lartet@lartet.com
www.lartet.com

OPERATION MANUAL

OF DISPLAYS

DN-107/V2

INDEX

INDEX 1-1
1 INTRODUCTION 1-1
2 GENERAL CHARACTERISTICS 2-1
2.1 General characteristics of the displays 2-1
2.1.1 General characteristics of the DN-107 displays 2-1
2.2 Dimensions and fixation of the displays 2-1
2.2.1 Dimensions of the DN-107/3 displays 2-1
2.2.2 Dimensions of the DN-107/4 displays 2-2
2.3 Fixing the DN-107 Displays 2-2
3 INSTALLATION 3-3
3.1 Locating Device Connectors 3-3
3.2 Power Connection 3-4
3.3 Serial Line Connection 3-5
3.3.1 RS-232 connection between a PC and a DN display 3-5
3.3.2 RS-485 connection between 3 DNs and a PC 3-6
3.4 Connecting the Ethernet Line 3-7
4 DISPLAY INITIALIZATION 4-1
4.1 Initial start-up 4-1
4.1.1 Setup with Display Discoverer 4-2
4.2 Display Settings 4-3
4.2.1 Overview of the set 4-4
4.2.2 General Settings 4-5
4.2.3 Wired Network Settings 4-7
4.2.4 WIFI wireless network settings 4-8
4.2.5 Communication Settings 4-9
5 WORK OPERATIONS 5-10
5.1 Work operations and accepted data types 5-10
6 ETHERNET BUS COMMUNICATION 6-1
6.1 Ethernet Communication Settings 6-1
6.2 MODBUS/TCP protocol 6-3
6.2.1 MODBUS Functions 6-3
6.2.2 Writing Registers 6-10
6.2.3 Coils Writing 6-13
6.2.4 Reading registers and *coils* 6-13

6.3 TCP/IP protocol 6-14
6.4 UDP Protocol 6-15
7 WIFI COMMUNICATION 7-15
7.1 WIFI Communication Settings 7-15
8 SERIAL BUS COMMUNICATION 8-1
8.1 Serial Communication Settings 8-1
8.2 KOSMOS (ASCII) and KOSMOS (ISO 1745) protocols 8-4
8.3 MODBUS RTU protocol 8-4
8.4 ASCII Protocol 8-5
8.4.1 ASCII Protocol Examples 8-6
9 PROFINET COMMUNICATION 9-1
9.1 PROFINET Communication Settings 9-1
9.2 Types and formats of data in Profinet communication 9-2
9.2.1 Float format 9-3
9.2.2 Word Format 9-4
9.2.3 ASCII format 9-4
9.2.4 Encoding characters in ASCII format 9-5
9.2.5 Practical Coding Examples 9-5
9.3 LED Indicators 9-6
10 DIGITAL INPUTS AND ASSOCIATED FUNCTIONS 10-7
10.1 Configuring Digital Inputs 10-8
10.2 Functions associated with digital inputs 10-12
10.2.1 Counter 10-12
10.2.2 Stopwatch 10-13
10.2.3 Tachometer 10-14
10.2.4 BCD Direct 10-15
11 RESET FACTORY SETTINGS 11-1
12 UPDATE THE VISUALIZER 12-1
ANNEX 1: Sending information with "Hercules" for TCP, UDP and serial communication 12-1
ANNEX 2: Sending Information with "QModMaster" for MODBUS TCP and MODBUS RTU Communication 12-1
ANNEX 3: Configure and use function blocks to send information using a PLC. 12-1

1 INTRODUCTION

The DN-107 **series numerical displays** are industrial numerical data display displays, which are presented with 3 or 4 red digits and different control buses.

They can be controlled by **RS-232/RS-485 serial BUS (RS422 Option),** by **Ethernet/Wifi network, Profinet** (optionally) or **Multifunction** (tachometer, counter, stopwatch...). It can be configured to work with different protocols, see more below.

They are manufactured on four display sides. The height of the digits is 57mm allowing a reading distance of 30 meters.

The field of application of these displays is very wide in applications where it is necessary to indicate numerical values resulting from industrial processes sent from a PLC/PC through the communication options available on the device.

The configuration of all parameters and protocols is done via a web server at the IP address defined by the user (by default the IP address is set to **10.30.90.11**).

2 GENERAL CHARACTERISTICS

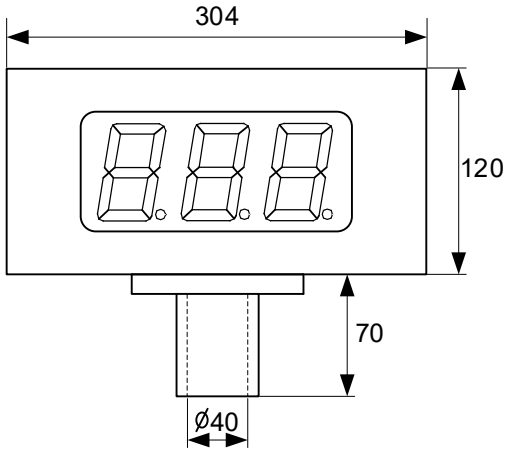
2.1 General characteristics of the displays

2.1.1 General characteristics of the DN-107 displays

Supply voltage .....80 to 240 VAC 50/60Hz. 24VDC option.
Consumption .....35Va.
Display .....7 segments of 57 mm height + decimal point.
.....Red LED. Reading distance 30 meters
Parameter memory .....Eeprom.
Environmental conditions .....Working temperature: -20 to 60ºC.
.....Storage temperature: -30ºC to 70ºC
.....Humidity: 20-90% RH non-condensing.
.....Maximum ambient lighting: 1000 lux.
.....Protection: IP41.

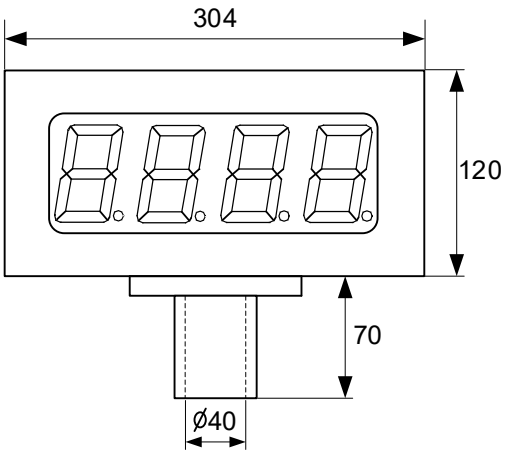
2.2 Dimensions and fixation of the displays

2.2.1 Dimensions of the DN-107/3 displays



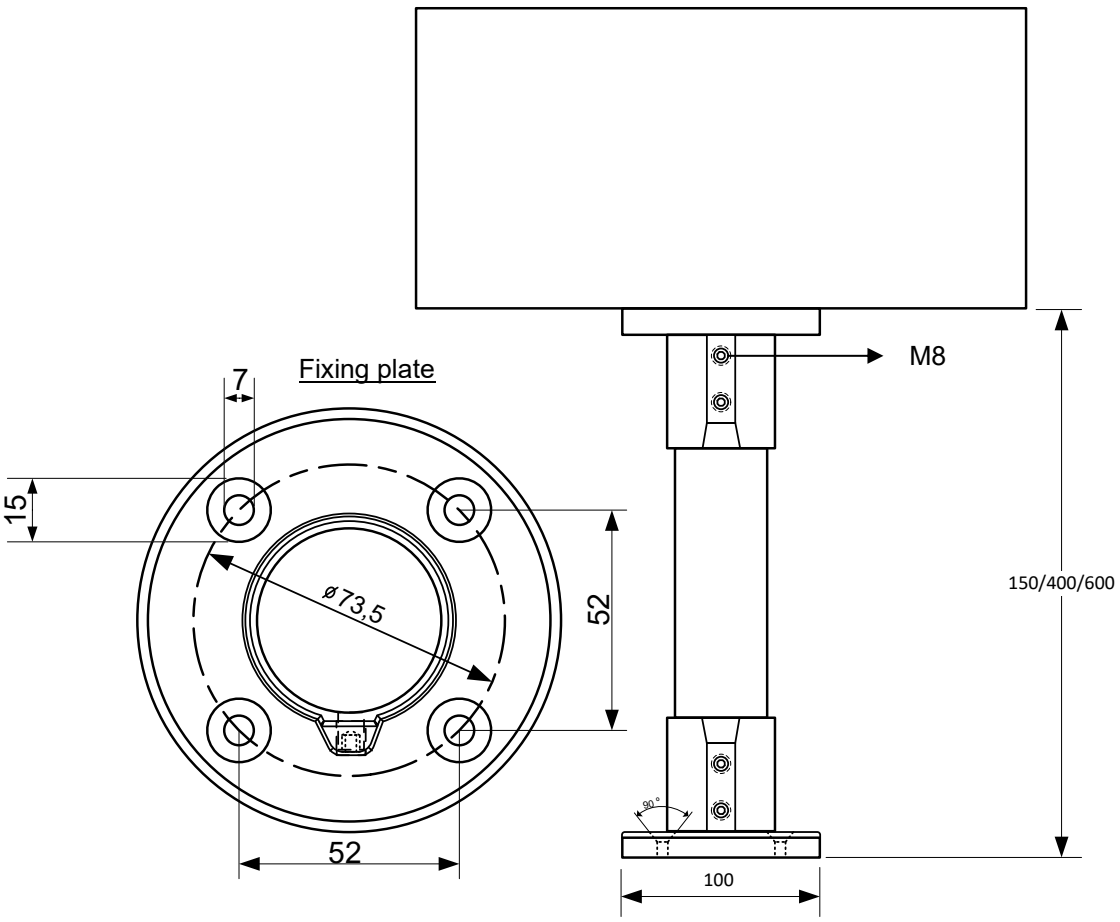
All measurements are in millimeters

2.2.2 Dimensions of the DN-107/4 displays



All measurements are in millimeters

2.3 Fixing the DN-107 Displays



All measurements are in millimeters

3 INSTALLATION

The installation of **the DN-107** is not particularly delicate, but some important considerations must be considered.

They must not be anchored in places subject to vibration, or in places that generally exceed the limits specified in the characteristics of the display, both in temperature and humidity.

The degree of protection of the **DN-107 display displays** is IP41, which means that it is protected against the penetration of foreign solid objects with a diameter greater than 1 mm, and against the vertical fall of water droplets

DN-107 **displays** should not be installed in locations where the illumination level is higher than 1000 lux. Nor should the direct incidence of the sun's rays on the display be allowed as this would lose visibility.

In the electrical installation, proximity to lines with very high currents, high voltage lines, as well as High Frequency generators and U/F converters for motors must be avoided.

3.1 Locating Device Connectors

The power supply device connectors are located on the inside of the display.

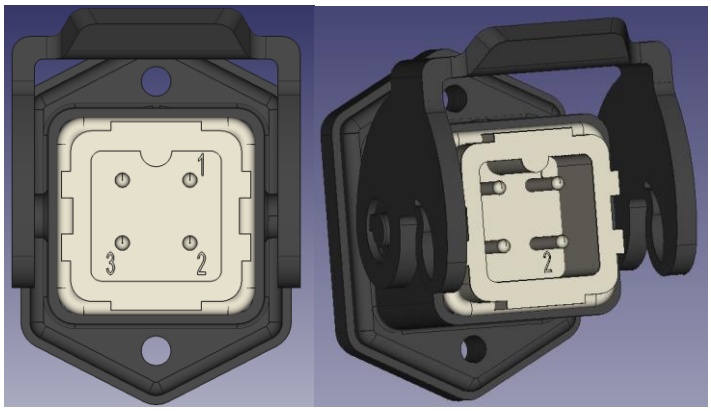


Fig. 1: Display power connector.

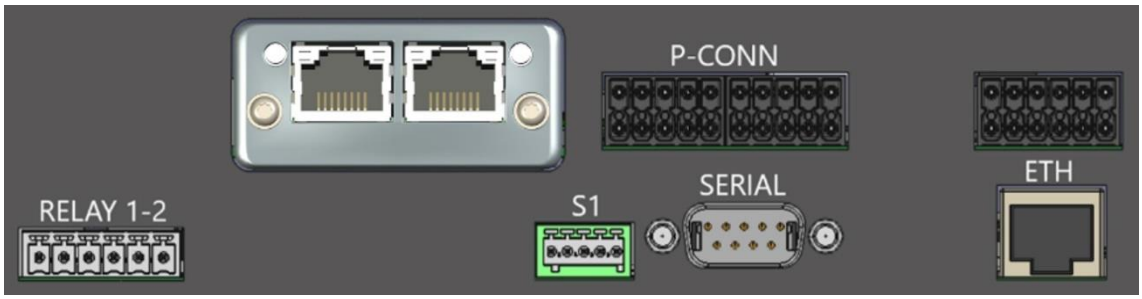


Fig. 2: Visualizer data transfer connectors.

⚠ **The image shown does not correspond to reality, it is only an illustrative example.**

Connectors Schematic:

- **ETH.** Ethernet.
- **SERIAL.** DB-9 connector. The connection diagrams can be seen in section 3.4.
- **S1.** Connector for digital detection probes. (particles, humidity, temperature...) (Not enabled)
- **RELAY 1-2.** Output of actuator relays.

1 = Relé 1 - NO 4 = Relé 2 - NO
2 = Relé 1 - C 5 = Relé 2 - C
3 = Relé 1 - NC 6 = Relé 2 - NC

- **P-CONN.** Multifunction parallel input. (DEVICE OPTIONS)
- **Profinet.** Profinet industrial bus connector.

Figure 4 corresponds to the maximum number of connectors in the display, depending on the characteristics of your device you may not have all connectors.

3.2 Power Connection

Power should be 80 to 240 VAC, 50/60 Hz or 24VDC, with the 24V option.

The section of the supply conductors will be according to consumption, the earth conductor will have a minimum section of 1.5 mm².

The 220 VAC power connector has 4 contacts and is located on the inside of the device. Connect the power cords following the diagram below.

The 24VDC power connector has 5 contacts and is located on the inside of the device. Connect the power cords following the diagram below.

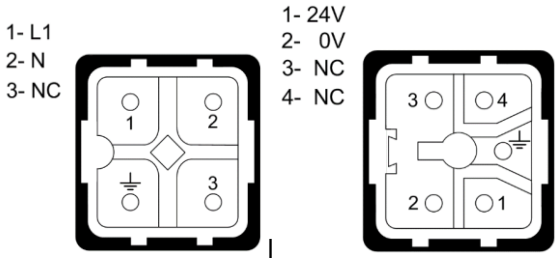


Fig. 3: On the left, diagram of the 220VAC power connector with 4 contacts. On the right, diagram of the 24VDC power connector with 5 contacts.

3.3 Serial Line Connection

The DN-107 series displays support two types of serial line connection: RS-232 and RS-485. Or RS-232 and RS-422, if the device has been purchased with the RS-422 option.

The serial line type is selected using the display's web server. See section 4.2.2 .

The connection scheme is shown below:

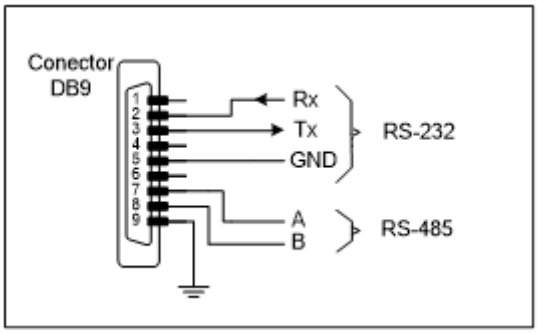


Fig. 4: Diagram of the RS-232/RS-485 connection.

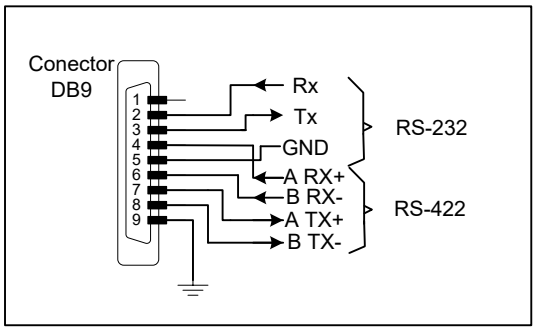


Fig. 5: Diagram of the RS-232/RS-422 connection.

Both options use the same connector, type DB-9, located at the bottom of the device.

3.3.1 RS-232 connection between a PC and a DN display

Using the RS-232 line, the total length of the cable should not be more than 15 meters (with communication speed at 9600 bps)

It is important for signal integrity to use shielded cable and connect the shielded cable to pin 9 of the DB9 connector.

The data cable should be placed away from high-voltage lines. The connector shown corresponds to the cable connector.

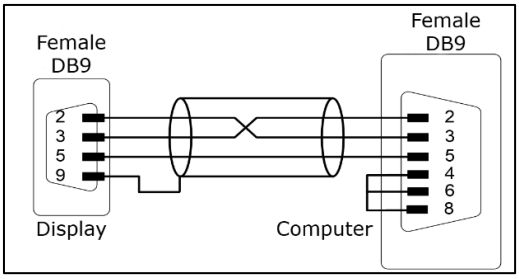


Fig. 6: Schematic of the RS-232 connection between a DN-107 display and a PC.

3.3.2 RS-485 connection between 3 DNs and a PC

Using an RS-485 serial line, the maximum length should not be more than 1000 m without amplifiers.

It is important to use braided and shielded cable, connecting the shield to pin 9 of the DB9 connector.

The data cable should be placed away from high-voltage lines. The connector shown corresponds to the cable connector. On both sides of the transmission line a 120 Ω end-of-line resistance should be placed.

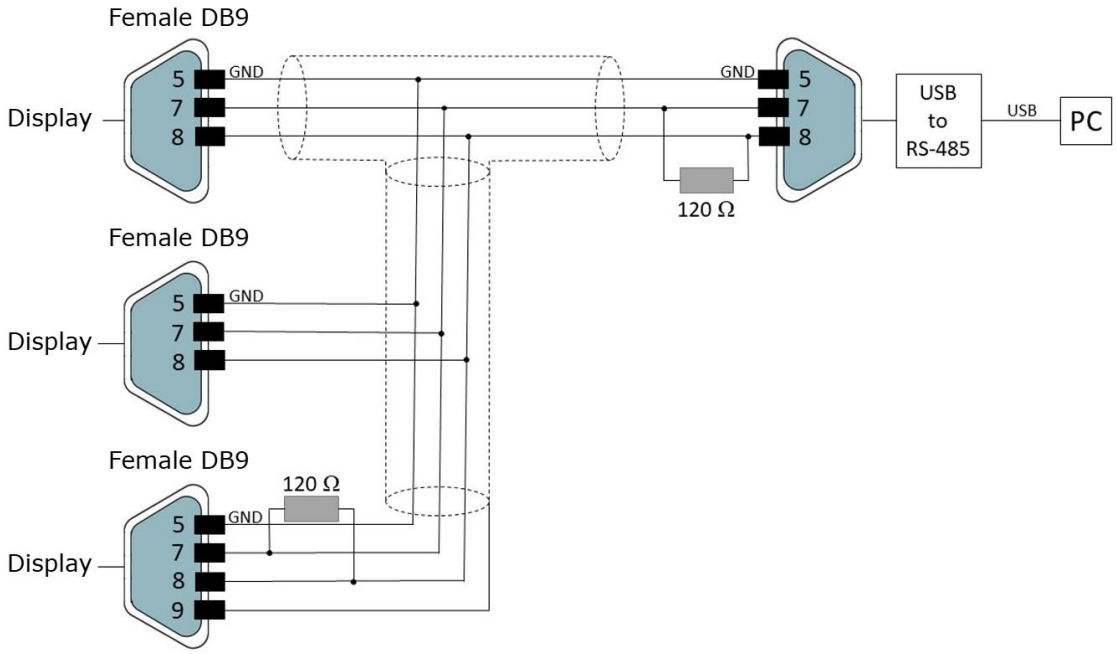


Fig. 7: Schematic of the RS-485 connection between 3 DN and a PC.

3.4 Connecting the Ethernet Line

The Ethernet line is connected via an RJ-45 connector located on the bottom of the device.

