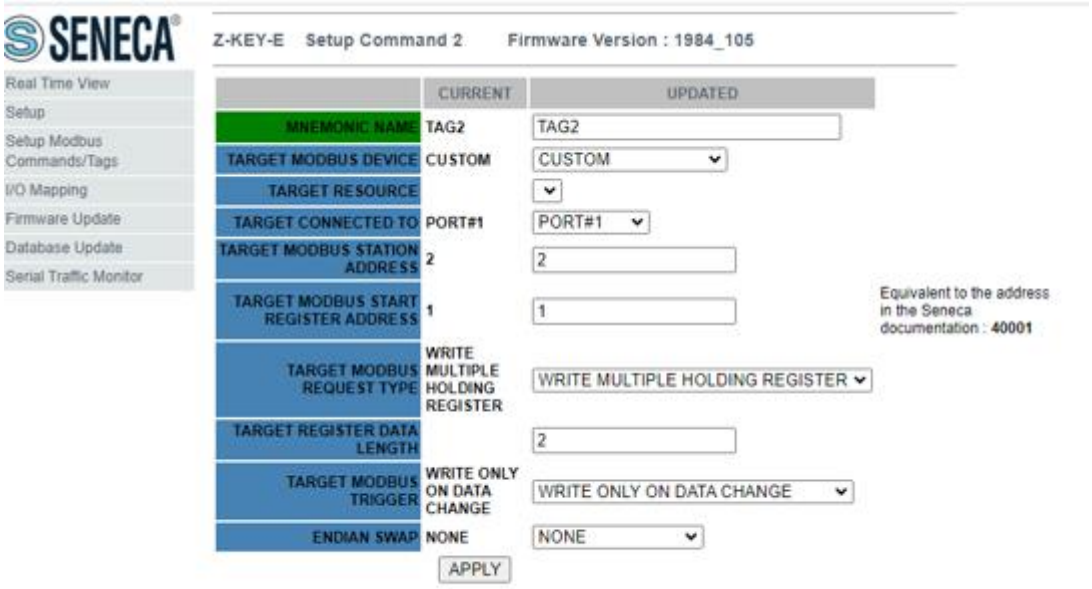
8.2. "-E" GATEWAY CONFIGURATION WITH WEBSERVER AND STUDIO 5000 LOGIX DESIGNER ® SOFTWARE

The version used in this chapter of the Studio 5000 software is 35.00.00.
First you need to configure the Gateway via the webserver:
Let's configure the basic Ethernet/IP parameters:

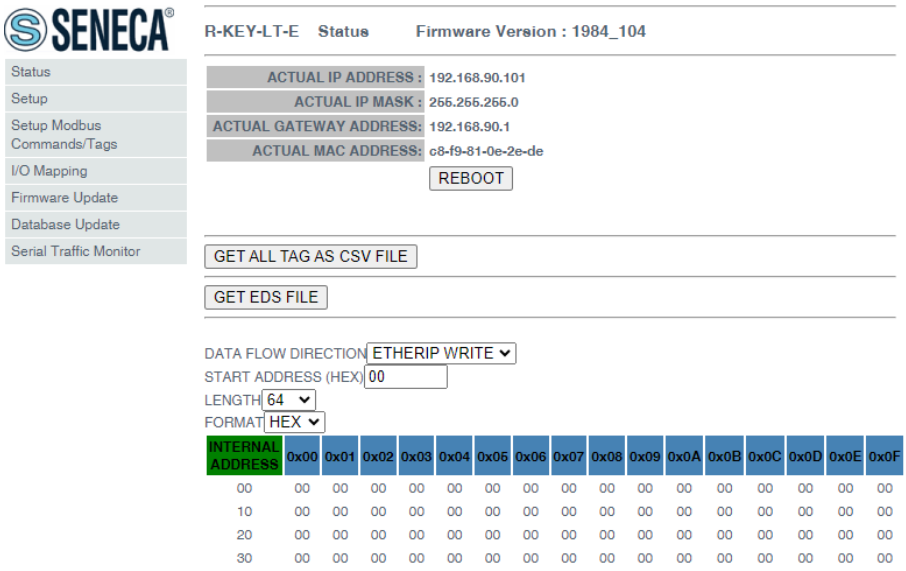
Confirm with "APPLY">

Let's configure 1 Modbus Read register and 2 Modbus Write registers:



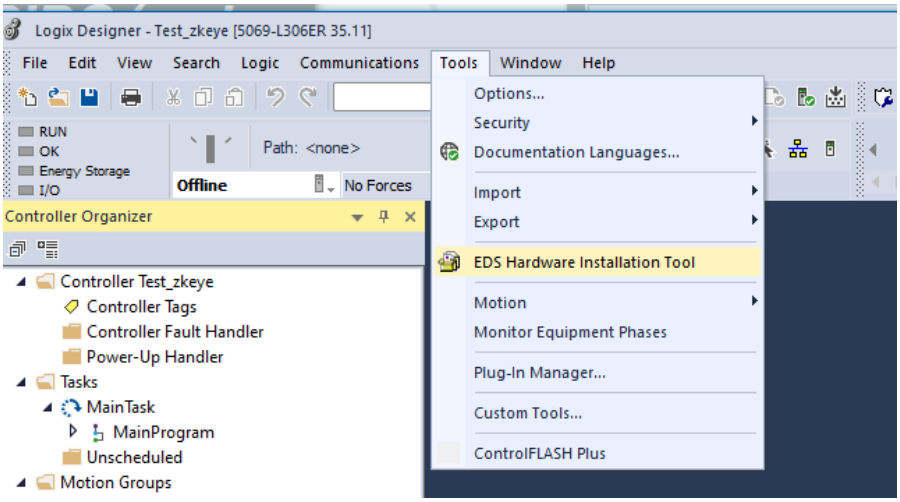


Now on the “Status” section let's export the EDS file using the “GET EDS FILE” button:

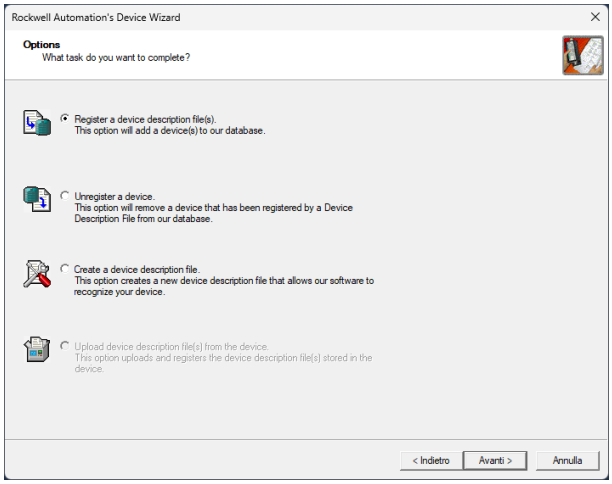


The gateway configuration is complete.