The connection between a display and a computer using an Ethernet link can be made in two ways: Direct connection or through a switch or hub and a 100Base-T4 cable, recommended category 5.

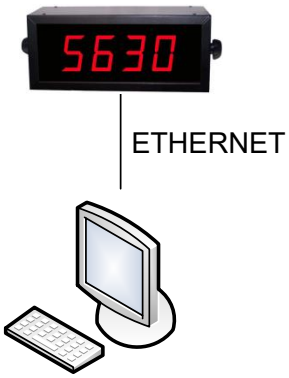


Fig. 8: Schematic of the Ethernet line connection between a DN display and a PC.

To connect several displays, a switch or hub hub with a port for each device must be used.

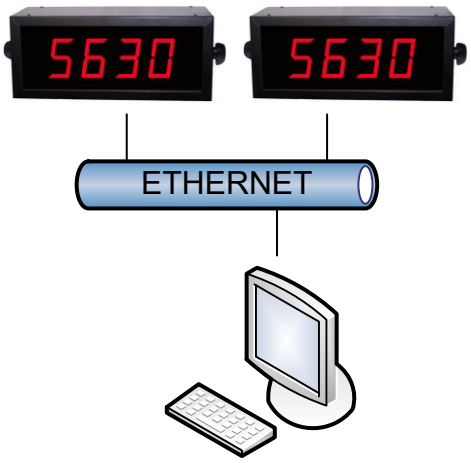


Fig. 9: Schematic of the Ethernet line connection between several DN displays and a PC using a switch or hub.

4 DISPLAY INITIALIZATION

4.1 Initial start-up

Before connecting the display to the network, we must make sure that all connections have been made correctly and that the display is firmly in place.

During the boot process, the different initialization stages will be indicated in the display, in order:

(1) Pr0: To avoid situations where a FW update has not completed successfully or an erroneous FW has been loaded that may cause the display to become inoperable. Each time the device is powered and before starting the main program, an access time is established to the Bootloader (update load manager) that will allow the FW update process to be retried by the emergency IP address **192.168.1.100** (not by the IP address that was previously defined on the device). This time is signaled by the message "PR0" and indicates the time when the update can be repeated. (See (9).

(2) Test sequence of the LED segments: During this period, all the segments of the digits are activated one by one. They are then deactivated in reverse order. This sequence is used to detect segments that stop working over the years.

(3) F.XX: Shows the display family. "XX" corresponds to a specific value of your display.

(4) Ux.X: Displays the version of the firmware loaded. "X.X" corresponds to the specific value of your display.

(5) Started Display: Displays the value sent from the PC/PLC or display dashes in case you have activated the "time out of data" and do not send information. If you don't have any devices connected, it will show zero.

To access the web server of the display, the program "Display Discoverer" (https://www.lartet.com/en/download-center/) must be downloaded.

To configure the display, it must be connected via an RJ45 cable to the same network as the PC in use. It is also possible to configure it via WIFI communication.

4.1.1 Setup with Display Discoverer

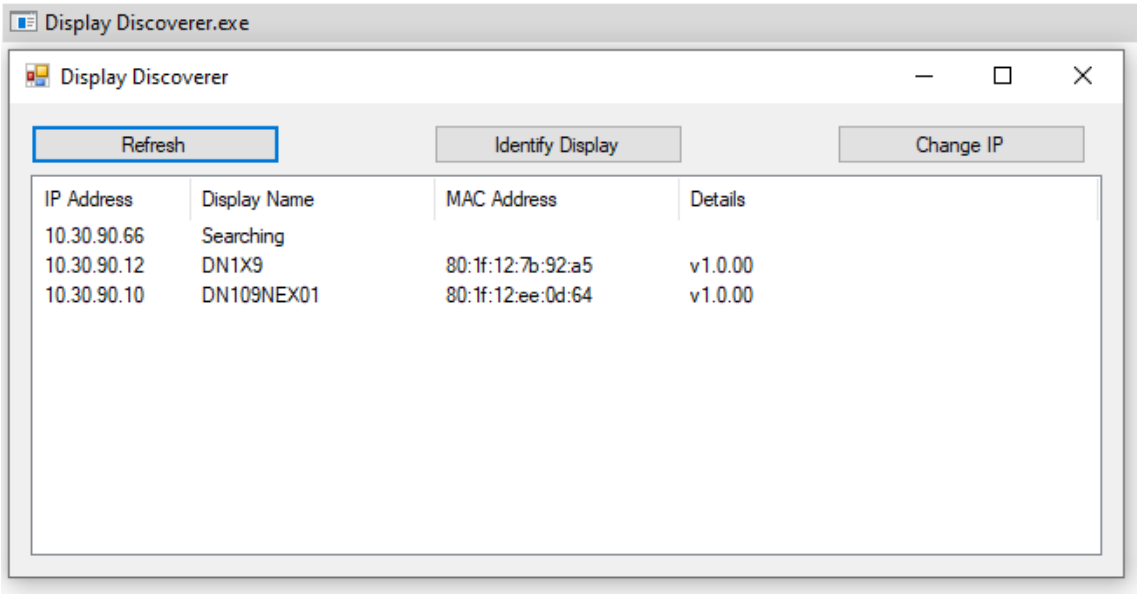


Fig. 10: Running the "Display Discoverer" software.

By default, the display comes with the IP address **10.30.90.11** out of the box. To change the IP you must select the IP and press the "Change IP" button. Once the new IP has been configured, click on "Refresh" to display the device with the new IP. The IP can also be changed later, during the configuration of the display.

If you want to set an automatic IP address, you must click on "Auto-Assign IP". This way the display will be set to DHCP mode. This mode can also be applied later, during the configuration of the display.

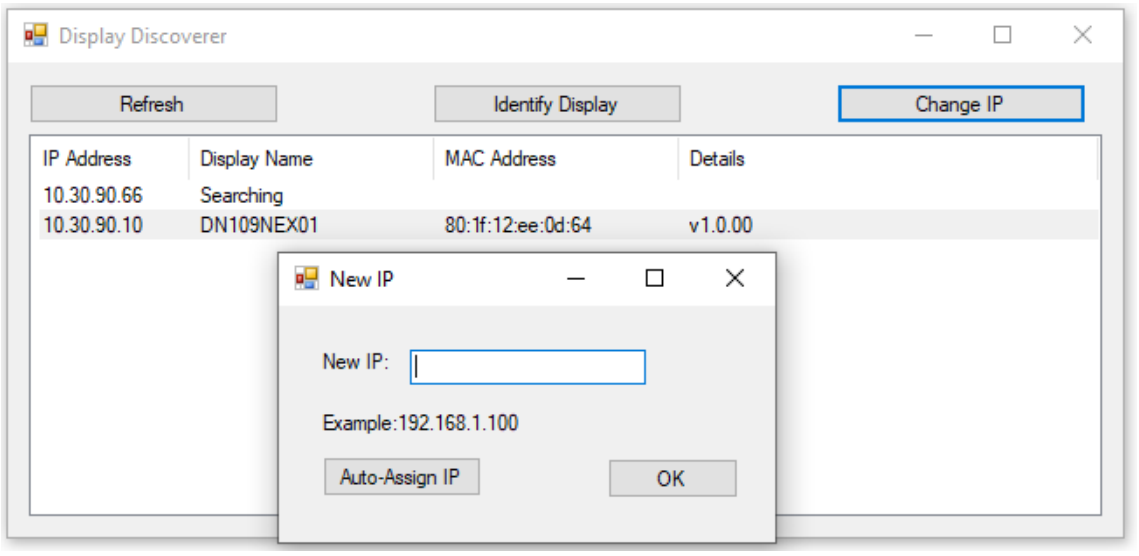


Fig. 11: IP change procedure.

Double-clicking on the already changed address will open the browser on the web server page. It can also be accessed by typing the IP directly into the browser.

IMPORTANT:

If several new devices are received for installation, it must be considered that they will all be configured with the same IP address. Therefore, prior to its configuration, the IP of some device must be changed individually to avoid duplication of addresses.

When working with multiple displays, it can be easy to get confused in the configuration of a particular display, of the set of installed devices. For this reason, the "Identify Display" function has been added. Selecting a display from the list of detected devices and clicking this button will cause it to flash 3 times, allowing you to easily identify which device is going to be configured.

4.2 Display Settings

The configuration of the display is carried out through the web server at the address that has been established according to the *Display Discoverer program* (See section 4.1.1), this server is internal to the display itself.

This requires connecting the display to a computer, either point-to-point or via the company's Ethernet network (and configuring it from any computer connected to the network).

If you want to access the public time servers, you need to have an Internet connection.

It is also possible to configure and use the display via WIFI connection. The WIFI connection has its own IP address.

Here's a breakdown of the different screens and elements that can be configured using the server, their use, and how they affect the display:

4.2.1 Overview of the set

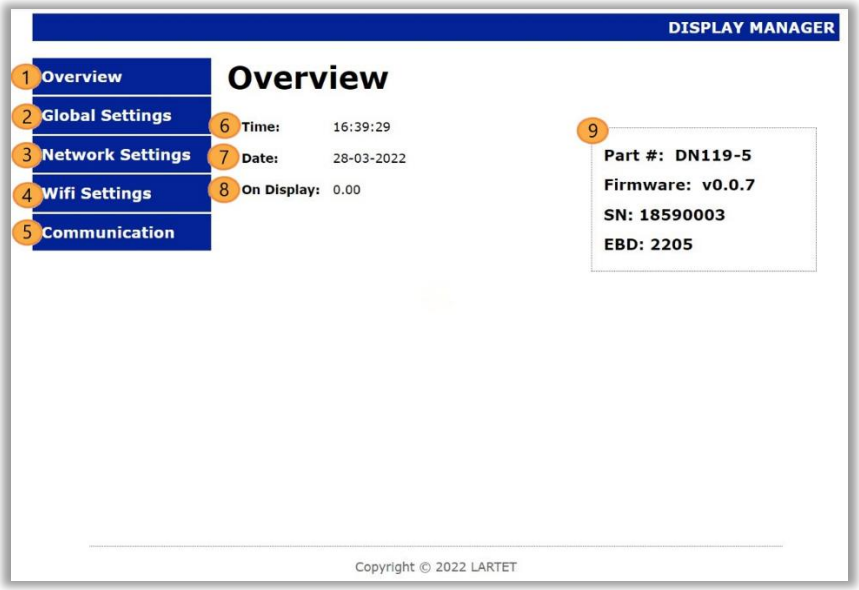


Fig. 12: Web server home screen. Overview of the display.

The initial screen of the server shows us a view of the basic information of the display and the buttons necessary to move through the different configuration options.

1. Link to the initial server screen. Displays real-time information from the display.
2. Link to the general settings screen. It allows you to select the communication interface, configure the data representation format, make time and brightness adjustments, etc.
3. Link to the network settings screen. Establish LAN communication. It allows the configuration of Ips, netmasks, *Gateway*, DNS, and DHCP.
4. Link to the wireless network settings screen.
5. Link to the communication settings screen (Ethernet and serial). Based on the communication set to **2**, it allows the adjustment of the necessary parameters.
6. Displays the time set in the display.
7. Displays the date set in the display.
8. Displays the value that is being indicated in the display:
    a. **OvH:** The value exceeds the maximum value that the display can display. The display displays "OvH".
    b. **OvL:** The value exceeds the minimum value that can be displayed by the display. The device shows "OvL".
    c. **TRIMMED:** The number of characters sent is higher than what can be displayed in the display. The display displays the possible characters. The full number is displayed on the web server, "TRIMMED" is marked in parentheses.
9. Displays information related to the manufacturing process. It has no relevance to the user. You can ask for the technical assistance service to resolve incidents with the device.

4.2.2 General Settings

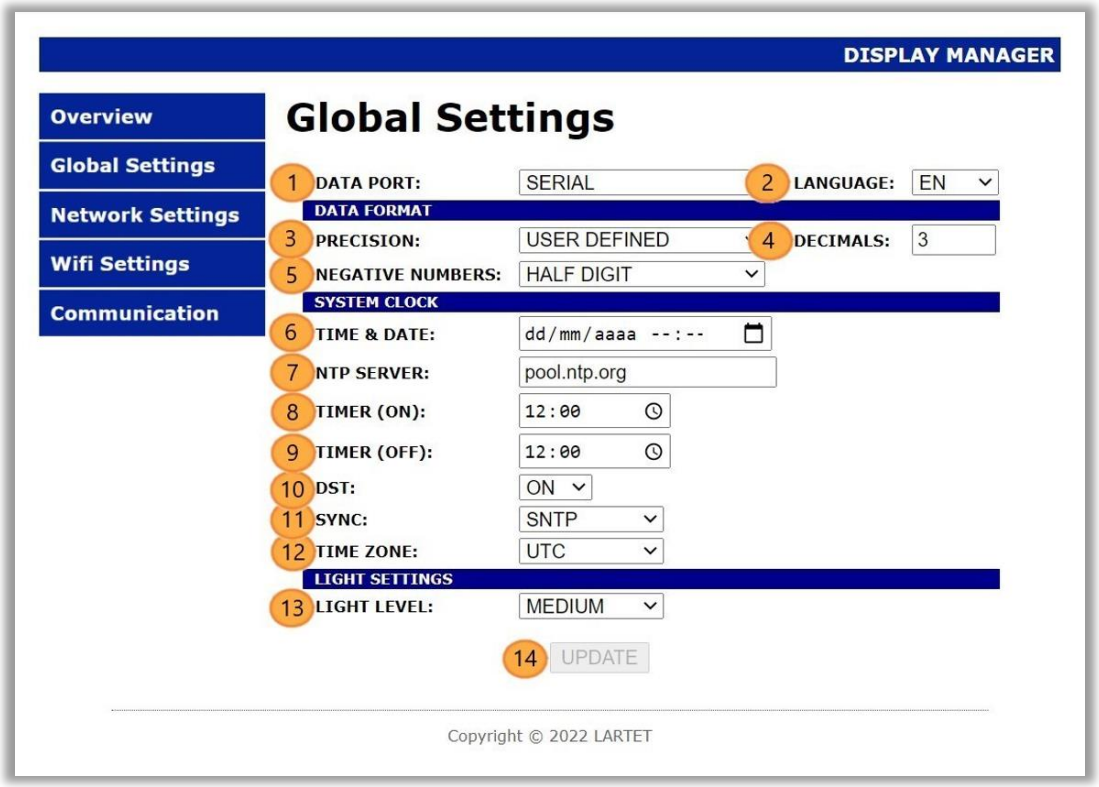


Fig. 13: Web server general settings screen.

Various parameters and options are modified on the general settings screen. Such modifications will also affect the web server's configurable parameters and subsequent configuration screens.

1. Configure which type of line will provide the information to the display.
   The device has different communication ports to receive the data to be displayed. The appropriate one must be selected according to the connection of the device.
2. Set the language in which the display will be configured. It is currently only available in English.
3. Configures the accuracy of the values displayed in the display. "**AUTO**" will automatically set the number of decimal places to be displayed, the displayed value will be adapted to display the full value by taking advantage of all the available digits of the display. If "**USER DEFINED**" is selected, you must specify the number of decimals to be displayed (parameter 4).
4. The number of decimal places that numeric values will have in the display is set.
   Below is a table of examples, in this case a 4-character display is considered.

VALUE TO VISUALIZE	PRECISION	DECIMALS	VISUALIZED VALUE
1.23	USER DEFINED	2	1.23
1.23	CAR	-	1.23
1.234	USER DEFINED	2	1.23
1.234	CAR	-	1.234
1.235	USER DEFINED	2	1.24
1.235	CAR	-	1.235
1.23	USER DEFINED	3	1.230
1.2345	USER DEFINED	4	1.235
1.2345	CAR	-	1.235

Table 1: Display examples for different values of "PRECISION" and "DECIMALS".

5. When you select "**FULL DIGIT",** the left character is used only to display the negative sign "-".
   If "**HALF DIGIT"** is selected, the left character can take the values "-" and "-1", thus increasing the number of digits of a negative number.

   Example: In a 5-digit display, the minimum value in "FULL DIGIT" is -9999, composed of the 5 characters "-","9","9","9","9","9","9". On the other hand, in "HALF DIGIT" it is -19999, composed of the 5 characters "-1","9","9","9","9","9".

Fig. 14: On the left, minimum value in "FULL DIGIT", -9999.On the right, minimum value in "HALF DIGIT", -19999.

6. Manually set the date and time of the display. Clicking on the default parameter will display the system date and time (PC).
7. SNTP address/domain that will be used to obtain the time accurately. By default, the public SNTP server is set *pool.ntp.org*. It can be configured with an internal SNTP server of the company or another publicly accessible server.
8. The time at which the display will turn on.
9. The time at which the display will turn off. To deactivate the automatic on/off, you must set both (on and off) at the same time.
10. Enable daylight saving time. The time change will be made automatically if "ON" is selected.
11. Allows you to select the desired method of synchronizing the clock.
    - **NONE:** It won't sync the clock.
    - **ETH_SNTP:** It will use the server set to **7**.
    - **WIFI_SNTP:** It will work identically to SNTP but using WIFI communication (It does not work in "ACCES POINT" mode).
12. Set the display's time zone.
13. Sets the degree of light intensity of the display.
14. Clicking "UPDATE" sends the new information to the display.

4.2.3 Wired Network Settings

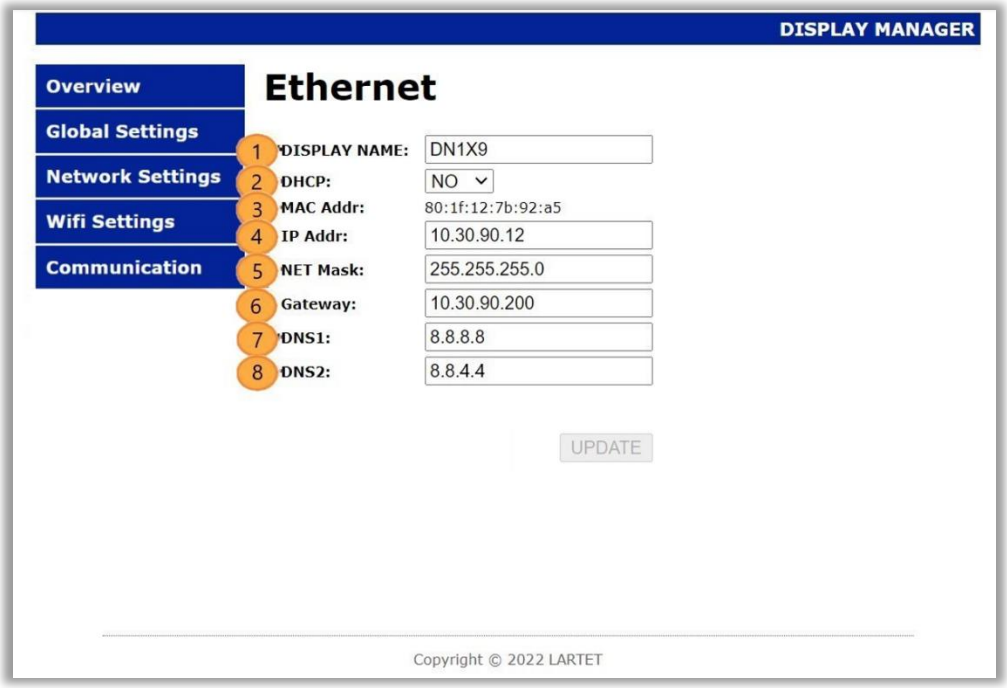


Fig. 15: Web server network settings screen.

The network settings screen configures the parameters related to the connectivity of the display.

1. Changes the name assigned to the display.
2. A drop-down that specifies whether the display will use the DHCP protocol. If the protocol is enabled, the Ethernet configuration parameters will be automatically assigned by the server.
3. Displays the MAC address of the display.
4. Modify the displays IP address.
5. Modify the netmask.
6. Modify the address of the "*Gateway*".
7. Modify the primary DNS address. If you do not select the DNS correctly, the SNTP server must be configured with your IP and not with your domain.
8. Modify the secondary DNS address.
9. Clicking "UPDATE" sends the new information to the display.

IMPORTANT:

A. The parameters only apply for wired Ethernet. For WIFI, a different configuration is available. Ips should not be repeated to avoid duplication conflicts, The display currently does not warn if this happens.

B. The IP address may become corrupted in the event of power loss during storage or if the memory is damaged. In these cases, the display will automatically reconfigure the emergency IP address **192.168.1.100**.

4.2.4 WIFI wireless network settings

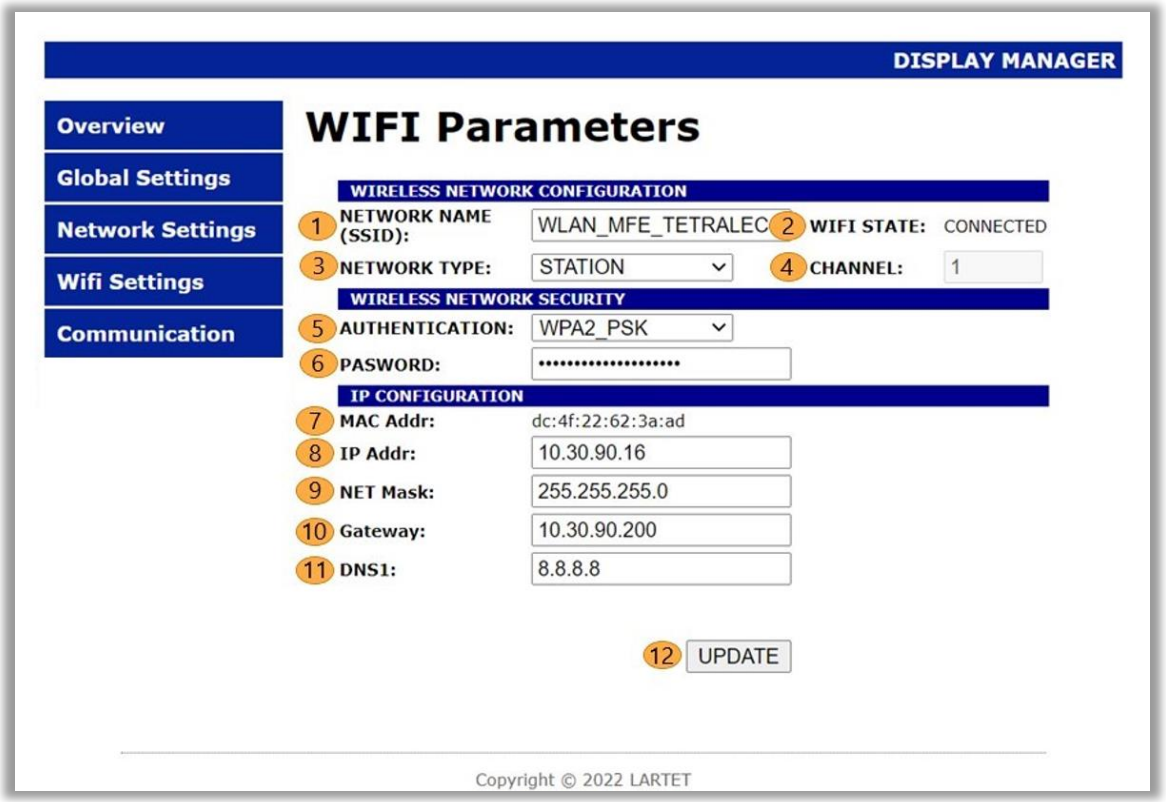


Fig. 16: Web server network settings screen.

The WIFI settings screen configures the parameters related to the wireless connectivity of the display.

1. STATION mode: Sets the name of the WIFI network to connect to.
   ACCESS POINT (AP) Mode: Sets the name of the WIFI network generated by the display.
2. Indicates the status of the connection.
   In **STATION mode**, if the correct password has not been entered, "ERROR" will be displayed, since the connection could not be made.
3. Configure the display to connect to an existing WIFI network ("**STATION")** or generate an AP ("**ACCES POINT").**
   By default, the network in AP mode is called **DIRECT_DN_DISPLAY** with 12345678 password.
4. In AP mode, it allows you to select the WIFI network channel. Allow any channel between 1 and 11.
5. Set the security type of the WIFI network.
6. Insert the password of the WIFI network.
   AP Mode: Set the password for the WIFI network.
   ATTENTION: In case of forgetting the password, it is necessary to access the display by wired connection to reconfigure it.
7. Displays the MAC address. There are two different directions, depending on whether the WIFI is set to **STATION** or **ACCES POINT mode**.

8. Allow you to configure the IP address
   ATTENTION: It is necessary to check that the IP address to be applied is not occupied.
9. Configure the netmask.
10. Configure the gateway.
11. Configure the DNS. If you do not select the DNS correctly, the SNTP server must be configured with your IP and not with your domain.

Modifying any parameter enables the button to send the new information to the display. The process of loading the new configuration takes approximately 15 seconds. In case of only changing the IP address, the process takes approximately 5 seconds, and an automatic redirection is performed.

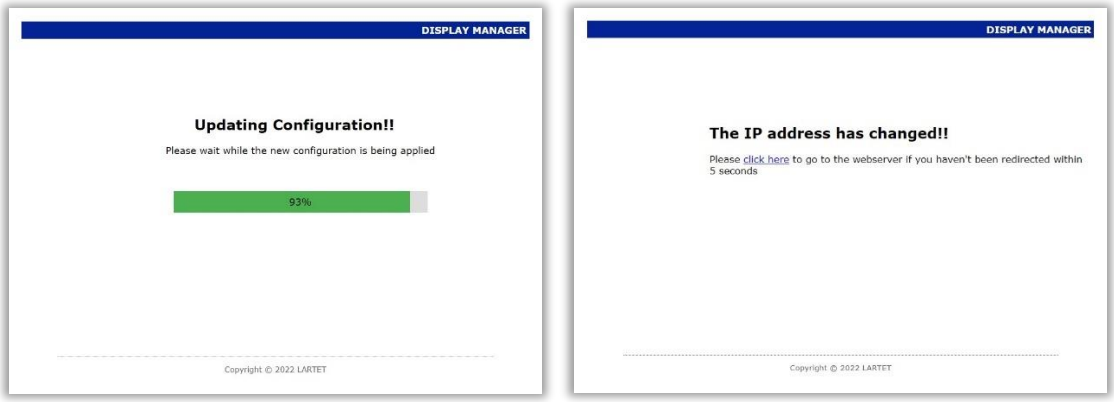


Fig. 17: On the left, progress bar during 15s charge. On the right, automatic redirection screen when changing IP.

Particularities of WIFI communication

WIFI communication is not identical in functionality to ETHERNET. The following are the factors to be considered:

- You can't modify your ETHERNET settings using WIFI.
- In case of forgetting the password, it must be reconfigured using wired ETHERNET, regardless of the mode of the WIFI configuration.
- When a direct WIFI connection is made to the display, if you are using a Windows operating system it may take too long to update the status of the connection as connected. One way to speed up the process is to immediately close and reopen the option to show "available Wi-Fi networks"
- For optimal performance, it is not recommended to send data to a display over WIFI while it is being set up.
- ETHERNET and WIFI IP addresses are independent

4.2.5 Communication Settings

See Communication Settings in Chapter 6.

5 WORK OPERATIONS

The numerical value notation used in this manual is as follows:

- When it is a **hexadecimal number**, the number will be written followed by "h".
- When it is a **decimal number**, the number will be written followed by "d".
- When it is a **binary number**, the number will be written followed by "b".
- When it comes to a number in **ASCII,** it will be described as such.
  For example, the X ASCII character can be seen as 58h, 88d, or 1011000b, depending on what needs to be described at the time. The ASCII number 15 can be described as 31h 35h, 49d 53d or 110001d 110101d depending on the context.

Definitions of words used in the description of this chapter:

XXX or **xxx**: 'X' sequences are used to indicate characters that can be variable, such as versions or dates.

5.1 Work operations and accepted data types

As mentioned above, this display can work with both numeric and text (ASCII) data types. If you are working in ASCII, specific control commands are available that allow you to put data to flash.

In any communication (**EXCEPT** MODBUS), the information is sent as an ASCII string. The display is responsible for converting this string to a numerical value in case it is made up exclusively of numerical information. Later it will also perform the necessary actions such as rounding, showing only the defined decimals, indicating if the value is out of range, etc.

In case the sent ASCII string contains alphanumeric characters, the display will internally switch to text mode. This means that it does not treat decimals or signal out-of-range values. Text mode allows you to represent non-numeric messages that can be displayed in 7-segments such as "E 345", "P-45" or "HOLA".

In MODBUS communication, the type of data to be sent depends on the record to be accessed. It will be an ASCII string in case the 0 register is used, but for the rest the frame must be formed according to the required numerical format (SWORD, UWORD, SDWORD OR UDWORD*). All information on the formation of MODBUS frames can be found in section 6.4.

*SWORD signed word (2 bytes signed) UWORD unsigned word (2 bytes unsigned) SDWORD signed double word (4 bytes signed) UDWORD unsigned double word (4 bytes unsigned)

Accepted ASCII character sequences

The display accepts alphanumeric ASCII characters that can be represented in a 7-segment. The valid characters accepted by the display are as follows:

Character	0	1	2	3	4	5	6	7	8	9	To	b
HEX	30h	31h	32h	33h	34h	35h	36h	37h	38h	39h	41h	42h
DEC	48d	49d	50d	51d	52d	53d	54d	55d	56d	57d	65d	66d

Character	C	c	d	And	F	H	h	i	J	L	n	or
HEX	43h	63h	44h	45h	46h	48h	68h	69h	4Ah	4Ch	4Eh (6Eh)	6Fh
DEC	67d	99d	100d	69d	70d	72d	104d	105d	74d	76d	110d	111d

Character	P	r	U	u		.	-
HEX	50h	72h	55h	76h	8pm	2Eh	2Dh
DEC	80d	114d	85d	117d	32d	46d	45d

Table 2: Accepted ASCII characters

6 ETHERNET BUS COMMUNICATION

6.1 Ethernet Communication Settings

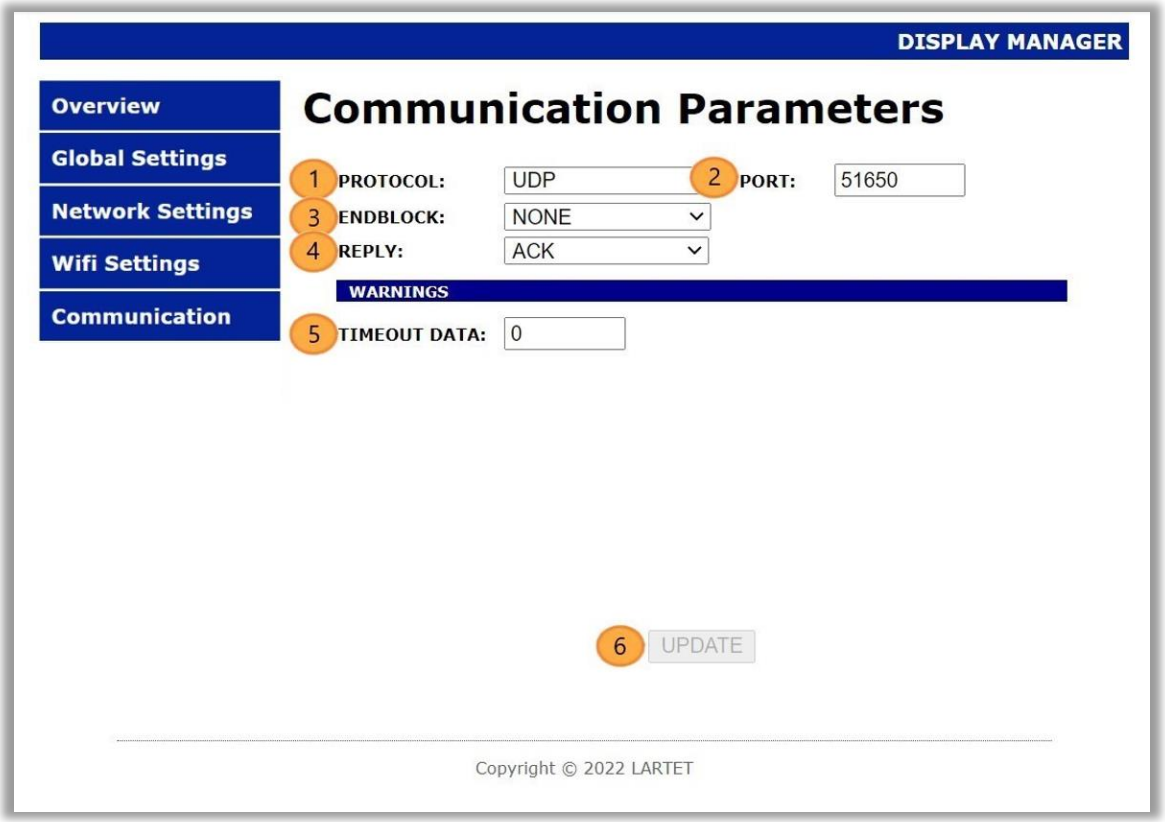


Fig. 18: Web server communication settings screen in case of Ethernet communication.

1. Drop-down menu to select the protocol to be used by the display.
   TCP, **UDP**, **and** MODBUS/TCP **protocols can be used**.
   The additional information on the protocols can be found in sections 6.2, 6.3 and 6.4, respectively.
2. Set the number of the communication port. It only affects TCP and UDP protocols. The port to choose must be in the ephemeral port range (49152 - 65535)

3. Allow you to select a frame ending. It is used as an enable, the display will only show the data to which the chosen frame end is added. Selecting NONE disables the enablement feature.

The available frame endings are as follows:

Endblock	
NONE	
02h	Value 02h
03h	Value 03h
04h	Value 04h
<CR> 0Dh	0Dh
<LF> 0Ah	0Ah
<CR LF> 0Dh 0Ah	0Dh 0Ah
<LF CR> 0Ah 0Dh	0Ah 0Dh
< * CR> 2Ah 0Dh	Omron 2Ah 0Dh Host-Link

Table 3: Content of the ASCII protocol block ends.

4. Allow you to select the response of the display. This response will be sent whenever the display receives a piece of data, regardless of whether it is viewed.

The available answers are as follows:

Reply	
NONE	No response from the display
ACK	Acknowledgment of receipt
06h	Value 06h
@ AH AL ED 0 * <CR>	40h Direc A Direc.B 45h 44h 30h 2Ah 0Dh
06h ENDBLOCK	06h End of block

Table 4: Content of ASCII protocol response messages.

5. Assigns the timeout (in **seconds**) of a new request before the device sets "-" to all characters. The value can be any integer multiple of 10 between 0 and 2550 (inclusive). In case of a value of 0 the display does not set any timeout, the last data will be displayed indefinitely.
6. Modifying any parameter enables the button to send the new information to the display.

6.2 MODBUS/TCP protocol

No end of block is required.

To use the MODBUS/TCP protocol, the communication port must be properly configured (See section 6.1).

6.2.1 MODBUS Functions

The MODBUS functions accepted by the displays are shown in the following table:

Type		Name	Code
Data Access	Access to internal bits and *physical* coils	Read Coils	01h
		Write Single Coil	05h
		Write Multiple Coils	0Fh
	16-bit access to internal registers	Read Holding Registers	03h
		Write Single Register	06h
		Write Multiple Registers	10h

Table 5: MODBUS functions accepted.

This section details how information is structured at the protocol level to debug communication problems with a MODBUS frame analyzer.

If you are already familiar with this protocol, you can go directly to section 6.4.2 where it is explained how the information must be encapsulated in the registers according to the type of data you want to represent, as well as the control characters that are available.

- **Read Coils:** Allows you to visualize the state of the internal bits or *designated physical* Coils.

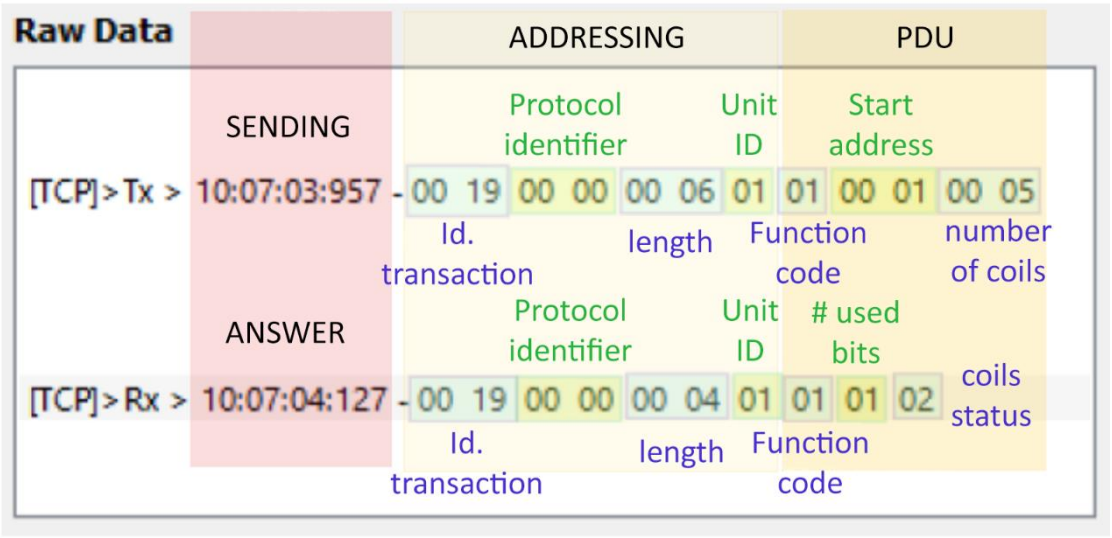
  The structure of this feature is as follows:

Request		
Function Code	1 Byte	01h
Start Address	2 Bytes	0001h to 0005h
Number of Coils	2 Bytes	0001h to 0005h
Reply		
Function Code	1 Byte	01h
Number of bytes used	1 Byte	N (N = # Inputs / 8)
Coils Status	n Bytes	n = N or N+1
Error		
Error Code	1 Byte	81h
Exception code	1 Byte	01 or 02 or 03 or 04

Table 6: Structure of the "Read Coils" function.

Example:

The display has only the 2nd Coil active, to know the status of all of them you can use this function. The following frames are sent and received:



IMPORTANT: When receiving the byte (02h) that indicates the status of the *Coils*, it should read as follows:

0				2			
X	X	X	0	0	0	1	0
			Coil #5	Coil #4	Coil #3	Coil #2	Coil #1

Table 7: Reading the Status of *Coils*. "X" indicates that it is not relevant, it is not used.

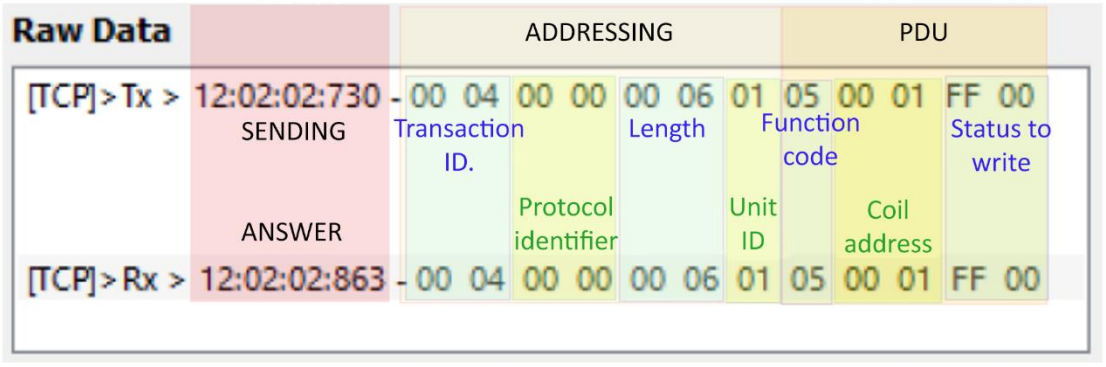
- **Write Single Coil:** Used to assign the ON/OFF state to a *Coil*.
  The structure of this feature is as follows:

Request		
Function Code	1 Byte	05h
Coil Steering	2 Bytes	0001h to 0005h
State to be written	2 Bytes	0000h(OFF) or FF00h(ON)
Reply		
Function Code	1 Byte	05h
Coil Steering	2 Bytes	0001h to 0005h
State to be written	2 Bytes	0000h(OFF) or FF00h(ON)
Error		
Error Code	1 Byte	85h
Exception code	1 Byte	01 or 02 or 03 or 04

Table 8: Structure of the "Write Single Coil" function.