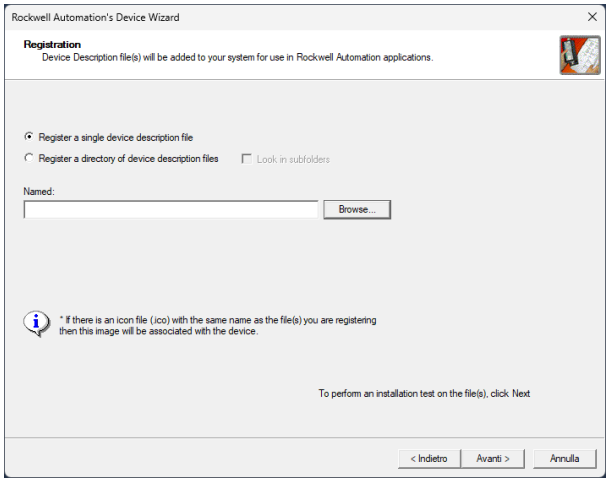
Now in the Studio 5000 software we import the EDS file we have just exported:
In the TOOLS-> EDS Hardware Installation Tool menu:

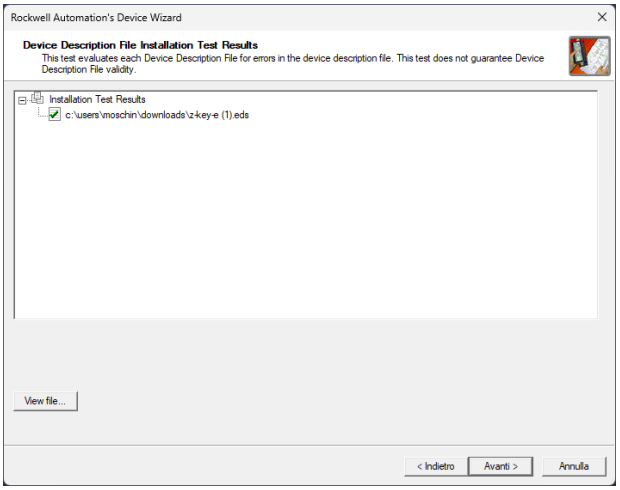


We select “Register a device description”:

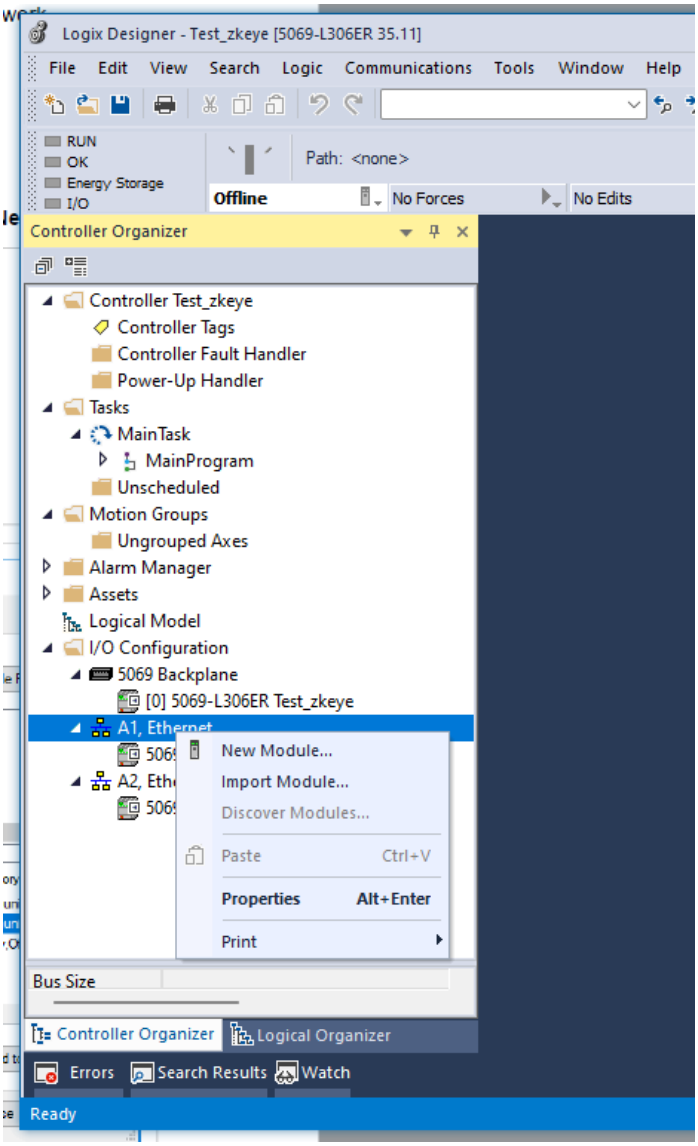


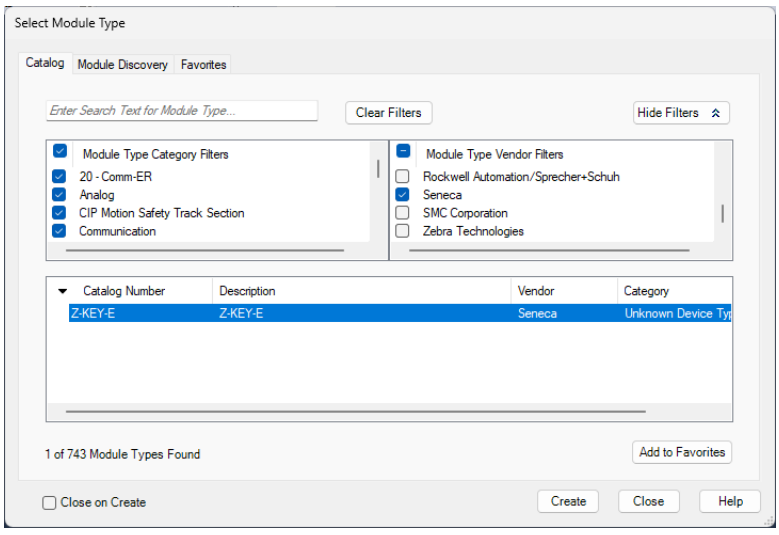
We select the EDS file exported from the webserver:



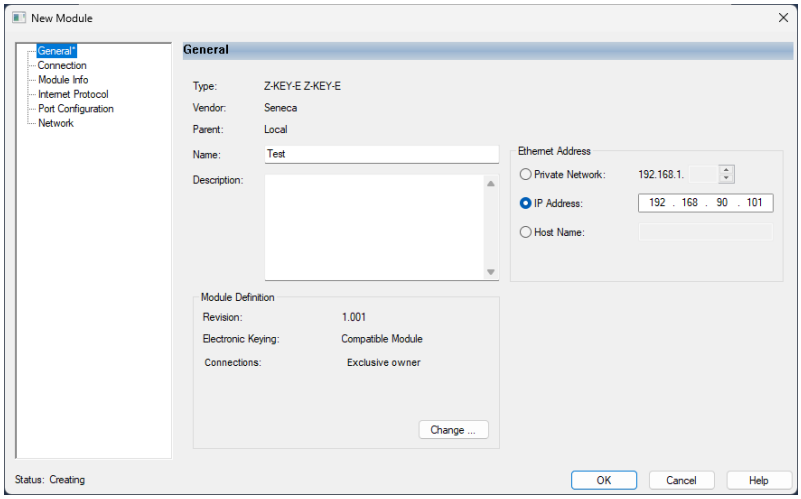


Now we insert the Seneca module by right clicking on the Ethernet port and selecting “New Module”:

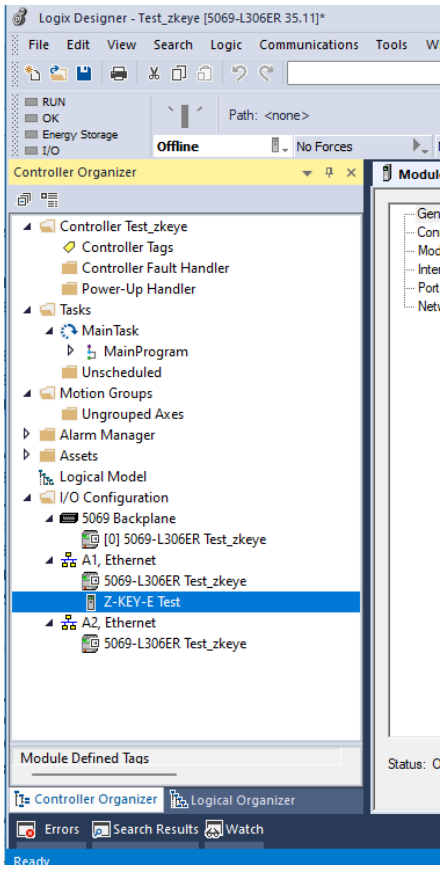




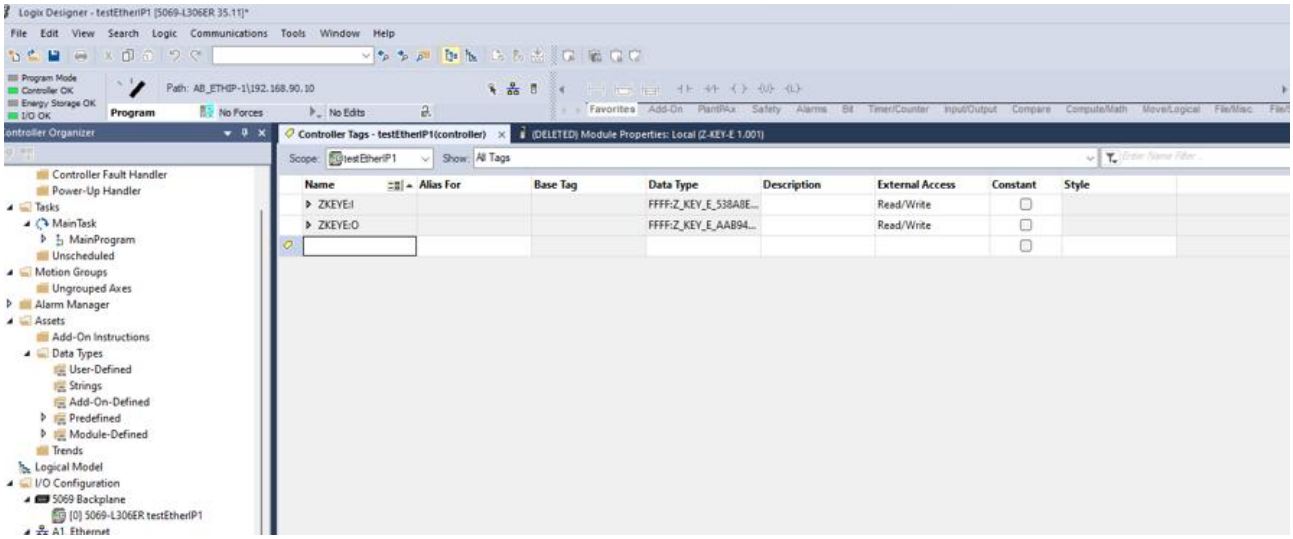
We configure it with the IP address chosen previously:



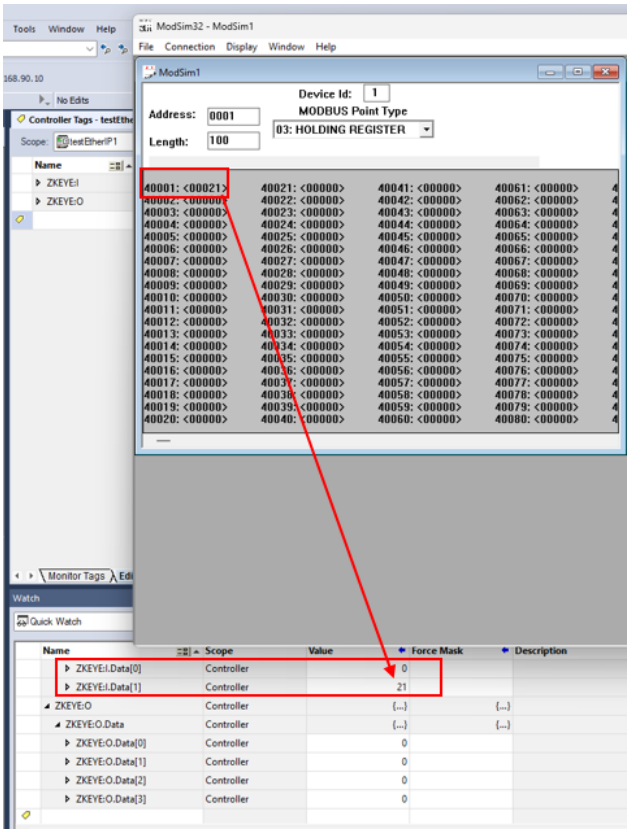
And we confirm with OK:



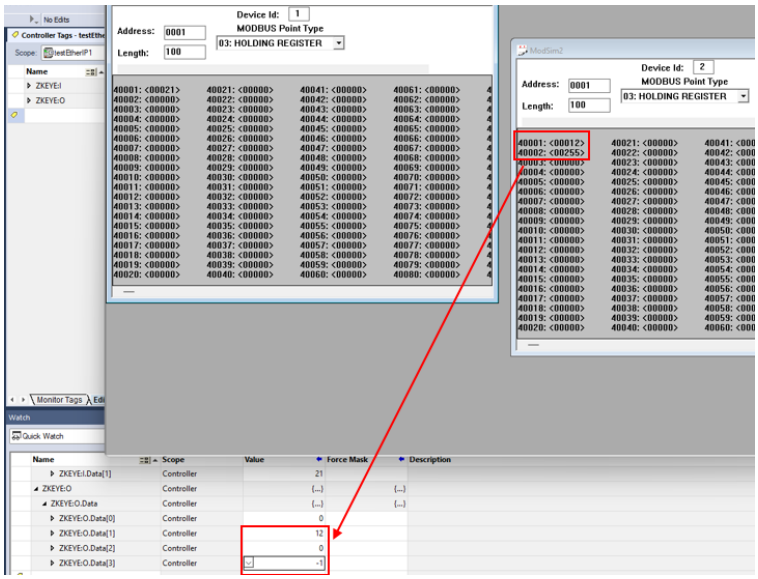
Having configured the gateway with 1 read register (2 bytes) and 2 write registers (4 bytes) you will have the following:
ZKEYE:I represents the 2 read bytes
ZKEYE:O represent the 4 write bytes:



The value read by modbus is visible in the ZKEYE:I bytes



While the writings are commanded by ZKEYE:O:



Where -1 in 2's complement equals byte 255

9. GATEWAY WEBSERVERS

9.1. "-P" GATEWAY WEBSERVER

9.1.1. WEBSERVER MODE AND PROFINET MODE

The device is normally in Profinet mode; in Profinet mode the device can be configured only through the Easy Setup 2 software.

In order to access the internal webserver it is necessary to put the device in Webserver mode using the Easy Setup2 or Seneca Device Discovery software, it is also possible to change the operating mode by pressing the button following the procedure.