Example:

You want to activate the 1st Coil. The following frames are sent and received:



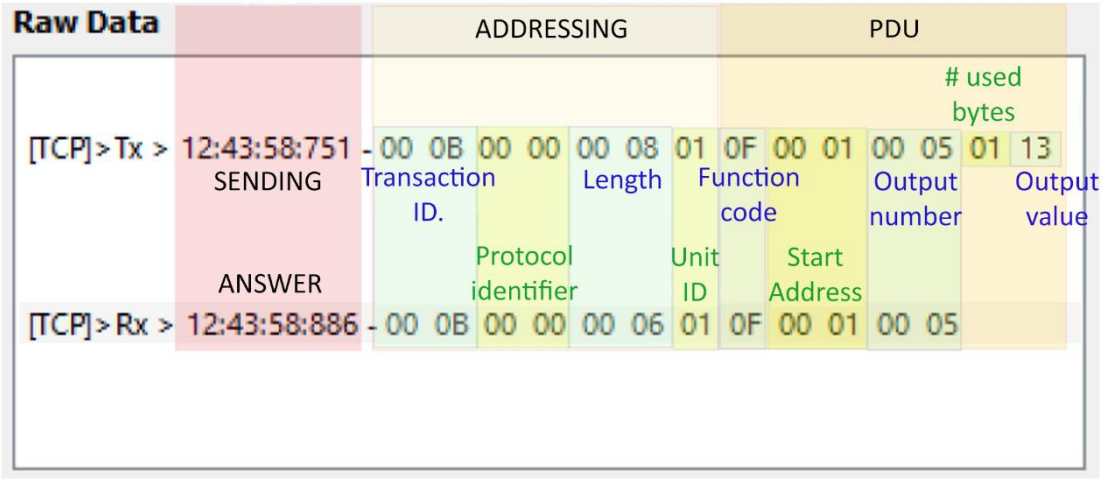
- **Write Multiple Coils:** Used to simultaneously assign the state of multiple consecutive addressing coils.
  The structure of this feature is as follows:

Request		
Function Code	1 Byte	0Fh
Start Address	2 Bytes	0001h to 0005h
Number of outputs	2 Bytes	0001h to 0005h
Number of bytes used	1 Byte	N (N = # Outputs / 8)
Value of the outputs	N x 1 Byte	XX... XX
Reply		
Function Code	1 Byte	0Fh
Start Address	2 Bytes	0001h to 0005h
Number of outputs	2 Bytes	0001h to 0005h
Error		
Error Code	1 Byte	8Fh
Exception code	1 Byte	01 or 02 or 03 or 04

Table 9: Structure of the "Write Multiple Coils" function.

Example:

You want to activate the 1st, 2nd and 5th Coils. The following frames are sent and received:



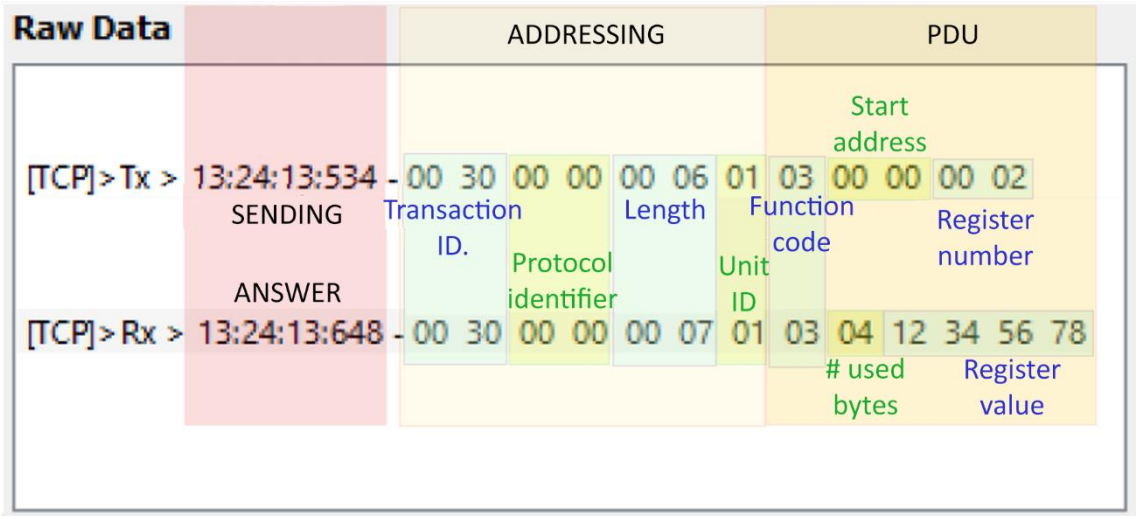
- **Read Holding Registers:** Allows you to view the contents of the desired registers. The structure of this feature is as follows:

Request		
Function Code	1 Byte	03h
Start Address	2 Bytes	0000h to 0011h
Number of registers	2 Bytes	0001h to 0012h
Reply		
Function Code	1 Byte	01h
Number of bytes used	1 Byte	2 x N (N = # of registers)
Value of registers	N x 2 Bytes	XX... XX
Error		
Error Code	1 Byte	83h
Exception code	1 Byte	01 or 02 or 03 or 04

Table 10: Structure of the "Read Holding Registers" function.

Example:

You want to read two registers with the contents "1234" and "5678", they are located at addresses 1 and 2. To do this, the following frames are sent and received:



- **Write Single Register:** Used to assign the value to only one register.
  The structure of this feature is as follows:

Request		
Function Code	1 Byte	06h
Registration Address	2 Bytes	0000h to 0011h
Registry value	2 Bytes	0000h to FFFFh
Reply		
Function Code	1 Byte	06h
Number of bytes used	2 Bytes	0001h to 000Ah*
Registry value	2 Bytes	0000h to FFFFh
Error		
Error Code	1 Byte	86h
Exception code	1 Byte	01 or 02 or 03 or 04

Table 11: Structure of the "Write Single Register" function.

*Depends on the type of data you want to write:

- 2 bytes (0002h): Word type value (with or without sign).

- 4 bytes (0004h): Word type value and reading brightness information.

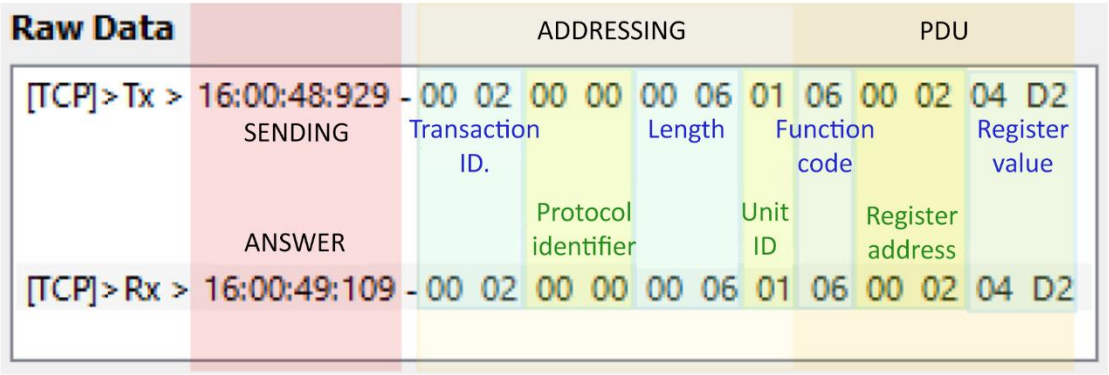
- 6 bytes (0006h): Double Word value with decimal information.

- 8 bytes (0008h): Double Word value and reading of bright information.

-10 bytes (000Ah): Reading the entire message sent to a 10-digit display.

Example:

You want to write the value "04D2h" to display "1234" in the address register "2". To do this, the following frames are sent and received:



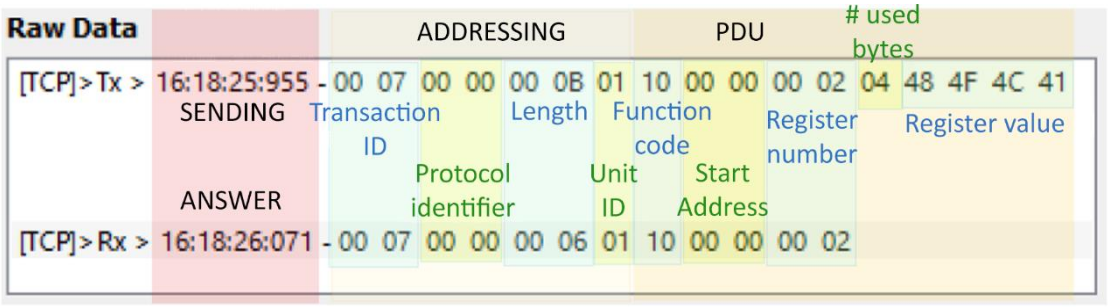
- **Write Multiple Registers:** Used to assign the value to multiple registers, simultaneously. The structure of this feature is as follows:

Request		
Function Code	1 Byte	10h
Start Address	2 Bytes	0000h to 0011h
Number of registers	2 Bytes	0001h to 0005h
Number of bytes used	1 Byte	2 x N (N = # of registers)
Value of registers	N x 2 Byte	XX... XX
Reply		
Function Code	1 Byte	10h
Start Address	2 Bytes	0000h to 0011h
Number of registers	2 Bytes	0001h to 0005h
Error		
Error Code	1 Byte	90h
Exception code	1 Byte	01 or 02 or 03 or 04

Table 12: Structure of the "Write Multiple Registers" function.

Example:

You want to write in 2 registers, with an initial address of "0". For the device to display the word "HOLA" it is sent to the logs "48h 4Fh 4Ch 41h". To do this, the following frames are sent and received:



6.2.2 Writing Registers

The different registers in the display allow you to interact in different ways, depending on the address to which you type, the following registers can be distinguished:

- **Register 02**: Send numerical values of the WORD type with sign.
- **Register 06**: Send Unsigned WORD Numeric Values
- **Register 10**: Send numerical values of the double *WORD type* with sign.
- **Register 14**: Send unsigned double WORD numeric values.
- **Register 00**: Send strings of ASCII characters representable in 7-segment digits.

All the possibilities are detailed below:

6.2.2.1 Register 02.

It is used to send numerical values of the WORD type with a sign. The range of values is made up of -32768 to 32767, inclusive.

You must send 2, 3 or 4 bytes of information, with the **start address** being 2.

The first two bytes indicate the value to be displayed (in complement to two).

The third byte can be used to manage flickering. Typing 08h starts the blinking, 09h ends it.

Finally, the luminosity is set by writing a value between 30h and 34h in the fourth byte (30h minimum – 34h maximum).

REGISTER 02		REGISTER 03	
Byte 1	Byte 2	Byte 3	Byte 4
<XXh>	<XXh>	<XXh>	<XXh>
Numerical value		Blinking	Luminosity*

Table 13: Summary of the values for registers 02 and 03.

Example:

To facilitate the sending of information, the Modbus 10h function is used.

To display the value "**-3270**" you must send the following content by Modbus.

REGISTER 02		REGISTER 03	
Byte 1	Byte 2	Byte 3	Byte 4
F3h	3Ah	00h	34h
Numerical value		Blinking	Luminosity*

Table 14: Example of the values to display "3270".

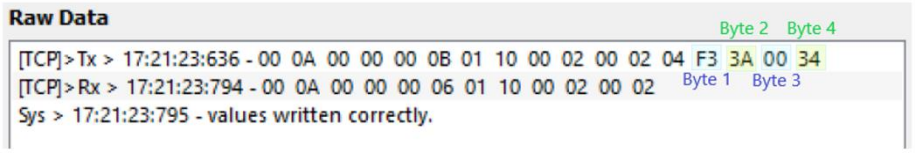


Fig. 19: Frame in MODBUS/TCP to display the value "-3270".

In these frames, the sending and response explained in the previous section can be observed.

6.2.2.2 Register 06.

It is important that in this case the **starting address** is 6.

This register is used identically to the previous one, but it is used for unsigned numerical values of the WORD type. Range changed from 0 to 65535. Luminosity is manipulated identically. Table 21 can be used interchangeably for this case.

For example, the same frame is sent as in the example in record 02, but the start address is changed to 06. In this case, 62266 is displayed, if the device is 5 digits or larger. Otherwise, the display shows "OvL"

6.2.2.3 Register 10.

It is used to send numerical values of the type of *double WORD* with a sign. The range of *values* is made up of -2,147,483,648 to 2,147,483,647, inclusive.

You must send between 6 and 8 bytes of information, with the **start address** being 10.

The first four bytes indicate the value to be displayed (in A2 complement). The next two modify the position of the decimal point. The last two (optional) are the flicker and brightness of the display.

The flicker and brightness values work identically to the previous cases.

REGISTER 10		REGISTER 11		REGISTER 12		REGISTER 13	
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
<XXh>	<XXh>	<XXh>	<XXh>	<XXh>	00h	<XXh>	<XXh>
Numerical value				Pos. Point	Not used	Blinking	Luminosity*

Table 15: Summary of values for registers 10, 11, 12, and 13.

Example:

To facilitate the sending of information, the Modbus 10h function is used.

To display the value "**-32.70**" you must send the following content by Modbus.

REGISTER 10		REGISTER 11		REGISTER 12		REGISTER 13	
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
FFh	FFh	F3h	3Ah	02h	00h	00h	34h
Numerical value				Pos. Point	Not used	Blinking	Brightness

Table 16: Example of the values to display "32.70".

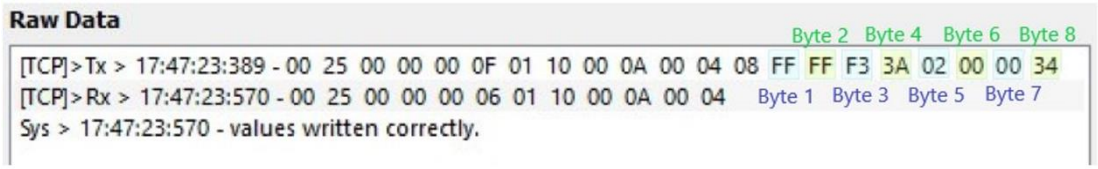


Fig. 20: Message in MODBUS/TCP to send the value "-32.70".

In these frames, the sending and response explained in the previous section can be observed.

6.2.2.4 Register 14.

It is important that in this case the **starting address** is 14.

This register is used identically to the previous one, but it is used for numerical values of the unsigned double *WORD* type. The range is changed from 0 to 4,294,967,295. The decimal point and luminosity are manipulated identically. Table 23 can be used interchangeably for this case.

For example, the same frame is sent as in the example in register 10, but the start address is changed to 14. In this case, 4,294,964,026 is displayed, if the device is 10 digits or larger. Otherwise, the display shows "OvL"

6.2.2.5 Register 00.

It is used to type text in the display (in ASCII characters). It should be noted that when writing on a 7-segment LED, only certain values can be set (See section 5.3.2). For this purpose, 10 registers are available, starting at address 0. Each record can contain two ASCII characters.

REGISTER 0		REGISTER 1		·····	REGISTER N	
Byte 1	Byte 2	Byte 3	Byte 4	·····	Byte 2*N+1	Byte 2*N+2
<XXh>	<XXh>	<XXh>	<XXh>	·····	<XXh>	<XXh>
ASCII 1	ASCII 2	ASCII 3	ASCII 4	·····	ASCII 2*N+1	ASCII 2*N+2

Table 17: Summary of the values for using register 00.

Example:

To facilitate the sending of information, the Modbus 10h function is used.

You want to display the text "E 523" to indicate an error in an industrial process. The following content must be sent by Modbus (See table of ASCII characters).

REGISTER 0		REGISTER 1		REGISTER 2	
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
45h	8pm	35h	32h	33h	00h
"E"	" "	"5"	"2"	"3"	

Table 18: Example of the "E 523" display values.

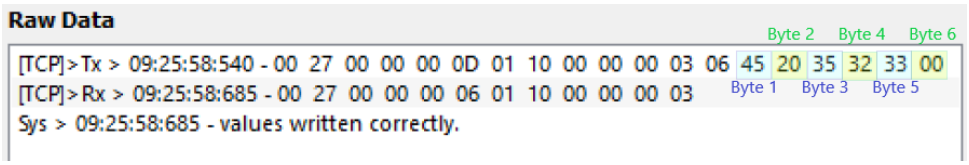


Fig. 21: Message in MODBUS/TCP to send the characters "E 523".

As you can see, as many registers are written as necessary. In this case, as the number of characters is 5 (odd), you must use 3 registers and set the last byte to 00h.

If this method is used, the light intensity of the display cannot be modified for reasons of compatibility with previous products.

6.2.3 Coils Writing

In addition to registers, the 0Fh "Write Multiple Coils*" function can be performed with MODBUS/TCP* to activate/deactivate the relays or set the flashing of the display.

They can also be modified with the 05h "*Write Single Coil*" function, but the 0Fh function is used in the examples for simplicity.

5 coils *are set* that can be activated or deactivated, starting from **START ADDRESS 1**. In order, the *coils* are used to establish the following parameters:

- ***Coil* 1.** Turns relay 0 on or off.
- ***Coil* 2.** Turn the relay on or off 1.
- ***Coil* 3.** Turns the relay on or off 2. (DEVICE OPTIONS)
- ***Coil* 4.** Turn the relay on or off 3. (DEVICE OPTIONS)
- ***Coil* 5.** Turn the display flicker on or off.

All *coils* are activated at "1" and deactivated at "0". Your display model may not have coils 3 and 4, in which case their value will not affect any functionality.

Example:

You want an alarm to be triggered for a certain value and the display to start flashing the displayed value. To do this, it is necessary to activate one of the relays and the flashing of the device. In this case, the COIL1 must be activated to computate the RELAY0 and the COIL5 to activate the flashing.

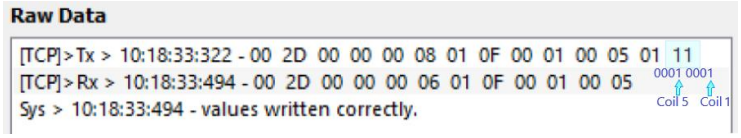


Fig. 22: Example of activation *coils* of the display.

6.2.4 Reading registers and *coils*

This new version of display allows the reading of both registers and *coils*.

The system works identically to the write, the desired registers or *coils* and their address are selected to proceed with the reading of the status.

It can be read with the MODBUS functions 01h "*Read Coils*" and 03h "*Read Holding Registers*".

The frames necessary to perform these functions are set out in section 6.4.1, "Read Coils" and "Read Holding Registers", respectively.

6.3 TCP/IP protocol

To use the TCP/IP protocol, the communication port must be properly configured (See section 6.1).

For the devices to display a raster, the raster must be finished with a raster end that is recognizable by the display.

The last character sent is displayed to the right of the display.

The communication port is set automatically but can be modified at any time.

The display will be able to receive decimal numbers and ASCII characters. Once the desired values have been sent, there are certain control parameters to modify, brightness and flicker. These parameters should be sent **AFTER** the values to be displayed.

Brightness	
Command	ASCII Code
Minimum	"Y0" or "y0" (7930h or 5930h)
Low	"Y1" or "y1" (7931h or 5931h)
Medium	"Y2" or "y2" (7932h or 5932h)
High	"Y3" or "y3" (7933h or 5933h)
Maximum	"Y4" or "y4" (7934h or 5934h)

Blinking	
Command	ASCII Code
Get Started	08h
Finish	09h

Examples:

The following table shows some examples by combining numerical values in ASCII format and control parameters.

Frame to send								Visualized Value
%QBx	%QBx+1	%QB x+2	%QB x+3	%QB x+4	%QB x+5	%QB x+6	%QB x+7	*
38h	39h	2Eh	35h	37h	32h			89.572(1)
2Dh	36h	37h	2Eh	31h	30h	08h		-67.10(2)
36h	2Eh	34h	36h	32h	33h			6.4623

(1) Value displayed without configuration

(2) Value Displayed in Blink

*For PLCs, %QBx corresponds to a certain position in the output address space, from that address the values must be consecutive.

6.4 UDP Protocol

The functionality of the parameters explained in the previous section (TCP/IP protocol) also applies to the UDP protocol.

7 WIFI COMMUNICATION

The bandwidth of WIFI communication is reduced compared to the wired connection. As a result, the setup experience will always be smoother via cable. This difference will not be noticeable once the device is configured and dedicated exclusively to data visualization.

For these same reasons, it is advisable not to transmit information with the display while it is being configured via WIFI.

7.1 WIFI Communication Settings

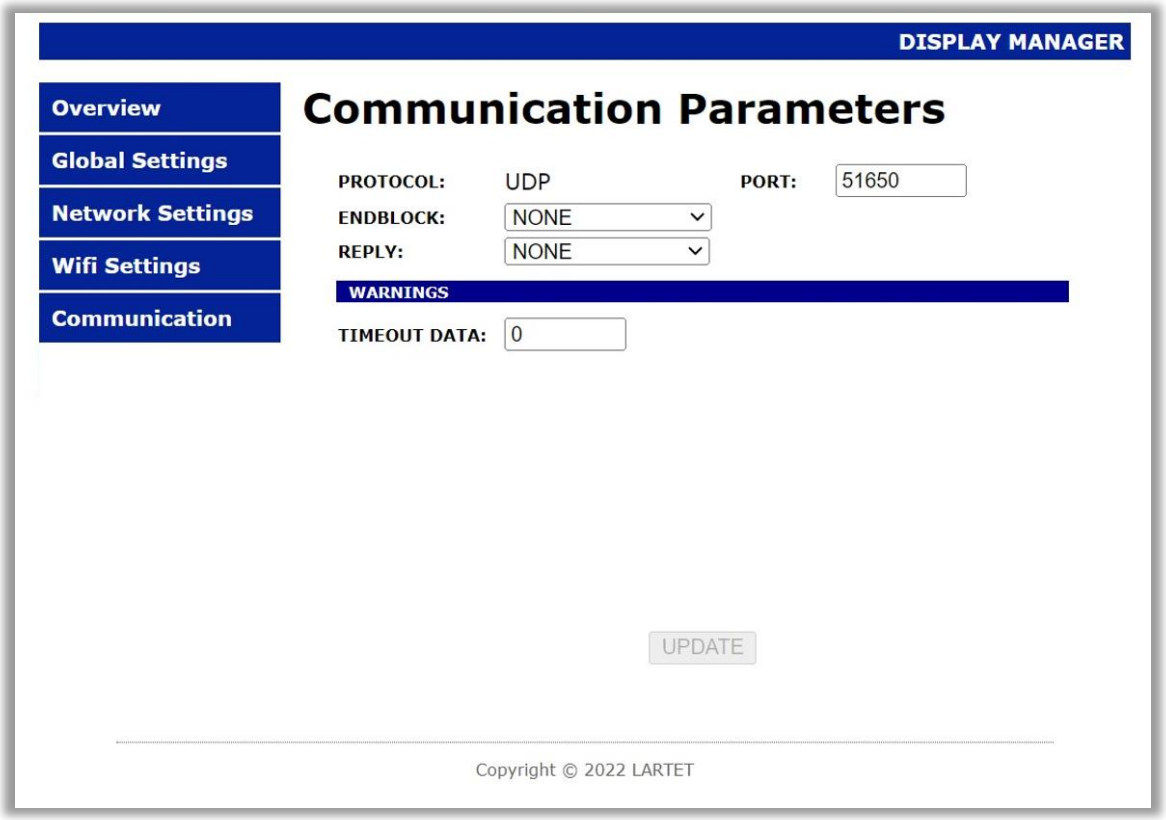


Fig. 23: WIFI communication settings screen.

The WIFI communication configuration screen is very similar to that of wired Ethernet. It differs in that there is no protocol option (there is only communication over UDP) and an additional response type is added, the "ECHO" response is limited to returning the same message that has been received.

The UDP protocol in WIFI works the same as it does over ETHERNET.

8 SERIAL BUS COMMUNICATION

This section includes the settings of the web server of the display and the operation of the protocols available in case of communication over the serial bus.

8.1 Serial Communication Settings

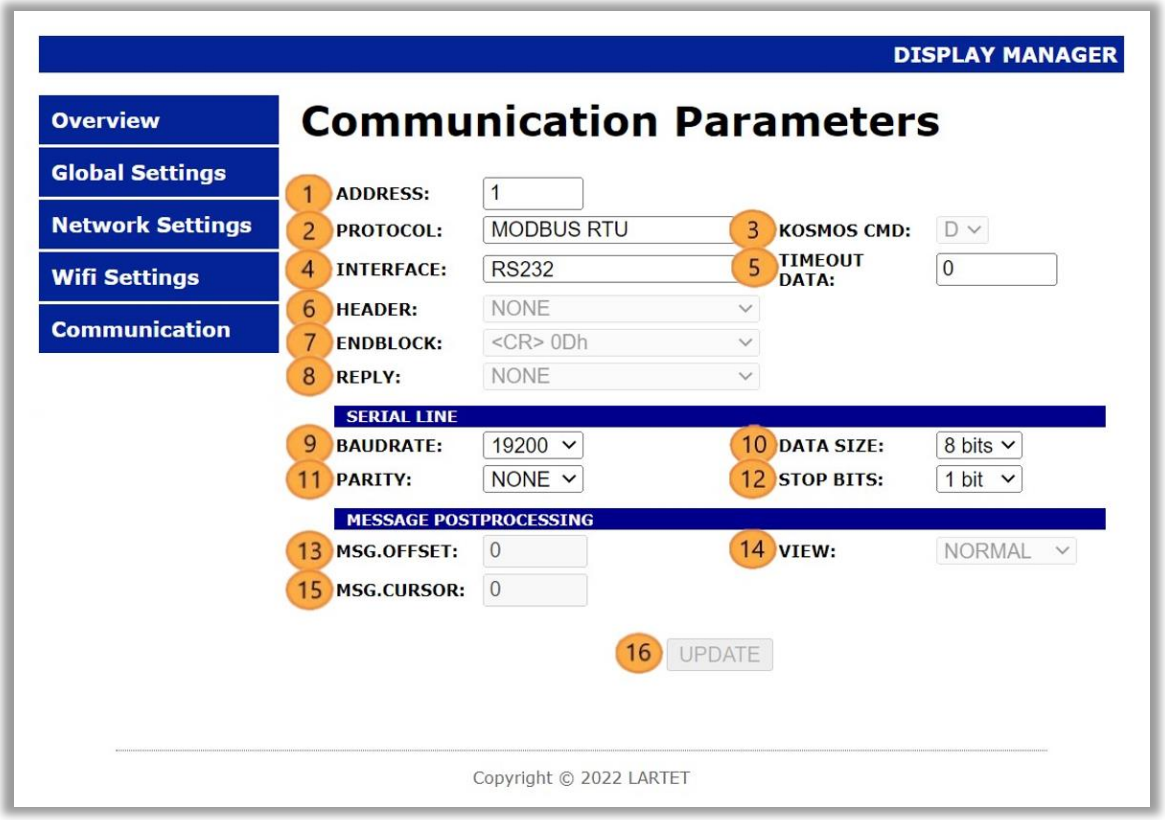


Fig. 24: Web server communication settings screen in case of serial communication.

On the communication settings screen, the parameters related to the display communication are configured:

1. Modifies the internal address that is assigned to the display.
2. Drop-down menu to select the protocol to be used by the display.
   The KOSMOS protocols are detailed in the manufacturer's manual.
   Information on the use of the **MODBUS RTU** and **ASCII** protocols is detailed in sections 7.3 and 7.4, respectively.

3. Drop-down menu to select the specific command in case a KOSMOS protocol has been selected.

Command	Function
D	Display value transmission
T	Transmission of the tare value
P	Peak value transmission
V	Transmission of the valley value
And	Transmission of the peak-peak value
Z	Transmission of total value

For more information on the function of these commands, please refer to the manufacturer's manual.

4. Select the connector's physical interface type.
   As standard, you can choose between **RS-232** and **RS-485** interfaces. Device that has been ordered with the **RS-422** option has the **RS-232** and **RS-422 interfaces**, with the connection modification explained in section 3.4.
5. Assigns the timeout (in **seconds**) of a new request before the display shows "-" on all digits. The value can be any integer multiple of 10 between 0 and 2550 (inclusive). In case of a value of 0 the display does not set any timeout, the last data will be displayed indefinitely.
6. Allow you to select a message header. ASCII protocol only.
   The following are the available headers:

Header	
NONE	
02h	Value 02h
02h AH AL	Value 02h + Display Direction. (Byte_High Byte_Low)
02h TO AH	Value 02h + Display Direction. (Byte_Low Byte_High)
@ AH AL E D	Host-Link de Omron
AH AL	Display directions. (Byte_High Byte_Low)
TO AH	Display directions. (Byte_Low Byte_High)

Table 19: Content of the ASCII protocol headers.

7. Allow you to select the end of the message.
   The block endings are listed below:

Endblock	
<CR> 0Dh	0Dh.
<LF> 0Ah	0Ah.
<CR LF> 0Dh 0Ah	0Dh 0Ah.
<LF CR> 0Ah 0Dh	0Ah 0Dh.
03h	Value 03h
< * CR> 2Ah 0Dh	Omron 2Ah 0Dh Host-Link.

Table 20: Content of the ASCII protocol block ends.

8. Allows you to select the response of the display.
The response messages are listed below:

Reply	
NONE	No response from the display
@ AH AL ED 0 * <CR>	40h Byte_High_Dir Byte_Low_Dir 45h 44h 30h 2Ah 0Dh
HEADER 06h ENDBLOCK	Header 06h End of block

Table 21: Content of ASCII protocol response messages.

9. Select the transfer rate in bits/s.
10. Select the number of bits per character.
7 **or** 8 **bits** can be selected.
If the MODBUS RTU protocol is used, it cannot be modified, the value is set to 8 bits.
11. Sets the parity bit for error handling.
The possible modes are **None**, **Even Parity**, and **Odd Parity**.
12. Select the number of stop bits, 1 or 2 bits for information synchronization.
13. Sets the position value of the first character you want to display. This option is intended to prevent the display of headers or labels that other computers may send along with the information. For example, a scale that could send "Weight (Kg): 203.5".

In case of applying the value 1, all the characters up to the first number will be ignored, very useful in case the label is variable (NET WEIGHT (Kg), AVERAGE WEIGHT (Kg), etc...).
Values greater than 1 allow us to focus on a part of the numerical data, useful in processes where the numerical value does not vary much. In this way, for example, thousands could be ignored in a process where only the units or the dozens.

Example: The information "WEIGHT 203.5" is sent. Depending on the value of this element (**13**), the following situations would occur:

A. **0.** As many characters as the display has been displayed. "WEIGHT 203.5" is displayed.
B. **1.** Everything is ignored up to the first numerical character. "203.5" is displayed.
C. **7.** The first 7 characters are omitted. "3.5" shown

14. Drop-down that allows you to reverse the order of the value to be displayed.
Example: The value "123456" is sent to the display, depending on the chosen parameter two situations can be displayed:

A. **NORMAL.** The value "123456" is displayed.
B. **INVERTED.** The value "654321" is displayed.

15. This parameter complements parameter **13**, but for the final part of the message. That is, you choose how many values should be displayed by counting from MSG. OFFSET.
The value of this parameter has different effects depending on the previous element (**14**).

A. **EYESIGHT = NORMAL.** Displays only the start of the message up to the designated position.
Example 1: With MSG. CURSOR = 3. If "123456" is sent, "123" is displayed.

Example 2: With MSG. CURSOR = 2. If "123456" is sent, "12" is displayed.

B. **VIEW = INVERTED.** Skip the start of the message (prior to the reversal) to the designated position.
Example 1: With MSG. CURSOR = 3. If "123456" is sent, "654" is displayed.
Example 2: With MSG. CURSOR = 2. If "123456" is sent, "6543" is displayed.

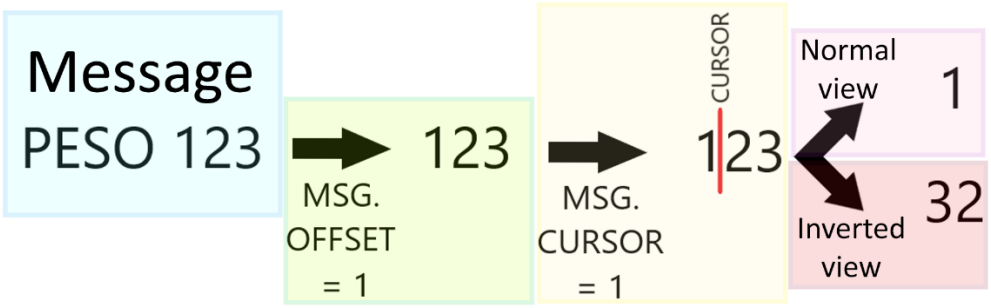


Fig. 25: Schema of use of MSG parameters. OFFSET i MSG. CURSOR.

16. Modifying any parameter enables the button to send the new information to the display.

8.2 KOSMOS (ASCII) and KOSMOS (ISO 1745) protocols

Information specific to the KOSMOS protocols should be found in the manufacturer's manual.

8.3 MODBUS RTU protocol

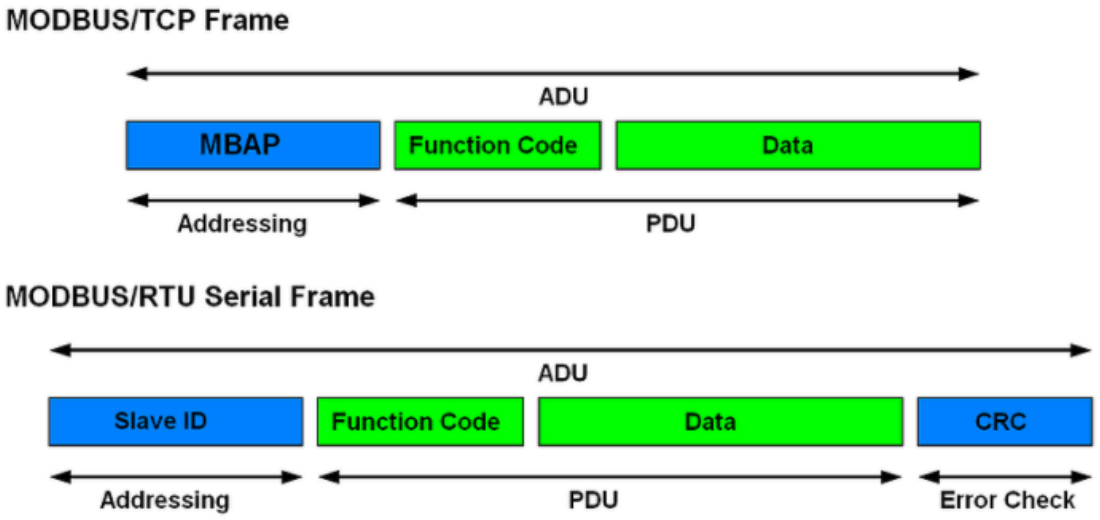


Fig. 26: Modbus/TCP and Modbus/RTU differences.

As can be seen, in the frames the addressing in RTU is slightly different and a CRC (Cyclic Redundancy Check) is added. The PDU remains unchanged between the two protocols

Regarding the use of Modbus functions and display registers, MODBUS RTU is identical to the MODBUS/TCP Ethernet protocol.

Refer to section 6.4 for any necessary information.

Example: To send "HOLA" (48h 4Fh 4Ch 41h) the following frames are established:

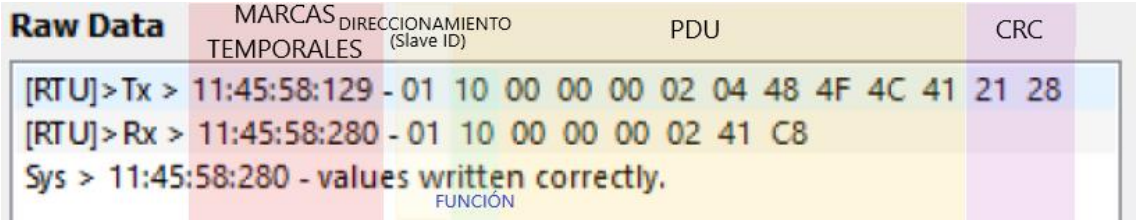


Fig. 27: Frames sent and received for display "HOLA" by MODBUS RTU

8.4 ASCII Protocol

This protocol allows easy communication with any device that has a serial line and can configure the protocol, such as a computer, a PLC working in RS-232 mode, etc. ernatively, several RS-485 network displays can be connected from the same device.

With this protocol the display works in slave mode, when it receives a piece of data it checks if the header and the end of the block correspond to those it has configured. If so, view the content of the data... The frame is configurable to adapt to a multitude of protocols that use the ASCII format.

To understand how the protocol can be configured, the terms used are described below:

Transmission block: It is made up of all the bytes necessary to be able to display value. For each successfully received stream block, the display will be updated with a new value. Each block consists of three parts: The header, the data, and the end of the block (Endblock).

HEADER: Used to identify the beginning of the block. You can choose between 6 formats or without *header*.

Data block: Contains the information to be displayed. It is possible to select the part of the block that you want to display.

ENDBLOCK: Used to identify the complete arrival of the block. You can choose between 6 types of *endblocks*.

In addition, there are control commands that allow you to start and end the flashing of one or more characters.

08h **Start of blinking characters**

09h **End of characters in blink**

Control commands must be placed at the end of the frame.

8.4.1 ASCII Protocol Examples

Example 1: Send a message from a computer to the display

The configuration for this example is as follows:

- **Display Address:** 08
- **Header:** 02h AH AL
- **Endblock:** CR
- **MSG. OFFSET:** 0
- **VIEW:** NORMAL
- **Data sent:** 358964

Transmission Block Sent

Transmission Block Sent in ASCII		0	8	3	5	8	9	6	4	CR
Transmission block sent in hexadecimal	02h	30h	38h	33h	35h	38h	39h	36h	34h	0Dh
	HEADER			Data sent						ENDBLOCK

Value displayed on a 4-digit device						Or	u	H
Value displayed on an 8-digit device			3	5	8	9	6	4

On the 4-digit computer, "OuH" is displayed because the value is too large to be represented.

Example 2: Send a message from a scale to the display.

The configuration for this example is as follows:

- **Display Direction:** 14
- **Header:** 02h AL AH
- **Endblock:** CR LF
- **MSG. OFFSET:** 1 (To display only the numeric value)
- **VIEW:** NORMAL
- **MSG. CURSOR:** 4
- **Data sent:** WEIGHT 15.8kg

Transmission Block Sent

Transmission Block Sent in ASCII		4	1	P	And	S	Or		1	5	.	8	k	g	CR	LF
Transmission block sent in hexadecimal	02h	34h	31h	50h	45h	53h	4Fh	8pm	31h	35h	2Eh	38h	6Bh	67h	0Dh	0Ah
	HEADER			Data sent											ENDBLOCK	

Value displayed on a 4-digit device				
	1	5.	8	

Having chosen **MSG. OFFSET = 1** the display has ignored all the characters prior to the first numeric value, without the need to count them.

In this case it is important to select **MSG. CURSOR = 4** ("15.8" is made up of 4 characters "." included), since after the first number the display will try to show everything up to the *endblock*.

If **NOT** selected, the following visualizations will be obtained. Due to the inability to represent some characters (*k* and *g*), "-" is displayed.