9.1.2. MANUAL PROCEDURE FOR SWITCHING FROM PROFINET MODE TO WEBSERVER MODE AND VICE VERSA

To force webserver mode:

1) Turn on the device
2) Keep the PS1 button pressed until all LEDs turn off
3) Release the button
4) The device restarts and the "PWR" led flash slowly to indicate webserver mode

To force Profinet mode:

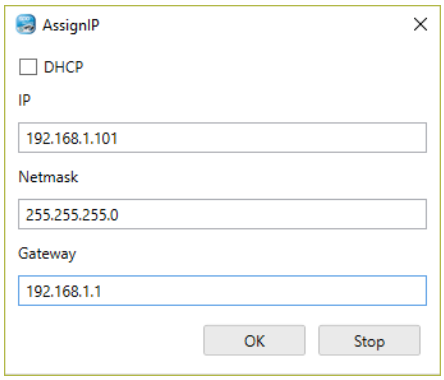
1) Turn on the device
2) Keep the PS1 button pressed until all LEDs turn off
3) Release the button
4) The device restarts and the "PWR" led flash slowly to indicate Profinet mode.

9.1.3. STEP BY STEP GUIDE FOR THE FIRST ACCESS TO THE WEBSERVER

STEP 1: POWER THE DEVICE AND CONNECT THE ETHERNET PORT, PUT THE DEVICE IN WEBSERVER MODE (SEE CHAPTER 9.1.1)

SENECA DISCOVERY DEVICE SOFTWARE STEP 2

Launch SCAN, select the device and press the "Assign IP" button, set a configuration compatible with your PC, for instance:



Confirm with OK. Now the device can be reached via Ethernet from your PC.

STEP 5 ACCESS TO THE CONFIGURATION WEBSERVER

ENTER your access credentials:
user: admin
password: admin

⚠ ATTENTION!

THE WEB BROWSERS WHICH HAVE BEEN TESTED FOR COMPATIBILITY WITH THE DEVICE WEBSERVER ARE:
MOZILLA FIREFOX AND GOOGLE CHROME.
THEREFORE, THE OPERATION WITH OTHER BROWSERS IS NOT GUARANTEED

9.1.4. WEBSERVER DEVICE CONFIGURATION

For further information on the access to the webserver of a new device, please refer to chapter 9.1.3.

ATTENTION!

THE WEB BROWSERS WHICH HAVE BEEN TESTED FOR COMPATIBILITY WITH THE DEVICE WEBSERVER ARE:
MOZILLA FIREFOX AND GOOGLE CHROME.
THEREFORE, THE OPERATION WITH OTHER BROWSERS IS NOT GUARANTEED

ATTENTION!

AFTER THE FIRST ACCESS CHANGE USER NAME AND PASSWORD IN ORDER TO PREVENT ACCESS TO THE DEVICE TO UNAUTHORIZED PEOPLE.

ATTENTION!

IF THE PARAMETERS TO ACCESS THE WEBSERVER HAVE BEEN LOST, TO ACCESS IT, IT IS NECESSARY TO GO THROUGH THE PROCEDURE TO RESET THE FACTORY-SET CONFIGURATION

9.1.5. WEBSERVER SECTIONS

The Webserver is divided into pages (sections) representing the various gateway functions:

Status
It is the section that displays the values of the configured tags in real time.

Setup
It is the section that allows the device basic configuration.

Setup Modbus Commands / Tags
It is the section that allows you to add/modify the Modbus commands or the tags (i.e. the variables) of the Modbus devices connected to the gateway.

I/O Mapping
In PROFINET IO / MODBUS MASTER GATEWAY mode only this is the section that allows you to export the current configuration into the GSDML file and to remap the bytes relating to the data coming from the Modbus protocol.

Firmware Update
This is the section that allows you to update the device firmware.

Database Update
It is the section that allows you to update the database of Modbus Seneca devices.

Serial Traffic Monitor
It allows to analyse the ModBUS frames of the serials.

9.1.6. "STATUS" SECTION

Depending on the selected operating mode, it displays:

PROFINET IO DEVICE / MODBUS MASTER GATEWAY
In the status section it is possible to view the mapping of the bytes associated with the registers coming from Modbus in real time.

PROFINET IO DEVICE / MODBUS SLAVE GATEWAY
In the status section it is possible to view the mapping of the bytes associated with the registers coming from Modbus in real time.

9.1.7. "SETUP" SECTION

DHCP (ETH) (default: Disabled)
Sets the DHCP client to get an IP address automatically.

STATIC IP* (default: 192.168.90.101*)
Sets the device static address. Careful not to enter devices with the same IP address into the same network.

STATIC IP MASK (default: 255.255.255.0)
Sets the mask for the IP network.

STATIC GATEWAY (default: 192.168.90.1)
Sets the gateway address.

WORKING MODE
Sets the operating mode:

TCP-IP PORT (default: 502)
Sets the communication port for the Modbus TCP-IP client protocol.

TCP-IP TIMEOUT [ms] (default 512 ms)
Sets the waiting time for a request to be considered in timeout.

PORT #1 MODBUS PROTOCOL (default RTU)
Sets the protocol on the serial between Modbus RTU or Modbus ASCII

PORT #2 MODBUS PROTOCOL (default RTU)
Sets the protocol on the serial between Modbus RTU or Modbus ASCII

PORT #1 BAUDRATE (default: 38400 baud)

Selects the communication speed of the COM #1 serial port

PORT #1 DATA BITS (default: 38400 baud)
Selects the communication speed of the COM #1 serial port

PORT #1 PARITY (default: None)
Sets the parity for the COM #1 serial communication port.

PORT #1 STOP BIT (default: 1)
Sets the number of stop bits for the COM #1 serial communication port.

PORT #1 TIMEOUT [ms]
Sets the wait time before defining fail.

PORT #1 WRITING RETRIES (default: 3)
Selects the number of writing attempts to be made on a serial slave before returning an error.

PORT #1 MAX READ NUM
Sets the maximum number of simultaneous serial reading ModBUS registers, the firmware will use this value to optimize the ModBUS readings.

PORT #1 MAX WRITE NUM
Sets the maximum number of simultaneous writing ModBUS registers of the serial, the firmware will use this value to optimize the ModBUS writings.

PORT #2 BAUDRATE (default: 38400 baud) (only for Z-KEY-P and Z-KEY-2ETH-P)
Selects the communication speed of the COM #2 serial port

PORT #2 DATA BITS (default: 38400 baud) (only for Z-KEY-P and Z-KEY-2ETH-P)
Selects the communication speed of the COM #2 serial port

PORT #2 PARITY (default: None) (only for Z-KEY-P and Z-KEY-2ETH-P)
Sets the parity for the COM #2 serial communication port.

PORT #2 STOP BIT (default: 1) (only for Z-KEY-P and Z-KEY-2ETH-P)
Sets the number of stop bits for the COM #2 serial communication port.

PORT# 2 TIMEOUT [ms] (only for Z-KEY-P and Z-KEY-2ETH-P)
Sets the wait time before defining fail.

PORT #2 WRITING RETRIES (default: 3) (only for Z-KEY-P and Z-KEY-2ETH-P)
Selects the number of writing attempts to be made on a serial slave before returning an error.

PORT #2 MAX READ NUM (only for Z-KEY-P and Z-KEY-2ETH-P)
Sets the maximum number of simultaneous reading ModBUS registers of the remote TCP-IP Modbus server, the firmware will use this value to optimize the ModBUS readings.

PORT #2 MAX WRITE NUM (only for Z-KEY-P and Z-KEY-2ETH-P)
Sets the maximum number of simultaneous writing ModBUS registers of the serial, the firmware will use this value to optimize the ModBUS writings.

WEB SERVER AUTHENTICATION USER NAME (default: admin)
Sets the username to access the webserver.

WEB SERVER PASSWORD (default: admin)
Sets the password to access the webserver and to read/write the configuration (if enabled).

WEB SERVER PORT (default: 80)
Sets the communication port for the web server.

IP CHANGE FROM DISCOVERY (default: Enabled)
Selects whether or not the device accepts the IP address change from the Seneca Discovery Device software.

PORT #1 AFTER FAIL DELAY [s]
Sets the number of quarantine seconds after a tag has been declared in fail (i.e. these tags are no longer considered) before being interrogated again.

PORT #2 AFTER FAIL DELAY [s] (only for Z-KEY-P and Z-KEY-2ETH-P)
Sets the number of quarantine seconds after a tag has been declared in fail (i.e. these tags are no longer considered) before being interrogated again.

PROFINET DEVICE NAME
Sets the name of the Profinet peripheral

MODBUS TCP-IP CLIENT
Enables or not the TCP-IP client Modbus

MODBUS TCP-IP SERVER#1…3 PORT
Sets the port for the max 3 remote TCP-IP Modbus servers

MODBUS TCP-IP SERVER#1…3 ADDRESS
Sets the IP address for the max 3 remote TCP-IP Modbus servers

MODBUS TCP-IP CLIENT TIMEOUT [ms]

Sets the timeout for remote TCP-IP Modbus servers

MODBUS TCP-IP CLIENT WRITING ATTEMPTS
Selects the number of writing attempts to be made on a remote TCP-IP Modbus server before returning an error and activating the quarantine.

MODBUS TCP-IP CLIENT MAX READ NUM
Sets the maximum number of simultaneous reading ModBUS registers of the remote TCP-IP Modbus server, the firmware will use this value to optimize the ModBUS readings.

MODBUS TCP-IP CLIENT MAX WRITE NUM
Sets the maximum number of simultaneous writing ModBUS registers of the remote TCP-IP Modbus server, the firmware will use this value to optimize the ModBUS writings.

SERVER AFTER FAIL DELAY
Sets the number of quarantine seconds after a tag has been declared in fail (i.e. these tags are no longer considered) before being interrogated again.

In addition, a configuration can be exported / imported via the webserver.

9.1.8. SAVING A CONFIGURATION ON A FILE

A configuration that includes:

CONFIGURATION
TAGS/COMMANDS

It can be saved to a file this way:

Go to the Setup section and select the file to save, press the "Save config" button

9.1.9. IMPORTING A CONFIGURATION FROM A FILE

A configuration that includes:

CONFIGURATION
TAGS/COMMANDS

It can be imported from a file this way:

Go to the Setup section and select the file to load, press the "Load config" button

9.1.10. "COMMANDS/TAGS" SECTION (ONLY FOR PROFINET IO / MODBUS MASTER GATEWAY MODE)

In this section you can add, edit or delete a tag.

Using the ADD button you can add a new command.
Using the CLONE button it is possible to clone an existing command.
Using the DELETE button it is possible to delete an existing command.
Using the MOVE UP/ DOWN button it is possible to move the command.
By holding down the CTRL button and clicking on one or more lines, it is possible to select multiple lines at the same time (they will turn yellow).

MODBUS COMMAND INDEX
It's the command index, firmware starts the execution form command index 1

MNEMONIC NAME
It is the identifying name of the command

TARGET MODBUS DEVICE
It represents the Seneca Modbus device selected from those available in the database.
In the case of a non-Seneca device, select CUSTOM.

TARGET RESOURCE
It represents the Seneca device variable you want to add.

TARGET CONNECTED TO
It selects the serial to be used for Modbus serial communication for the specified TAG.

TARGET MODBUS STATION ADDRESS
It selects the station address to use for the command.

TARGET MODBUS START REGISTER
It represents the starting Modbus address of the command (in the case of a Seneca device it is filled in automatically). Convention is 1-based type (for example, to access to the first register "Holding register 40001", enter 1)

TARGET MODBUS REQUEST TYPE
It represents the type of Modbus command to use (Holding Register, Coil etc.).
In the case of a Seneca device it is filled in automatically.

TARGET MODBUS TRIGGER
If the command is about writing, it allows you to select the writing technique on the Modbus side: Periodic, or Data change or both.

Periodic: writing is carried out continuously with the set time interval
Data Change: writing occurs only if the command registers change their values.
Periodic or data Change: combines the two previous modes.

TARGET MODBUS WRITE PERIODIC TIME [ms]
It represents the time interval of the periodic reading.

ENDIAN SWAP
Allows you to swap a register read by Modbus, i.e.:
NONE: no swap
BYTE: shifts the high byte with low byte (for example Modbus reading 0xAABB will be converted to 0xBBAA)
WORD: In the case of a data type greater than a Modbus register (e.g. single precision Floating Point registers) it allows you to set which word (register) to use as the most significant part, for example:
Register 1 = 0xAABB
Register 2 = 0xCCDD
will become a single value 0xAABBCCDD if the parameter is NONE, otherwise 0xCCDDAABB if this parameter is active
BYTE AND WORD: as in the previous case but there will also be a byte swap, for example:
Register 1 = 0xAABB
Register 2 = 0xCCDD
Will become 0xDDCCBBAA

9.1.11. "I/O MAPPING" SECTION

It allows exporting the GSDML file created (in the case of IO / Modbus Master Gateway mode) and to move the contents of the bytes of the read and write buffers.

9.1.12. "FIRMWARE UPDATE" SECTION

In order to improve, add, optimize the functions of the product, Seneca releases firmware updates on the device section on the www.seneca.it website

⚠ ATTENTION!
NOT TO DAMAGE THE DEVICE DO NOT REMOVE THE POWER SUPPLY DURING THE FIRMWARE UPDATE OPERATION.

9.1.13. "DATABASE UPDATE" SECTION

Seneca releases new Database files of its updated modbus devices on the "KEY" device section of the www.seneca.it website.
To update the database, select the file and press the "Update Database" button.
The device is already updated at the factory with the most recent database at the time of production

9.1.14. SERIAL "SERIAL TRAFFIC MONITOR"

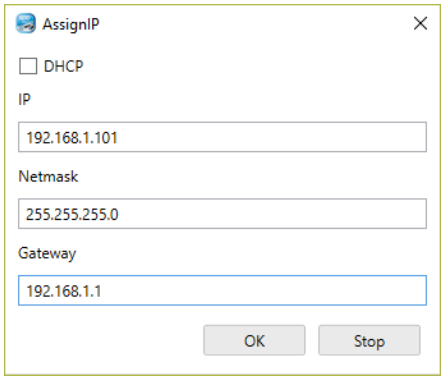
Allows you to view the serial packets that are in transit.