Value displayed on a 4-digit device				1	5.	8	-
Value displayed on an 8-digit device			1	5.	8	-	-

9 PROFINET COMMUNICATION

The display incorporates a Profinet interface for easy integration into industrial networks. This interface is equipped with two RJ45 connectors, allowing direct connectivity to Profinet systems, via a wired network.

The GSD configuration file defines the characteristics of memory space and presents the interfaces necessary for it to be recognized in the Profinet master application or PLC. The interface is displayed under the name Profinet payload and consists of 20 characters (uint8_t).

The communication protocol employs a configurable 20-byte buffer, which acts as an available memory space for receiving data. The contents of this buffer can be interpreted in both numeric and ASCII formats, offering flexibility to adapt to various application requirements.

In Profinet mode, the device operates as a slave. Since Profinet is a connection-based protocol, to establish communication, it is necessary to pre-configure the device as a Profinet slave. Subsequently, the connection must be initiated and managed by the teacher.

9.1 PROFINET Communication Settings

From the webserver, on the Global Settings page, it is necessary to configure the **DATA PORT** as Profinet. In this way, it is possible to activate the functionality and establish the connection with the device.

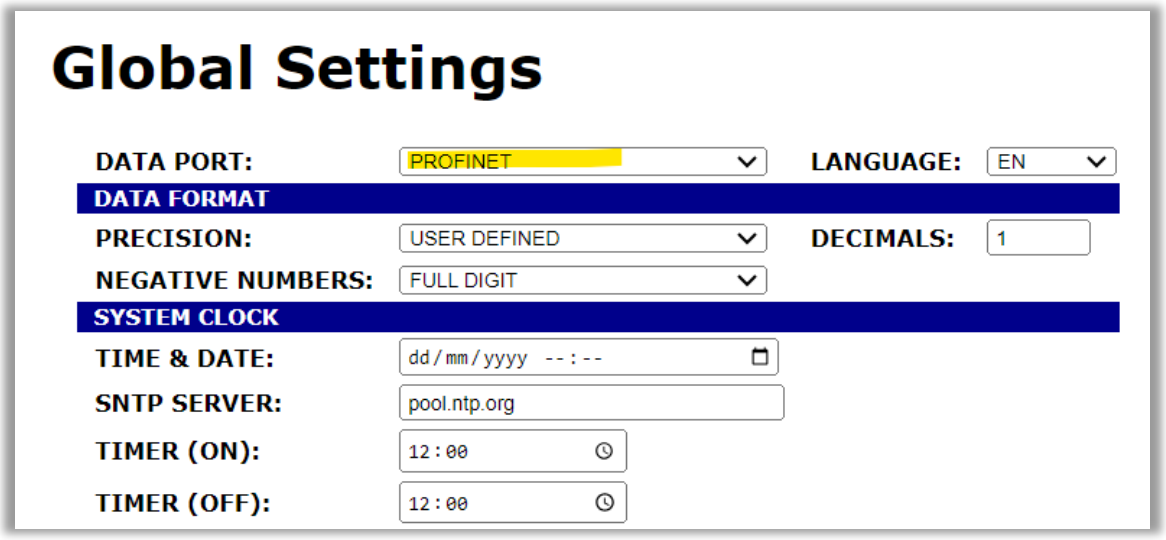


Fig. 28: Global Settings screen; Profinet configuration as a data entry port.

Once the Profinet connection is established, the device displays the contents of the 20-byte buffer in real time. This ensures a dynamic and accurate display of the transmitted data, complying with Profinet's communication standards.

To establish a valid IP address and Profiname to be able to establish communication with the PLC, it is first necessary to identify the Profinet device connected to the network with the Proneta application or similar and make the change of the IP address and the Profiname.

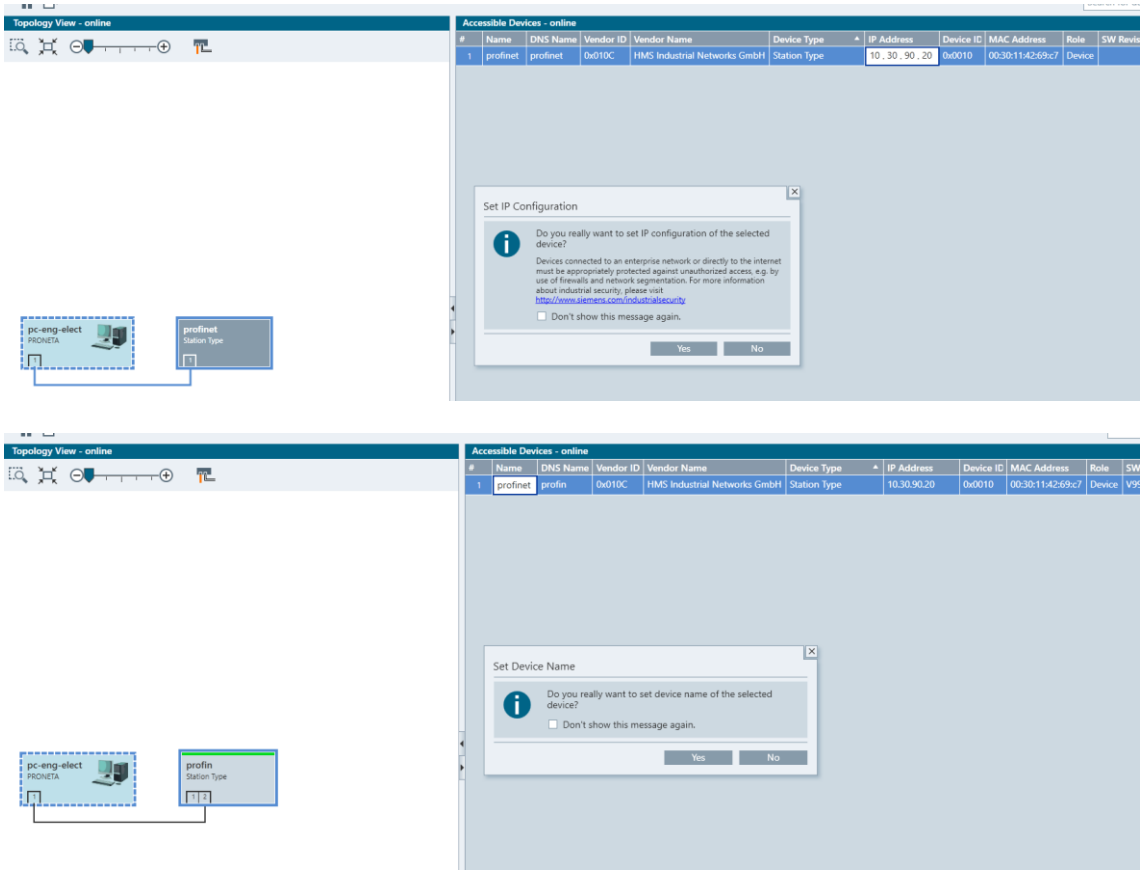


Fig. 29: Changing the IP address and Profiname of the Profinet module in Proneta

9.2 Types and formats of data in Profinet communication

The device supports five different formats of buffering data interpretation, including signed or unsigned numeric values, and Float or Word types. Additionally, data can be represented in ASCII format for added versatility.

The data format settings can be adjusted on the Communication Parameters screen on the webserver, after you have selected the Profinet option in **DATA PORT** of Global Settings.

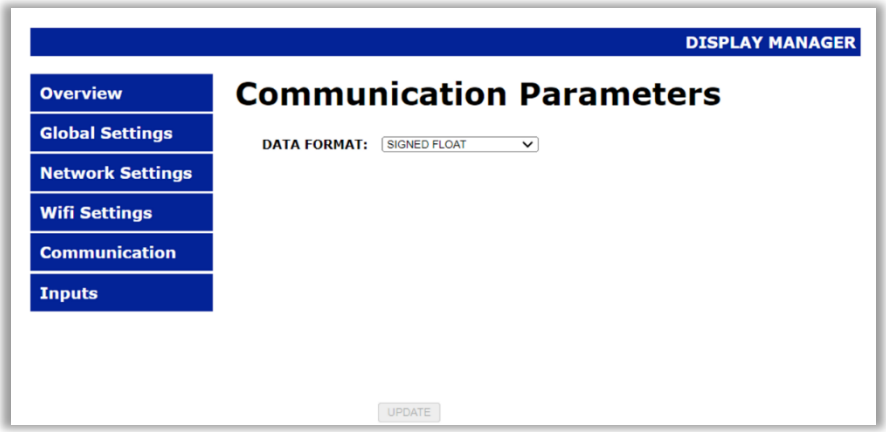


Fig. 30: Data format settings screen in Profinet mode

It is necessary to click on the update button for the changes made to be applied.

9.2.1 Float format

The data block reaches a maximum of 6 bytes when working in this format. The alignment of the bytes matches the addresses of the Profinet buffer, with **B0** corresponding to **Profinet payload(1), B1** to **Profinet payload(2),** etc. The byte allocation is as follows:

- **B0**: Highest weight byte of numeric value (MSB).
- **B1** and **B2**: Bytes of numeric value.
- **B3**: Lowest Weight Byte of Numerical Value (LSB).
- **B4**: Highest weight byte of decimal point position (MSB).
- **B5**: Lowest weight byte of decimal point position (LSB).

B0	B1	B2	B3	B4	B5
MSB num[3]	num[2]	num[1]	LSB num[0]	MSB dp[1]	LSB dp[0]

Table 22: Data format for the float type.

Likewise, the decimal point is coded according to the following table:

B4 to B5	Decimal Point Position
01h	00000000.0
02h	0000000.00
04h	000000.000
08h	00000.0000
10h	0000.00000
11h	000.000000
12h	00.0000000
2pm	0.00000000

Table 23: Decimal point position coding.

9.2.2 Word Format

When using the Word type, only the B0 and B1 bytes are used:

- **B0**: Highest weight byte of numeric value (MSB).
- **B1**: Lowest weight byte of numerical value (LSB).

This means that the 16-bit number (2 bytes) is encoded in big-endian order, i.e., the most significant byte is transmitted first, followed by the least significant.

For example, to convey the value 300 (decimal) in Word format, which corresponds to the 0x012C value in hexadecimal, we would send the value 0x01 in B0 and the value 0x2C in B1. The maximum value to send is 65535 (0xFFFF in hexadecimal)

9.2.3 ASCII format

In this mode, numeric values are sent in ASCII code. This format has the added advantage of allowing control commands, such as turning on flickering, using the 0x08 character, or turning it off, after the 0x09 character has been sent.

08h Start of flashing function (must be sent at the beginning of the frame)

09h End of function flashing (must be sent at the beginning of the frame)

The number of bytes sent in ASCII format will depend on the number of digits in the display plus the control commands used, with a maximum limit of 20 bytes, which corresponds to the size of the internal buffer allocated. Inserting a dot character [.] or the 0x2E code will turn on the dot segment for the previous digit.

The representation of bytes in the display follows a specific sequence: The first byte of the Profinet buffer corresponds to the rightmost digit of the display. The sequence of bytes will be represented in the digits from right to left in the following order:

Profinet Frame					Visualized value
Payload(1)	**(2)**	**(3)**	**...**	**(n+1)**	Dn, ..., D2, D1, D0
D0	D1	D2	...	Dn	

9.2.4 Encoding characters in ASCII format

The numerical value of each digit is encoded in ASCII and sent in the sequence mentioned above. Valid characters include some alphanumeric characters that can be represented in digits of 7 segments. The valid characters accepted by the display are as follows:

Carácter	0	1	2	3	4	5	6	7	8	9	A	b
HEX	30h	31h	32h	33h	34h	35h	36h	37h	38h	39h	41h	42h
DEC	48d	49d	50d	51d	52d	53d	54d	55d	56d	57d	65d	66d

Carácter	C	c	d	E	F	H	h	i	J	L	n	o
HEX	43h	63h	64h	45h	46h	48h	68h	69h	4Ah	4Ch	6Eh	6Fh
DEC	67d	99d	100d	69d	70d	72d	104d	105d	74d	76d	110d	111d

Carácter	P	r	U	u		,	.	-	_	‘	¯	
HEX	50h	72h	55h	75h	20h	2Ch	2Eh	2Dh	16h	27h	28h	
DEC	80d	114d	85d	117d	32d	44d	46d	45d	22d	39d	40d	

Table 24: A subset of characters supported in ASCII mode.

Sending a character that cannot be represented by the table above will be presented with the character "-" (2Dh).

9.2.5 Practical Coding Examples

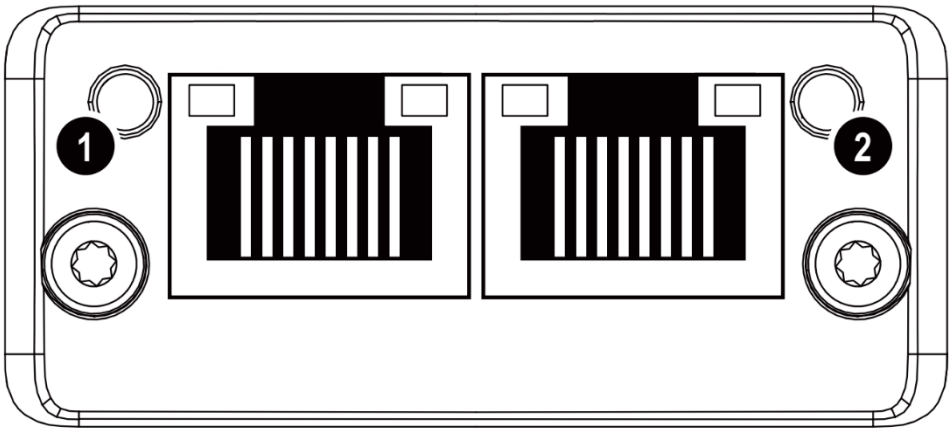
The following table shows seven practical examples of coding in each of the cases.

Type	Frame to send						Visualized value
Payload	(1)	(2)	(3)	(4)	(5)	(6)	
FLOAT with sign	00h	00h	04h	D2h	00h	02h	**12,34**
	FFh	FFh	E9h	D2h	00h	01h	**–567,8**
Unsigned FLOAT	00h	00h	B2h	6Eh	00h	00h	**45678**
	00h	01h	86h	9Fh	00h	04h	**99,999**
WORD with sign	FDh	A8h					**–600**
	05h	F5h					**1525**
Unsigned WORD	F4h	3Dh					**62525**

Table 25: Examples of visualization of different formats

9.3 LED Indicators

The Profinet module has two LEDs integrated into its front to indicate the module's Status, Network Status and Connection/Activity.



NS LED Status (1)		Description
Off	Offline	- No power- No connection to IO controller
Green	Online (RUN)	Connection with IO driver established, IO driver in running state.
Green, 1 blink	Online (STOP)	Connection to IO controller established, IO controller in STOP state or erroneous IO data, IRT synchronization not completed.
Green, 3 blinks	Identifying	Flashes 3 times (1 Hz) continuously to identify the slave (DCP_Identify).
Green, constant flashing	-	Connection with IO controller established, IO controller in STOP state.
Red	Fatal event	Significant internal error (this indication is combined with a red module status LED).
Red, 1 blink	Station Name Error	Station name not established.
Red, 2 blinks	IP address error	IP address not configured.
Red, 3 blinks	Configuration error	The expected ID differs from the actual ID.

MS LED Status (2)		Description
Off	Not initialized	There is no power or the module is in SETUP or NW_INIT state.
Green	Normal Operation	The module has changed from the NW_INIT state.
Green, 1 blink	Diagnostic Event	Diagnostic event(s) present.
Green, constant flashing	-	It flashes 1 Hz continuously to identify the slave (DCP_Identify).
Red	Exception error	Module in Exception status.
	Fatal event	Significant internal error (this indication is combined with a red network status LED).
Red, 1 blink	-	
Red, 2 blinks	-	
Red, 3 blinks	-	
Red, 4 blinks	-	
Red/Green ernating	Firmware update	DO NOT turn off the module. Turning it off during this phase could cause permanent damage.

10 DIGITAL INPUTS AND ASSOCIATED FUNCTIONS

ernatively, by displaying values on the display using the above options, the device can be configured to display status-dependent information on the digital inputs, provided it is correctly configured in advance. We can find 4 functions associated with digital inputs: **counter**, **stopwatch**, **tachometer** and **direct BCD display**.

If you want to display information from any other port such as a repeater (Modbus RTU/TCP, Profinet...) it is imperative that the digital inputs are disabled, otherwise only the information related to the digital inputs function will remain active.

Below is the connection scheme of the digital inputs, with the two connectors designed for this purpose:

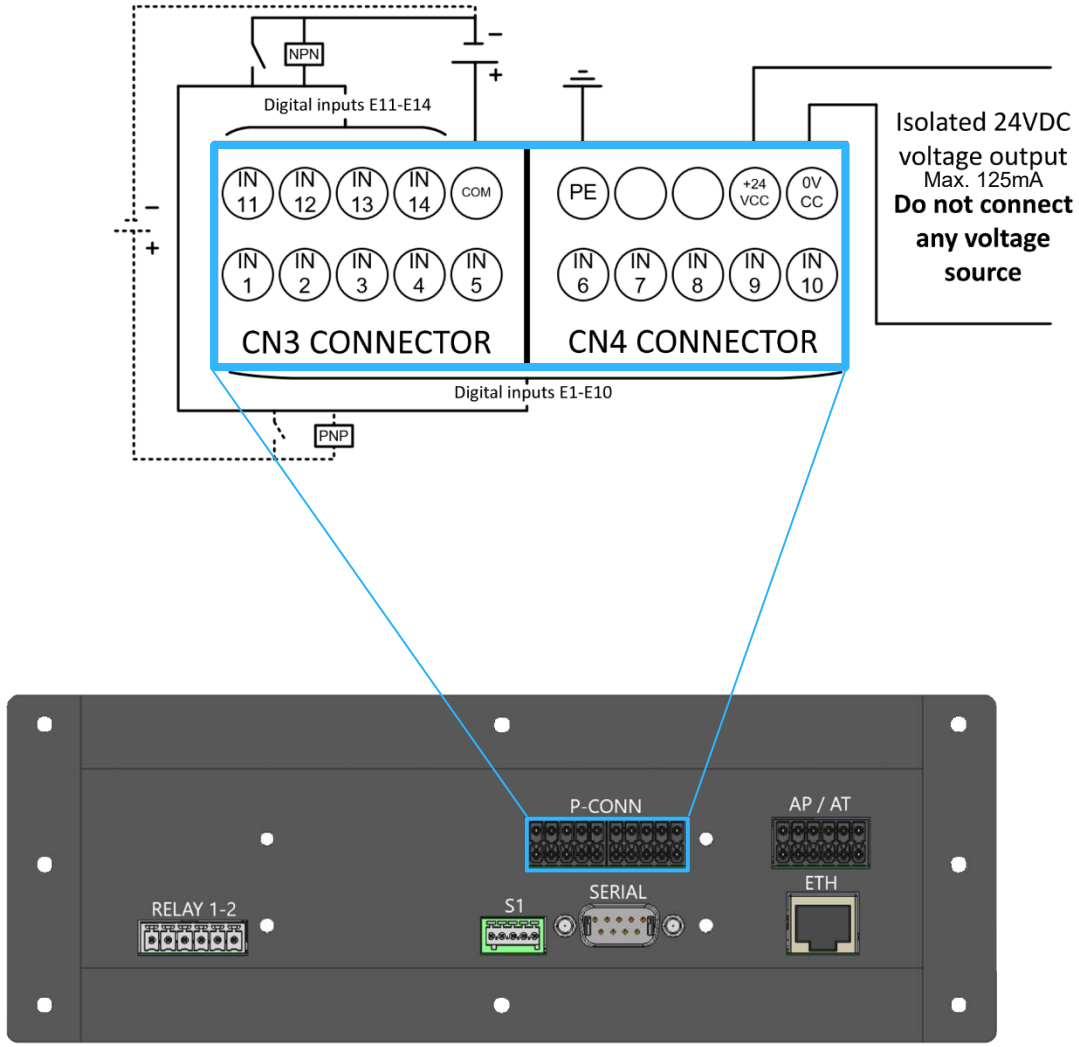


Fig. 31: Diagram of connection of the digital inputs

10.1 Configuring Digital Inputs

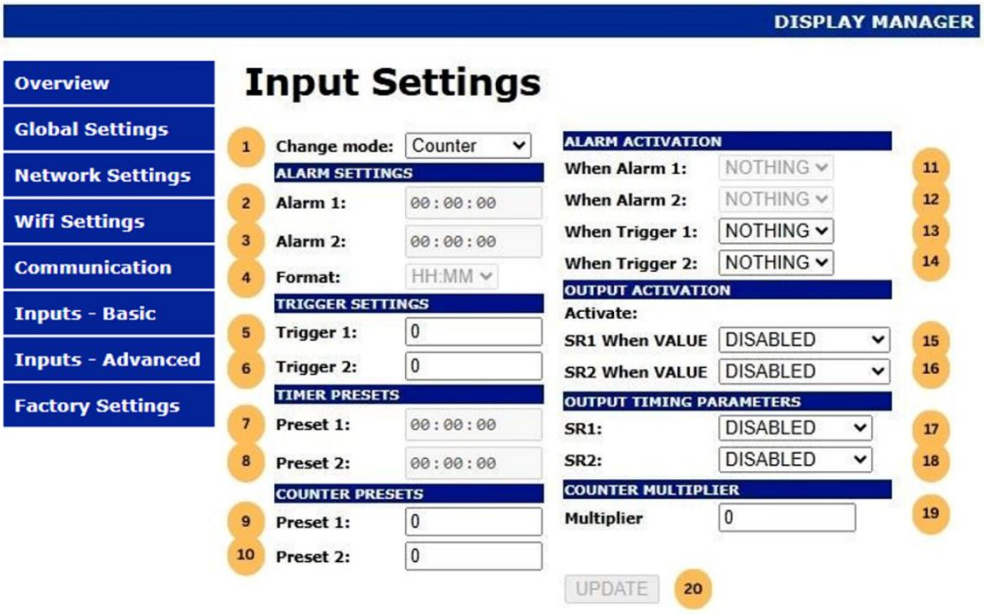


Fig. 32: Web Server Input Functionality Settings Screen

The digital input module can be configured using the dedicated pages on the web server "Inputs-Basic" and "Inputs-Advanced".