9.2. "-E" GATEWAY WEBSERVER

9.2.1. STEP BY STEP GUIDE FOR THE FIRST ACCESS TO THE WEBSERVER

STEP 1: POWER THE DEVICE AND CONNECT THE ETHERNET PORT, PUT THE DEVICE IN WEBSERVER MODE (SEE CHAPTER 9.1.1)

SENECA DISCOVERY DEVICE SOFTWARE STEP 2
Launch SCAN, select the device and press the "Assign IP" button, set a configuration compatible with your PC, for instance:



Confirm with OK. Now the device can be reached via Ethernet from your PC.

STEP 5 ACCESS TO THE CONFIGURATION WEBSERVER

ENTER your access credentials:
user: admin
password: admin

⚠ ATTENTION!
THE WEB BROWSERS WHICH HAVE BEEN TESTED FOR COMPATIBILITY WITH THE DEVICE WEBSERVER ARE:
MOZILLA FIREFOX AND GOOGLE CHROME.
THEREFORE, THE OPERATION WITH OTHER BROWSERS IS NOT GUARANTEED

9.2.2. WEBSERVER DEVICE CONFIGURATION

For further information on the access to the webserver of a new device, please refer to chapter 9.1.3.

ATTENTION!

THE WEB BROWSERS WHICH HAVE BEEN TESTED FOR COMPATIBILITY WITH THE DEVICE WEBSERVER ARE:
MOZILLA FIREFOX AND GOOGLE CHROME.
THEREFORE, THE OPERATION WITH OTHER BROWSERS IS NOT GUARANTEED

ATTENTION!

AFTER THE FIRST ACCESS CHANGE USER NAME AND PASSWORD IN ORDER TO PREVENT ACCESS TO THE DEVICE TO UNAUTHORIZED PEOPLE.

ATTENTION!

IF THE PARAMETERS TO ACCESS THE WEBSERVER HAVE BEEN LOST, TO ACCESS IT, IT IS NECESSARY TO GO THROUGH THE PROCEDURE TO RESET THE FACTORY-SET CONFIGURATION

9.2.3. WEBSERVER SECTIONS

The Webserver is divided into pages (sections) representing the various gateway functions:

Status
It is the section that displays the values of the configured tags in real time.

Setup
It is the section that allows the device basic configuration.

Setup Modbus Commands / Tags
It is the section that allows you to add/modify the Modbus commands or the tags (i.e. the variables) of the Modbus devices connected to the gateway.

I/O Mapping
This is the section that allows you to export the current configuration into the EDS file and to remap the bytes relating to the data coming from the Modbus protocol.

Firmware Update
This is the section that allows you to update the device firmware.

Database Update
It is the section that allows you to update the database of Modbus Seneca devices.

Serial Traffic Monitor
It allows to analyse the ModBUS frames of the serials.

9.2.4. "STATUS" SECTION

In the status section it is possible to view the mapping of the bytes associated with the registers coming from Modbus in real time and to export the EDS file from the current configuration.

9.2.5. "SETUP" SECTION

DHCP (ETH) (default: Disabled)
Sets the DHCP client to get an IP address automatically.

***STATIC IP* (default: 192.168.90.101)**
Sets the device static address. Careful not to enter devices with the same IP address into the same network.

STATIC IP MASK (default: 255.255.255.0)
Sets the mask for the IP network.

STATIC GATEWAY (default: 192.168.90.1)
Sets the gateway address.

WORKING MODE
Sets the operating mode:

TCP-IP PORT (default: 502)
Sets the communication port for the Modbus TCP-IP client protocol.

TCP-IP TIMEOUT [ms] (default 512 ms)
Sets the waiting time for a request to be considered in timeout.

PORT #1 MODBUS PROTOCOL (default RTU)
Sets the protocol on the serial between Modbus RTU or Modbus ASCII

PORT #2 MODBUS PROTOCOL (default RTU)
Sets the protocol on the serial between Modbus RTU or Modbus ASCII

PORT #1 BAUDRATE (default: 38400 baud)
Selects the communication speed of the COM #1 serial port

PORT #1 DATA BITS (default: 38400 baud)
Selects the communication speed of the COM #1 serial port

PORT #1 PARITY (default: None)
Sets the parity for the COM #1 serial communication port.

PORT #1 STOP BIT (default: 1)
Sets the number of stop bits for the COM #1 serial communication port.

PORT #1 TIMEOUT [ms]
Sets the wait time before defining fail.

PORT #1 WRITING RETRIES (default: 3)
Selects the number of writing attempts to be made on a serial slave before returning an error.

PORT #1 MAX READ NUM
Sets the maximum number of simultaneous serial reading ModBUS registers, the firmware will use this value to optimize the ModBUS readings.

PORT #1 MAX WRITE NUM
Sets the maximum number of simultaneous writing ModBUS registers of the serial, the firmware will use this value to optimize the ModBUS writings.

PORT #2 BAUDRATE (default: 38400 baud) (only for Z-KEY-E and Z-KEY-2ETH-E)
Selects the communication speed of the COM #2 serial port

PORT #2 DATA BITS (default: 38400 baud) (only for Z-KEY-E and Z-KEY-2ETH-E)
Selects the communication speed of the COM #2 serial port

PORT #2 PARITY (default: None) (only for Z-KEY-E and Z-KEY-2ETH-E)
Sets the parity for the COM #2 serial communication port.

PORT #2 STOP BIT (default: 1) (only for Z-KEY-E and Z-KEY-2ETH-E)
Sets the number of stop bits for the COM #2 serial communication port.

PORT #2 TIMEOUT [ms] (only for Z-KEY-E and Z-KEY-2ETH-E)
Sets the wait time before defining fail.

PORT #2 WRITING RETRIES (default: 3) (only for Z-KEY-E and Z-KEY-2ETH-E)
Selects the number of writing attempts to be made on a serial slave before returning an error.

PORT #2 MAX READ NUM (only for Z-KEY- E and Z-KEY-2ETH- E)
Sets the maximum number of simultaneous reading ModBUS registers of the remote TCP-IP Modbus server, the firmware will use this value to optimize the ModBUS readings.

PORT #2 MAX WRITE NUM (only for Z-KEY- E and Z-KEY-2ETH- E)

Sets the maximum number of simultaneous writing ModBUS registers of the serial, the firmware will use this value to optimize the ModBUS writings.

WEB SERVER AUTHENTICATION USER NAME (default: admin)
Sets the username to access the webserver.

WEB SERVER PASSWORD (default: admin)
Sets the password to access the webserver and to read/write the configuration (if enabled).

WEB SERVER PORT (default: 80)
Sets the communication port for the web server.

IP CHANGE FROM DISCOVERY (default: Enabled)
Selects whether or not the device accepts the IP address change from the Seneca Discovery Device software.

PORT #1 AFTER FAIL DELAY [s]
Sets the number of quarantine seconds after a tag has been declared in fail (i.e. these tags are no longer considered) before being interrogated again.

PORT #2 AFTER FAIL DELAY [s] (only for Z-KEY- E and Z-KEY-2ETH- E)
Sets the number of quarantine seconds after a tag has been declared in fail (i.e. these tags are no longer considered) before being interrogated again.

MODBUS TCP-IP CLIENT
Enables or not the TCP-IP client Modbus

MODBUS TCP-IP SERVER#1…3 PORT
Sets the port for the max 3 remote TCP-IP Modbus servers

MODBUS TCP-IP SERVER#1…3 ADDRESS
Sets the IP address for the max 3 remote TCP-IP Modbus servers

MODBUS TCP-IP CLIENT TIMEOUT [ms]
Sets the timeout for remote TCP-IP Modbus servers

MODBUS TCP-IP CLIENT WRITING ATTEMPTS
Selects the number of writing attempts to be made on a remote TCP-IP Modbus server before returning an error and activating the quarantine.

MODBUS TCP-IP CLIENT MAX READ NUM
Sets the maximum number of simultaneous reading ModBUS registers of the remote TCP-IP Modbus server, the firmware will use this value to optimize the ModBUS readings.

MODBUS TCP-IP CLIENT MAX WRITE NUM

Sets the maximum number of simultaneous writing ModBUS registers of the remote TCP-IP Modbus server, the firmware will use this value to optimize the ModBUS writings.

SERVER AFTER FAIL DELAY

Sets the number of quarantine seconds after a Modbus command has been declared in fail (i.e. This command is no longer executed) before being interrogated again.

ETHERIP O->T RUN/IDLE HEADER

This option adds a 32-bit header for each class 1 packet sent from the Observer to the Target. Bit 0 indicates the RUN or IDLE status of the device.

ETHERIP T->O RUN/IDLE HEADER

This option adds a 32-bit header for each class 1 packet sent from the Target to the Observer. Bit 0 indicates the RUN or IDLE status of the device.

ETHERIP VENDOR ID

It allows to customise the Vendor ID in the EDS file.

ETHERIP DEVICE TYPE

It allows to customise the Device Type in the EDS file.

ETHERIP PRODUCT CODE

It allows to customise the Product Code in the EDS file.

ETHERIP MAJOR REVISION

It allows to customise the Major Revision in the EDS file.

ETHERIP MINOR REVISION

It allows to customise the Minor Revision in the EDS file.

DIAGNOSTIC

It allows you to activate or not the 9 bytes of Modbus diagnostics. The diagnostic bytes are inserted at the end of the reading area.

STOP MODBUS READING WHEN NO ETHERNET IP CONNECTION

If the connection with the PLC is lost, the device stops polling the Modbus registers and therefore allows any safety timeouts to be triggered on the outputs.

In addition, a configuration can be exported / imported via the webserver.

9.2.6. SAVING A CONFIGURATION ON A FILE

A configuration that includes:

CONFIGURATION
TAGS/COMMANDS

It can be saved to a file this way:

Go to the Setup section and select the file to save, press the "Save config" button

9.2.7. IMPORTING A CONFIGURATION FROM A FILE

A configuration that includes:

CONFIGURATION
TAGS/COMMANDS

It can be imported from a file this way:

Go to the Setup section and select the file to load, press the "Load config" button

9.2.8. "COMMANDS/TAGS" SECTION

In this section you can add, edit or delete a tag.

Using the ADD button you can add a new command.
Using the CLONE button it is possible to clone an existing command.
Using the DELETE button it is possible to delete an existing command.
Using the MOVE UP/ DOWN button it is possible to move the command.
By holding down the CTRL button and clicking on one or more lines, it is possible to select multiple lines at the same time (they will turn yellow).

MODBUS COMMAND INDEX
It's the command index, firmware starts the execution form command index 1

MNEMONIC NAME
It is the identifying name of the command

TARGET MODBUS DEVICE
It represents the Seneca Modbus device selected from those available in the database.
In the case of a non-Seneca device, select CUSTOM.

TARGET RESOURCE
It represents the Seneca device variable you want to add.

TARGET CONNECTED TO
It selects the serial to be used for Modbus serial communication for the specified TAG.

TARGET MODBUS STATION ADDRESS
It selects the station address to use for the command.

TARGET MODBUS START REGISTER
It represents the starting Modbus address of the command (in the case of a Seneca device it is filled in automatically). Convention is 1-based type (for example, to access to the first register "Holding register 40001", enter 1)

TARGET MODBUS REQUEST TYPE
It represents the type of Modbus command to use (Holding Register, Coil etc.).
In the case of a Seneca device it is filled in automatically.

TARGET MODBUS TRIGGER
If the command is about writing, it allows you to select the writing technique on the Modbus side: Periodic, or Data change or both.

Periodic: writing is carried out continuously with the set time interval
Data Change: writing occurs only if the command registers change their values.
Periodic or data Change: combines the two previous modes.

TARGET MODBUS WRITE PERIODIC TIME [ms]
It represents the time interval of the periodic reading.
ENDIAN SWAP
Allows you to swap a register read by Modbus, i.e.:
NONE: no swap
BYTE: shifts the high byte with low byte (for example Modbus reading 0xAABB will be converted to 0xBBAA)
WORD: In the case of a data type greater than a Modbus register (e.g. single precision Floating Point registers) it allows you to set which word (register) to use as the most significant part, for example:
Register 1 = 0xAABB
Register 2 = 0xCCDD
will become a single value 0xAABBCCDD if the parameter is NONE, otherwise 0xCCDDAABB if this parameter is active
BYTE AND WORD: as in the previous case but there will also be a byte swap, for example:
Register 1 = 0xAABB
Register 2 = 0xCCDD
Will become 0xDDCCBBAA

9.2.9. "I/O MAPPING" SECTION

Allows you to move the contents of the bytes of the read and write buffers.

9.2.10. "FIRMWARE UPDATE" SECTION

In order to improve, add, optimize the functions of the product, Seneca releases firmware updates on the device section on the www.seneca.it website

⚠ ATTENTION!
NOT TO DAMAGE THE DEVICE DO NOT REMOVE THE POWER SUPPLY DURING THE FIRMWARE UPDATE OPERATION.

9.2.11. "DATABASE UPDATE" SECTION

Seneca releases new Database files of its updated Modbus devices on the device section of the www.seneca.it website.
To update the database, select the file and press the "Update Database" button.
The device is already updated at the factory with the most recent database at the time of production

9.2.12. SERIAL "SERIAL TRAFFIC MONITOR"

Allows you to view the serial packets that are in transit.

10. RESETTING THE DEVICE TO ITS FACTORY CONFIGURATION

The factory configuration removes all configured commands and resets all parameters to default.

To reset the device to the factory configuration it is necessary to follow the procedure below:

Z-KEY-P/E / Z-KEY-2ETH-P/E:

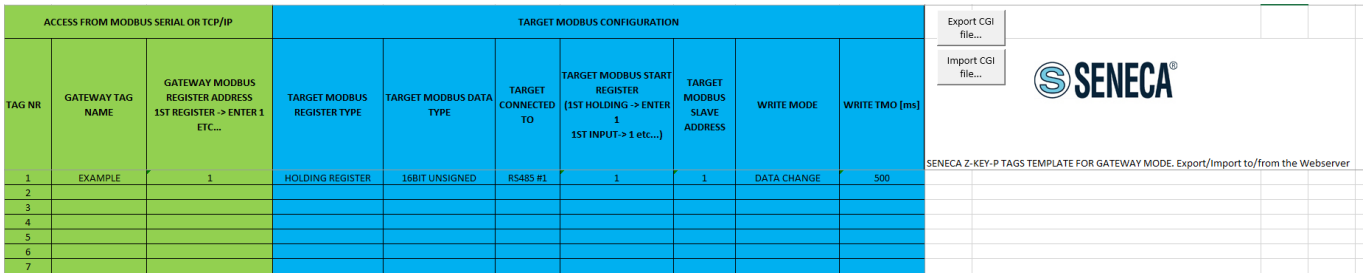
1) Remove power from the device
2) Turn dip switches 1 and 2 to ON
3) Power up the device and wait at least 10 seconds
4) Remove power from the device
5) Turn dip switches 1 and 2 to OFF
6) At the next restart the device will have loaded the factory configuration

R-KEY-LT-P/E:

1) Remove power from the device
2) Turn 2 SW2 dip switches to ON
3) Power up the device and wait at least 10 seconds
4) Remove power from the device
5) Turn 2 SW2 dip switches to OFF.
6) At the next restart the device will have loaded the factory configuration

11. EXCEL TEMPLATE

Excel templates are available on the Seneca website.