First, the configuration related to the "Inputs-Basic" page will be detailed. Mode selection in (1) will enable or disable the relevant controls for each functionality, as well as disable digital inputs. The words "Alarm" and "Trigger" in the Webserver are equivalent, as they both set a trigger threshold for their respective modes.

1. Select the mode of operation and behavior of the digital inputs. The functionalities to choose from are: **disabled, counter**, **stopwatch**, **tachometer** and **direct BCD input**.
2. Set the alarm threshold for ALARM 1 to stopwatch mode.
3. Set the alarm threshold for the ALARM 2 to stopwatch mode.
4. Allows you to select the displayed stopwatch format.
5. Sets the firing threshold for TRIGGER 1 in counter mode.
6. Set the firing threshold for the TRIGGER 2 in counter mode:
7. Sets the preload value (PRESET 1) to stopwatch mode.
8. Sets the preload value (PRESET 2) to stopwatch mode.
9. Sets the preload value (PRESET 1) to counter mode.
10. Sets the preload value (PRESET 2) to counter mode.
11. Allows you to configure the action performed when the threshold of ALARM 1 is exceeded in stopwatch mode:
    A. **NOTHING:** The ALARM 2 alarm is disabled. It is still disabled even if a value is set in ALARM 2.
    B. **RESET:** The stopwatch is immediately (asynchronous) brought to zero.
    C. **PRESET1:** The stopwatch is loaded immediately (asynchronously) with the value set to PRESET1.
    D. **PRESET2:** The stopwatch is loaded immediately (asynchronous) with the value set to PRESET2.

12. Allows you to configure the action performed when the threshold of the ALARM 2 is exceeded in stopwatch mode:
    A. **NOTHING:** The ALARM 2 alarm is disabled. It is still disabled even if a value is set in ALARM 2.
    B. **RESET:** The stopwatch is immediately (asynchronous) brought to zero.
    C. **PRESET1:** The stopwatch is loaded immediately (asynchronously) with the value set to PRESET1.
    D. **PRESET2:** The stopwatch is loaded immediately (asynchronous) with the value set to PRESET2.
13. Allows you to configure the action performed when the TRIGGER1 threshold is exceeded in counter mode:
    A. **NOTHING:** Shooting is TRIGGER1 disabled. It is still disabled even if there is a value set to TRIGGER 1.
    B. **RESET:** The counter is immediately brought (asynchronously) to zero.
    C. **PRESET1:** The counter is loaded immediately (asynchronously) with the value set to PRESET1.
    D. **PRESET2:** The counter is loaded immediately (asynchronously) with the value set to PRESET2.
14. Allows you to configure the action performed when the TRIGGER 2 threshold is exceeded in counter mode:
    A. **NOTHING:** TRIGGER 2 shooting is disabled. It is still disabled even if there is a value set to TRIGGER 2.
    B. **RESET:** The counter is immediately brought (asynchronously) to zero.
    C. **PRESET1:** The counter is loaded immediately (asynchronously) with the value set to PRESET1.
    D. **PRESET2:** The counter is loaded immediately (asynchronously) with the value set to PRESET2.
15. Defines the conditions for the activation of the SR1 relay.
16. Defines the conditions of SR2 relay activation

The available activation conditions are as follows:

Selection	Activation conditions
DISABLED	Not activated
= ALARM 1	Activates relay output if value = ALARM 1
>= ALARM 1	Activates relay output if value >= ALARM 1
= ALARM 2	Activates relay output if value = ALARM 2
<= ALARM 2	Activates relay output if value <= ALARM 2
> ALARM 1	Activates relay output if value > ALARM 1
< ALARM 1	Activates relay output if value < ALARM 1
> ALARM 2	Activates relay output if value > ALARM 2
< ALARM 2	Activates relay output if value < ALARM 2
= 0	Activates relay output if value = 0
<= AL1 && >= AL2	Activates relay output if value <= ALARM 1 and >= ALARM 2
>= AL1 && <= AL2	Activates relay output if value >= ALARM 1 and <= ALARM 2

Table 26: Relay activation conditions

17. Select the timing of the SR1 relay output according to the table.
18. Select the timing of the SR2 relay output according to the table.

19. Multiplier value of the counter from 1 to 20, so that with each ascending or descending pulse the counter will add or subtract the pulse by the multiplying factor
20. Pressing the UPDATE button is required to save the changes

Selection	Departure timing conditions
DISABLED	Not activated
0.5 s	Single pulse of 500 milliseconds.
0.8 s	Single pulse of 800 milliseconds
1 seconds	Single 1-second pulse
1.5 s	Single 1.5-second pulse
0.5 Hz	Output repeatedly active for 1 second and off for 1 second. The activation cycle persists until the alarm condition disappears.
2 Hz	Output repeatedly active for 500 milliseconds and off for 500 milliseconds. The activation cycle persists until the alarm condition disappears.
1.25 Hz	Output repeatedly active for 800 milliseconds and off for 800 milliseconds. The activation cycle persists until the alarm condition disappears.
1 Hz	Output repeatedly active for 1000 milliseconds and off for 1000 milliseconds. The activation cycle persists until the alarm condition disappears.
CONTINUOUS	Output activated as long as the alarm condition is met.

Table 27: Relay timing conditions

Now we will proceed to indicate the advanced functionalities related to digital inputs, which can be found in the Webserver tab "Inputs-Advanced":

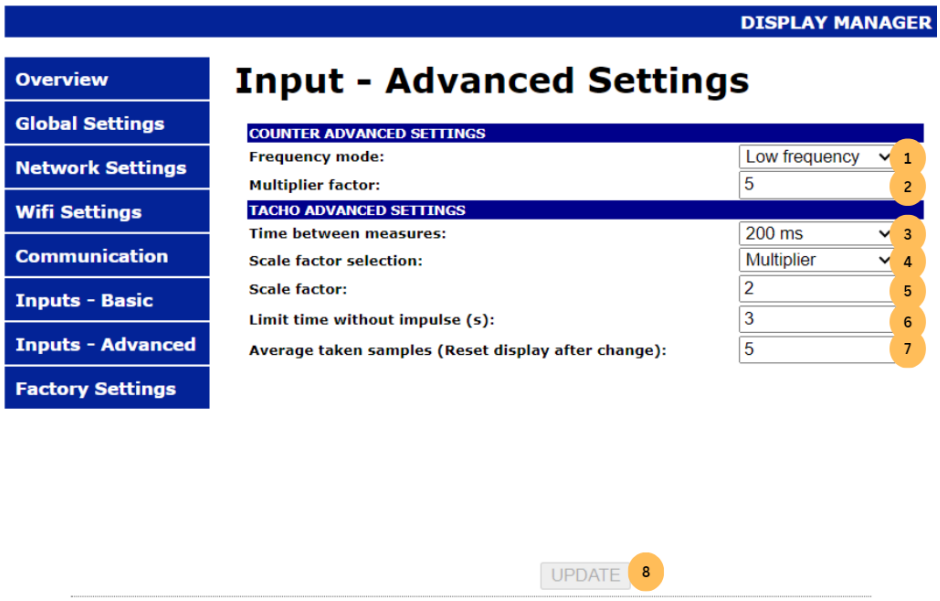


Fig. 33: Advanced Functionality Settings Screen for Web Server Entries

1. Selecting the frequency mode for the counter.
2. Multiplier factor of the count.
3. Time between measurements for the tachometer function.
4. Selection of the scale factor, between multiplier or divisor.
5. A scale factor that is applied to the output of the display.
6. Time limit without tachometer impulse.
7. Number of samples measured for averaging.
8. You need to press the UPDATE button to save your changes.

10.2 Functions associated with digital inputs

10.2.1 Counter

The counter function allows you to display the number of beats accumulated since the last reset. It has specific inputs for pulse input, counting direction (increase or decrease) as well as a reset input and two preset inputs. The control inputs used are as follows:

Digital Input	Action taken when activated/deactivated
E1	Increase the value of the meter
E2	Decrement the meter value
E3	Reset
E4	Loading Preselection 1
E5	Loading Preselection 2

Table 28: Control inputs used in the counter function.

The multiplicity of the count, i.e. the number of counts that are increased or decreased for each pulse is configurable by the counter multiplier factor, valid from 1 to 20.

On the other hand, you can select between low frequency mode (1 – 100 Hz) or high frequency mode (100 Hz – 10 Khz) depending on the frequency of the counting signal. In low-frequency mode, filtering is applied to prevent bounces from contacts from falsifying the measurements.

The configuration of up to two TRIGGERs, which can be set to any value via the web server, is allowed. The conditions by which the TRIGGERs will be triggered can also be configured according to the following table.

Selection	Activation conditions
DISABLED	Not activated
= ALARM 1	Activates relay output if value = ALARM 1
>= ALARM 1	Activates relay output if value >= ALARM 1
= ALARM 2	Activates relay output if value = ALARM 2
<= ALARM 2	Activates relay output if value <= ALARM 2
> ALARM 1	Activates relay output if value > ALARM 1
< ALARM 1	Activates relay output if value < ALARM 1
> ALARM 2	Activates relay output if value > ALARM 2
< ALARM 2	Activates relay output if value < ALARM 2
= 0	Activates relay output if value = 0
<= AL1 && >= AL2	Activates relay output if value <= ALARM 1 and >= ALARM 2
>= AL1 && <= AL2	Activates relay output if value >= ALARM 1 and <= ALARM 2

Table 29: Activation conditions for the counter function.

ernatively, if desired, a TRIGGER condition can be linked to one of the two relay outputs. The timing of the action is also configurable, by single pulses of a certain length or by frequencies of periodic activation. The trigger condition is restarted by either the reset input or by the output of the condition that caused the TRIGGER in the first instance.

10.2.2 Stopwatch

The stopwatch function allows the counting of time, in multiple scales and units. The control inputs used in this functionality are as follows:

Digital Input	Action taken when activated/deactivated
E1	RUN Stopwatch = ON/ STOP Stopwatch = OFF
E2	Reset
E3	Loading Preselection 1
E4	Loading Preselection 2
E5	Increase Stopwatch = ON/ Decrement Counter = OFF

Table 30: Control inputs used in the stopwatch function.

Depending on the number of digits set on the display, the displayed format is automatically adjusted.

Available digits	Format Displayed
3 Digits	D.DD
4 Digits	DD.DD

Table 31: Variability of the displayed format depending on the number of digits.

The configuration of up to two ALARMs is allowed, which can be set to any value via the web server. The conditions under which the ALARMs will be triggered can also be configured according to the following table.

Selection	Activation conditions
DISABLED	Not activated
= ALARM 1	Activates relay output if value = ALARM 1
>= ALARM 1	Activates relay output if value >= ALARM 1
= ALARM 2	Activates relay output if value = ALARM 2
<= ALARM 2	Activates relay output if value <= ALARM 2
> ALARM 1	Activates relay output if value > ALARM 1
< ALARM 1	Activates relay output if value < ALARM 1
> ALARM 2	Activates relay output if value > ALARM 2
< ALARM 2	Activates relay output if value < ALARM 2
= 0	Activates relay output if value = 0
<= AL1 && >= AL2	Activates relay output if value <= ALARM 1 and >= ALARM 2
>= AL1 && <= AL2	Activates relay output if value >= ALARM 1 and <= ALARM 2

Table 32: Activation conditions for the stopwatch function.

In addition, if desired, an ALARM condition can be linked to one of the two relay outputs. The timing of the action is also configurable, by single pulses of a certain length or by frequencies of periodic activation. The trigger condition is restarted by either the reset input or by the output of the condition that caused the ALARM in the first instance.

10.2.3 Tachometer

The tachometer function allows the measurement of the frequency of a signal supplied on a digital input. For its correct operation, it is necessary to connect the output signal of the tachometer to the digital input 13 (E13) and its ground to the pin marked COM.

The device displays the frequency value of the input signal in Hertz. The current measurement range is 1 Hz up to 20 KHz, with any type of waveform having a single crossover by zero.

On the other hand, you can adjust different parameters related to this functionality in the Webserver:

- Time between measurements for the tachometer function, to be chosen between 200 ms and 20 s. As this time increases, variations in the displayed signal are reduced if it is of high frequency. On the other hand, it is not advisable to put a very high time between measurements if the signal to be measured is of low frequency, since the response time will be high.
- Selection of the scale factor, between multiplier or divisor. This option allows you to choose whether the scale factor that is subsequently indicated to apply to the output of the display is multiplied or divided.
- A scale factor that is applied to the output of the display, with the aim of scaling the value that is displayed depending on the input frequency. The scale factor must be an integer from 1 to 15000.
- Time limit without tachometer impulse in seconds. This is the time from which no frequency pulse will be applied to the pins of the tachometer until it is desired to be visible that no pulse is present. When the programmed time is exceeded, the display displays the value 0. This value can range from 1 to 10 seconds, and it is recommended that your default value is 1 second.
- Number of samples measured for the average displayed on the display. The number of samples is programmable from 1 to 100 and allows the oscillations of the input signal to be smoothed out when it varies significantly. Important, when changing the sample value on the display, you need to reset the display to apply the changes.

10.2.4 BCD Direct

The direct BCD function allows the display of a 3-digit number, controlled by 14 digital inputs.

Entries E1-E4 control digit 1, the rightmost digit of the display.

The E5-E8 inputs control digit 2, the middle digit.

Entries E9-E12 control digit 3, the third digit on the right.

The position of the decimal point and the negative symbol (-) is controlled by 2 additional digital inputs, E13 and E14.

BCD digit	Digital ticket statuses
Units	E4 E3 E2 E1
Tens	E8 E7 E6 E5
Hundreds	E12 E11 E10 E9

Table 33: Mapping Inputs to Digits for Direct BCD

Decimal point	E13 and E14 digital ticket statuses
None	E14 = 0, E13 = 0
Tens	E14 = 0, E13 = 1
Hundreds	E14 = 1, E13 = 0
Symbol -	E14 = 1, E13 = 1

Table 34: Decimal Point Setting for Direct BCD

For example, to visualize:

8	5.	3

The following must be coded:

Point position		8				5				3			
0	1	1	0	0	0	0	1	0	1	0	0	1	1
E14	E13	E12	E11	E10	E9	E8	E7	E6	E5	E4	E3	E2	E1

Table 35:Direct BCD Practical Example

11 RESET FACTORY SETTINGS

To restore all Ethernet settings to factory settings if the user makes a mistake in their configuration, the Webserver Factory settings tab has been enabled, through which pressing the Reset button executes this action.

This tab can be accessed via WiFi, resetting all Ethernet settings, but resetting the display is necessary for the configuration changes to occur.

IMPORTANT: Do not reset your Ethernet settings unless it is essential!

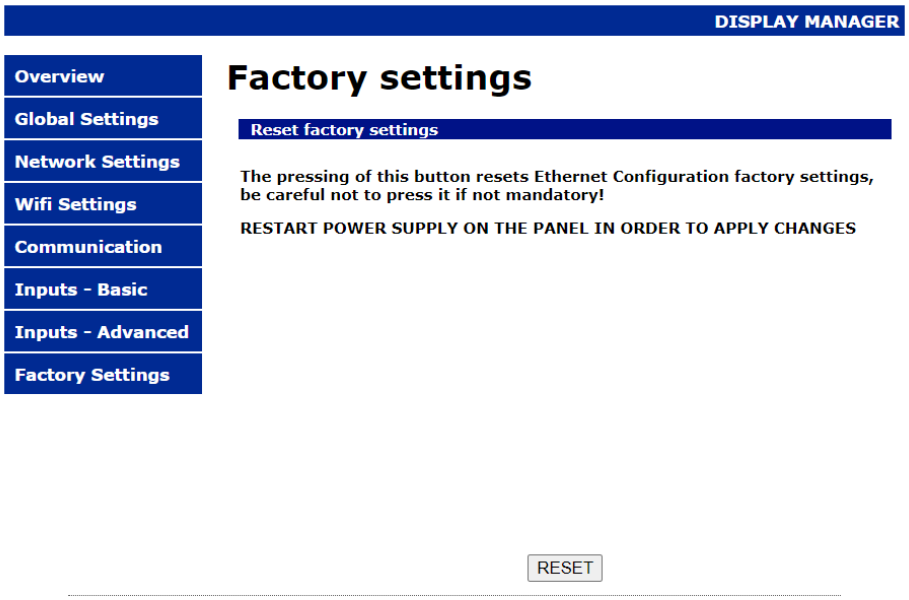


Fig. 34: Advanced Functionality Settings Screen for Web Server Entries

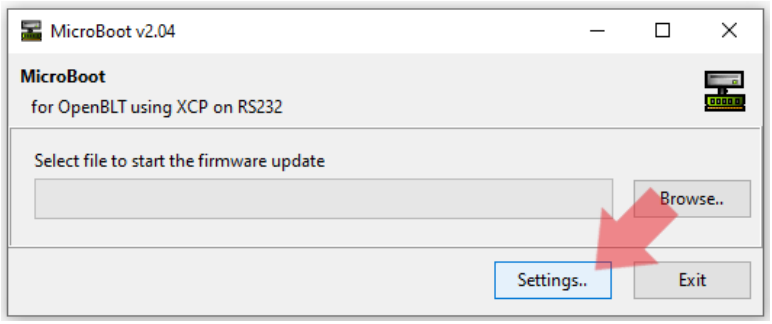
12 UPDATE THE VISUALIZER

Under the need to update the *display firmware*, it is possible to perform the update using a PC with the **MicroBoot** (https://www.lartet.com/en/download-center/) program.

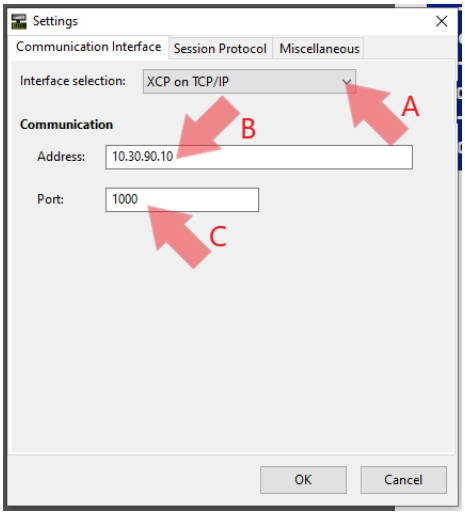
IMPORTANT: The update must be carried out via a wired ETHERNET connection.