ACCESS FROM MODBUS SERIAL OR TCP/IP			TARGET MODBUS CONFIGURATION						
TAG NR	GATEWAY TAG NAME	GATEWAY MODBUS REGISTER ADDRESS 1ST REGISTER -> ENTER 1 ETC...	TARGET MODBUS REGISTER TYPE	TARGET MODBUS DATA TYPE	TARGET CONNECTED TO	TARGET MODBUS START REGISTER (1ST HOLDING -> ENTER 1 1ST INPUT-> 1 etc...)	TARGET MODBUS SLAVE ADDRESS	WRITE MODE	WRITE TMO [ms]
1	EXAMPLE	1	HOLDING REGISTER	16BIT UNSIGNED	RS485 #1	1	1	DATA CHANGE	500
2									
3									
4									
5									
6									
7									

Export CGI file...
Import CGI file...
SENECA
SENECA Z-KEY-P TAGS TEMPLATE FOR GATEWAY MODE. Export/Import to/from the Webserver

These allow you to quickly add TAGs to an Excel sheet and import them into the web pages of the devices. It is also possible to export the TAGs from the web page to the Excel sheet.

12. SUPPORTED MODBUS COMMUNICATION PROTOCOLS

The Modbus communication protocols supported are:

- Modbus RTU/ASCII master (from #1 and #2 serial ports)
- Modbus RTU/ASCII slave (from #1 and #2 serial ports)
- Modbus TCP-IP Client (from the Ethernet port) up to 3 remote TCP-IP Modbus Servers

For more information on these protocols, see the website:
http://www.modbus.org/specs.php.

12.1. SUPPORTED MODBUS FUNCTION CODES

The following Modbus functions are supported:

- Read Coils (function 1)
- Read Discrete Inputs (function 2)
- Read Holding Registers (function 3)
- Read Input Registers (function 4)
- Write Single Coil (function 5)
- Write Single Register (function 6)
- Write multiple Coils (function 15)
- Write Multiple Registers (function 16)

⚠ ATTENTION!

All 32-bit variables are contained in 2 consecutive Modbus registers

All 64-bit variables are contained in 4 consecutive Modbus registers

13. MODBUS DIAGNOSTICS

Diagnostics management takes the timeout or exceptions to the Modbus requests into account.
9 Bytes are made available for diagnostics:

GLOBAL DIAGNOSTIC READ BYTE (1 byte)
PORT#1 DIAGNOSTIC MODBUS DEVICE ADDRESS (4 byte)
PORT#2 DIAGNOSTIC MODBUS DEVICE ADDRESS (4 byte)

⚠ ATTENTION!
The diagnostic bytes are inserted at the end of the configured reading area

In particular the BIT have the following meaning:

If the BYTE[0] is 0 -> No Error
If the BYTE[0] is 1 -> At least one device is in fail

The other bytes indicate which station address on the serials is in fail for the serial port 1 or 2;

On BYTE[1], BYTE [2], BYTE [3], BYTE [4]
The first 4 Modbus addresses of the failing devices in Modbus port 1 from lowest to highest address are shown

On BYTE[5], BYTE [6], BYTE [7], BYTE [8]
The first 4 Modbus addresses of the failing devices in Modbus port 2 from lowest to highest address are shown

For example if devices with the following station address are connected to serial port #1: 1, 8, 15, 24 and the station 15 and 24 are in fail it will be:

BYTE [0]-> 1
BYTE[1] -> 15
BYTE[2] -> 24
BYTE [3]-> 0
BYTE [4]-> 0
It is possible to export the TAGS from the device web page and import them into the excel page.

14. INFORMATION ABOUT MODBUS REGISTERS

The following abbreviations are used in the following chapter:

MS	Most Significant
LS	Least Significant
MSBIT	Most Significant Bit
LSBIT	Least Significant Bit
MMSW	"Most" Most Significant Word (16bit)
MSW	Most Significant Word (16bit)
LSW	Least Significant Word (16bit)
LLSW	"Least" Least Significant Word (16bit)
RO	Read Only
RW*	Read-Write: REGISTERS CONTAINED IN FLASH MEMORY: WRITABLE ABOUT 10,000 TIMES MAXIMUM
RW**	Read-Write: REGISTERS THAT CAN BE WRITTEN ONLY AFTER WRITING THE COMMAND "ENABLE WRITE CUSTOM ENERGIES = 49616"
UNSIGNED 16 BIT	Unsigned integer register that can assume values from 0 to 65535
SIGNED 16 BIT	Signed integer register that can take values from -32768 to +32767
UNSIGNED 32 BIT	Unsigned integer register that can assume values from 0 to 4294967296
SIGNED 32 BIT	Signed integer register that can take values from -2147483648 to 2147483647
UNSIGNED 64 BIT	Unsigned integer register that can assume values from 0 to 18446744073709551615
SIGNED 64 BIT	Signed integer register that can assume values from -2^63 to 2^63-1
FLOAT 32 BIT	32-bit, single-precision floating-point register (IEEE 754) https://en.wikipedia.org/wiki/IEEE_754
BIT	Boolean register, which can take the values 0 (false) or 1 (true)

14.1. 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

14.2. 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".***

14.3. 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

14.4. 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

14.5. 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							

14.6. *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															

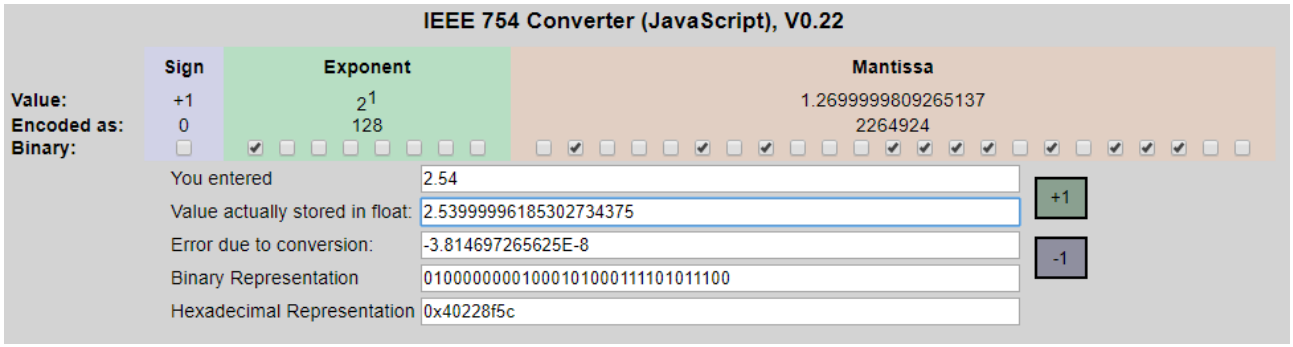
$$Value_{32bit} = Register_{LSW} + (Register_{MSW} * 65536)$$

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.

14.7. 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).