The steps to update the *firmware* are as follows:

1. Start the program and go into settings.



2. Check the program settings:



A. Establish TCP/IP communication.
B. Configure the IP address that has been defined for the wired network.
C. Configure port 1000.

3. Accept the settings and select the desired firmware.

During the firmware update process, the display will display "Pr1".

If the update takes too long to start, cancel the process, review the IP configured on the MicroBoot, the Ethernet cable and the Firewall rules and repeat the firmware upload process.

The program itself has a *"timeout" warning* if it detects that too much time has passed, but it has no way of knowing if the process has finished and does not interrupt the loading of the process. It is only a time warning.

NOTE: If the charging process is interrupted (network drop, cable disconnection, etc.), the update is incomplete and the display lacks a valid program. In these circumstances, the only way to charge a *firmware* Proper system involves turning on the computer and loading a valid program (using MicroBoot) to the emergency IP address **192.168.1.100** when the display shows "Pr0".

ANNEX 1: Sending information with "Hercules" for TCP, UDP and serial communication

When communicating using the "Hercules" program, certain aspects must be considered so as not to make a mistake when sending values in decimal or hexadecimal.

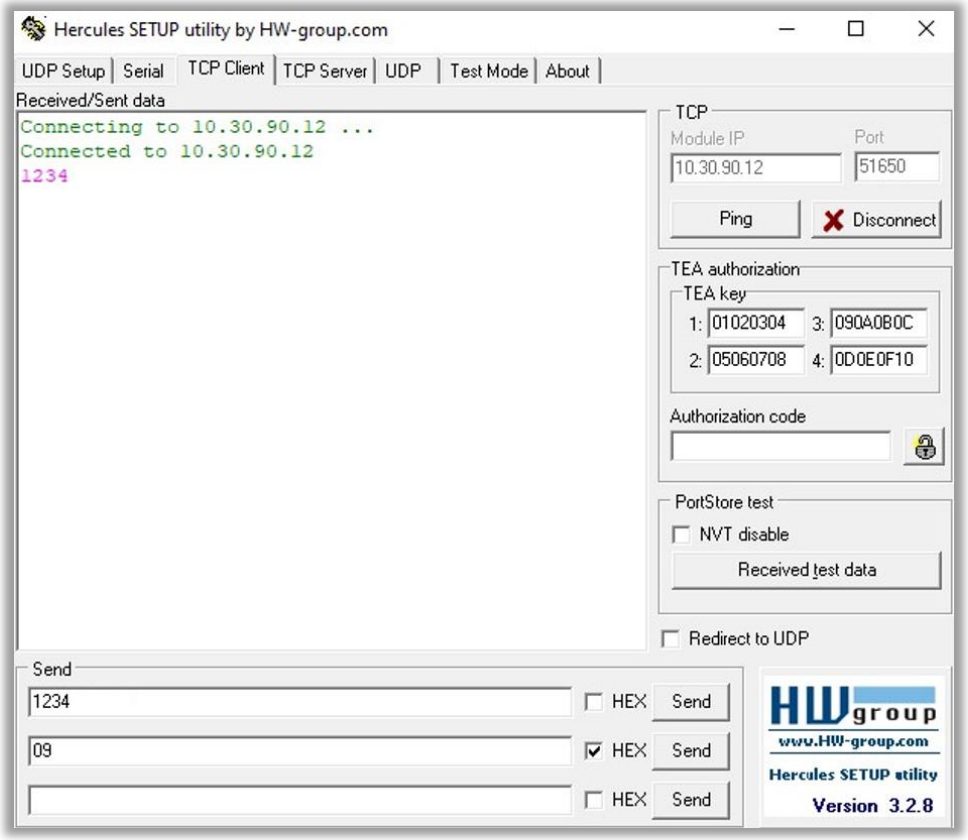


Fig. 35: Example in TCP using the "Hercules" program. "1234" is displayed.

First, it is necessary to establish communication information as defined on the web server, according to the type of communication established.

There are several ways to send information:

- Directly typing the values will be applied one at a time as separate messages.
- Write the commands under "Send", this allows the entire frame to be sent simultaneously. The program will automatically read numbers as decimals and letters as ASCII characters. To indicate to the program that we want to enter a hexadecimal number, a "$" must be applied before the value.
- Type the commands under "Send" by selecting the "HEX" box. This allows the user to directly write ASCII values in a hexadecimal manner, without the need to add symbols.

This way of sending information is common to TCP, UDP, and serial modes.

Example of web server configuration for TCP:

1. "General Settings" window, **DATA PORT** = ETHERNET.
2. "Network Settings" window, set the network settings correctly, in this case **IP Addr** = 10.30.90.12 as defined in "Hercules".
3. "Communication Settings" window, **PROTOCOL** = TCP.

If you want to enable the display, you can configure an "ENDBLOCK", but it is not necessary for communication.

Example of web server configuration for UDP:

1. "General Settings" window, **DATA PORT** = ETHERNET.
2. "Network Settings" window, set the network settings correctly, in this case **IP Addr** = 10.30.90.12 as defined in "Hercules".
3. "Communication Settings" window, **PROTOCOL** = UDP.

If you want to enable the display, you can configure an "ENDBLOCK", but it is not necessary for communication.

Web server configuration example for RS-232 serial:

1. General Settings window, **DATA PORT** = SERIAL.
2. "Communication Settings" window:
    - **ADDRESS** = 14.
    - **PROTOCOL** = ASCII.
    - **INTERFACE** = RS232.
    - **HEADER** = 02h AL AH (one has been chosen, the message must be consistent with the established configuration.)
    - **ENDBLOCK** = <CR LF> 0Dh 0Ah
    - **REPLY** = NONE
    - **BAUDRATE** = 19200
    - **PARITY** = NONE
    - **DATA SIZE** = 8 bits
    - **STOP BITS** = 1 bit
    - **MSG. OFFSET** = 0
    - **VIEW** = NORMAL
    - **MSG. CURSOR** = 0

To send the message with "Hercules" a USB to RS-232 converter is used. To find out which "COM" port is set to, you must check it in the Windows "Device Manager". The program is configured identically to the server, and the following frame is sent:

02 34 31 31 32 33 34 0D 0A

This frame includes all the parameters necessary to display 1234 including the headers and frame endings.

ANNEX 2: Sending Information with "QModMaster" for MODBUS TCP and MODBUS RTU Communication

When communicating over MODBUS, using QModMaster there are few differences when working over RTU or TCP.

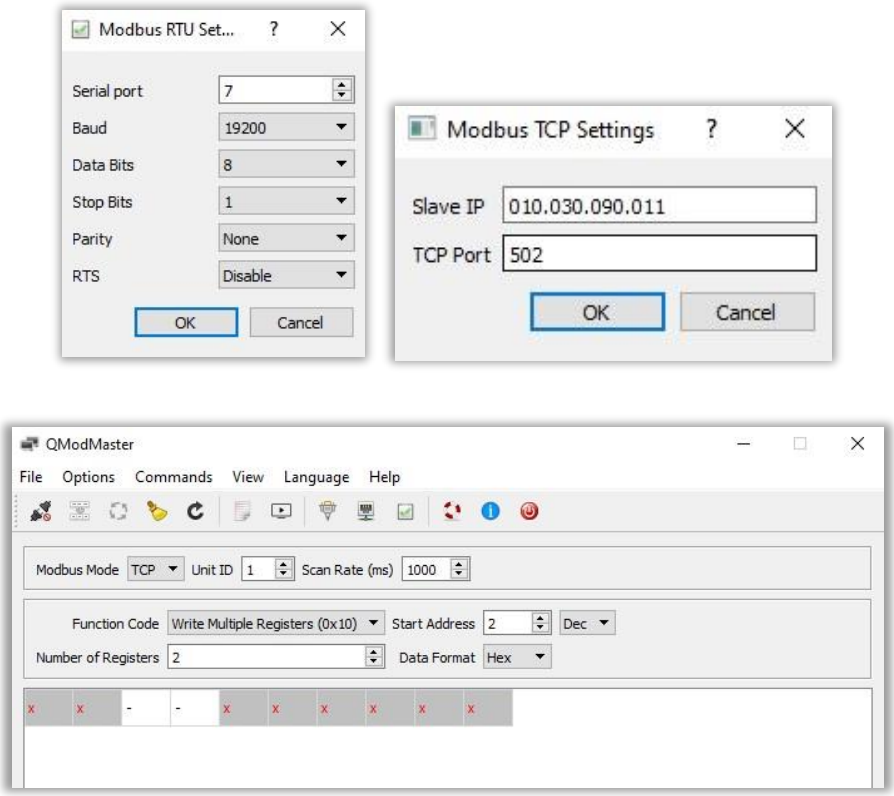


Fig. 36: How QModMaster works.

First, you need to check that the connection settings are correct. These settings vary between RTU or TCP, but in both cases, they must be related to the information set on the display's web server. In MODBUS TCP the port is always 502.

Next, you must configure the "Unit ID" and the "Scan Rate", if there is only one computer connected, an image like the previous one will be displayed.

At this point, the type of frame to be sent will be configured according to the value to be displayed, following the protocols explained in section 6.4. The parameters to be configured are as follows:

- **Function Code:** You must select the type of action you want. Whether it's reading or writing in registers or coils. The drop-down menu specifies all the possibilities as well as the corresponding function number.
- **Start Address:** Indicates the first register to which to read or write. It is advisable to keep its value in decimal.
- **Number of Registers:** You must indicate how many registers you want to work with.

- **Data Format:** This drop-down menu allows you to change the contents of the registers to the desired system. It is very useful to enter data in the most comfortable way possible, if there is already data written the program converts it automatically.

Example 1: You want to type "HOLA" into the display using MODBUS TCP.

First, you adjust the settings on the web server as follows:

1. "General Settings" window, **DATA PORT** = ETHERNET.
2. "Network Settings" window, set the network settings correctly, in this case **IP Addr** = 10.30.90.11 as defined in "QModMaster".
3. "Communication Settings" window, **PROTOCOL** = MODBUS

Second, the "Write Multiple Registers" function (10h) is used to write 2 registers with the content "484Fh 4C41h" from address 0.

The configuration in QModMaster would result as follows:

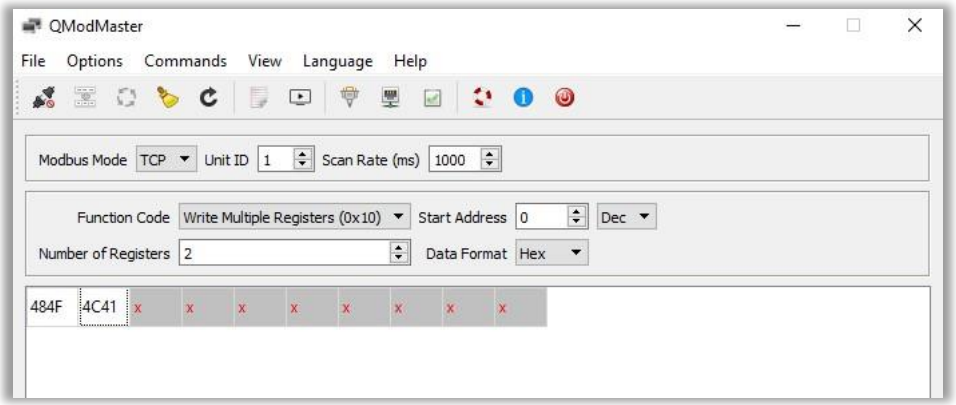


Fig. 37: Example of communication in QModMaster. Send to the display "HOLA".

Once the configuration is done, you proceed to click the connector button at the top left of the panel. This will initiate communication with the display.

Finally, clicking the button immediately to the right of the connector will send the frame and show "HOLA" in the display.

NOTE: It is very useful to open the "Monitor Bus", in the "View" tab. Doing so opens a window showing each of the frames that are sent and received during communication.

Example 2: You want to type "HOLA" into the dispaly using MODBUS RTU.

First, you adjust the settings on the web server as follows:

1. "General Settings" window, **DATA PORT** = SERIAL.

2. "Communication Settings" window:
   - **ADDRESS** = 1.
   - **PROTOCOL** = MODBUS RTU.
   - **INTERFACE** = RS232.
   - **BAUDRATE** = 19200
   - **PARITY** = NONE
   - **DATA SIZE** = 8 bits
   - **STOP BITS** = 1 bit

To send the message with "QModMaster" a USB to RS-232 converter is used. To find out which "COM" port is set to, you must check it in the Windows "Device Manager". The program is configured identically to the RTU server.

To send the information logs is done identically to the previous example in MODBUS TCP.

ANNEX 3: Configure and use function blocks to send information using a PLC.

The examples that make up this annex have been made with a PLC "CPU 1512 SP-1 PN".

MODBUS_RTU: The following blocks are used to communicate with the display:

First, the blocks used to set up and make the connection are displayed.

At the "PORT" entrance, the corresponding communications card is used. In our case, "CM PtP".

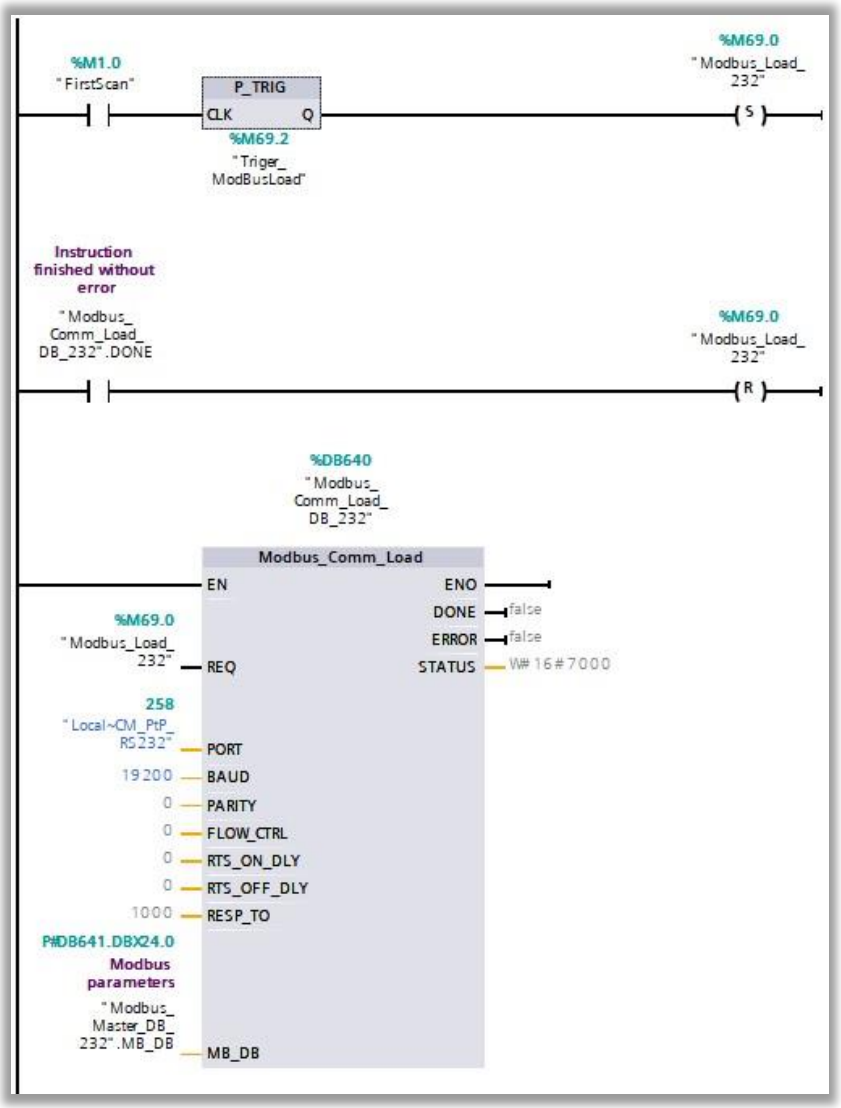
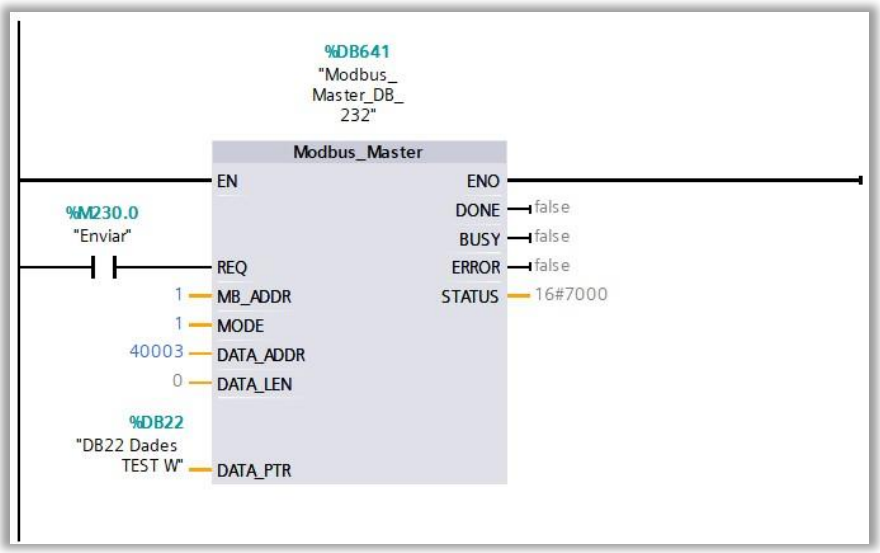


Fig. 38: Communication configuration blocks.

Second, the blocks and variables used to generate and send a log write message are displayed.

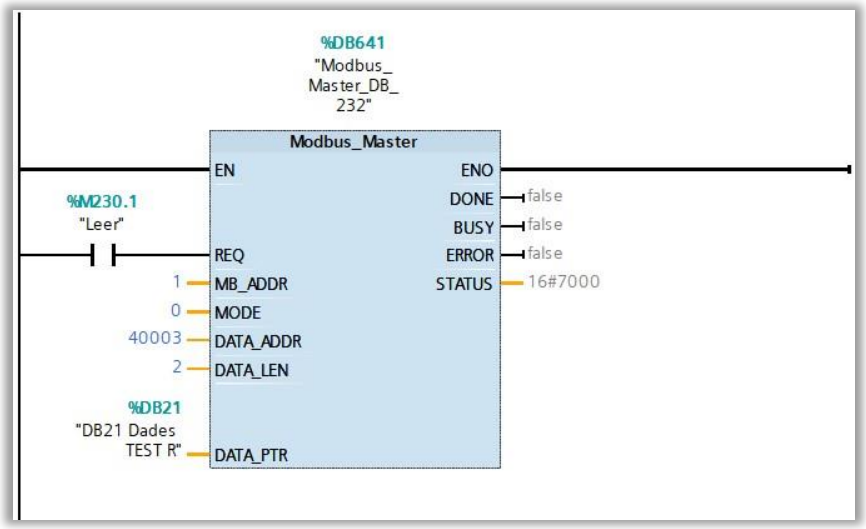


DB22 Dades TEST W

		Nombre	Tipo de datos	Offset	Valor de arranq...	Remanen...	Accesible d...	Escrib...	Visible en ..	Valor de a..
1		Static								
2		W_W1	Word	0.0	16#1E61		☑	☑	☑	
3		W_W2	Word	2.0	16#3034		☑	☑	☑	

Fig. 39: Generation of the content of the registers to be sent.

Finally, the blocks and variables used to generate and send a log read message are shown.



DB21 Dades TEST R

	Nombre	Tipo de datos	Offset	Valor de arranq...	Remanen...	Accesible d...	Escrib...	Visible en ...	Valor de a...
1	Static								
2	R_W1	Word	0.0	16#0					
3	R_W2	Word	2.0	16#0					

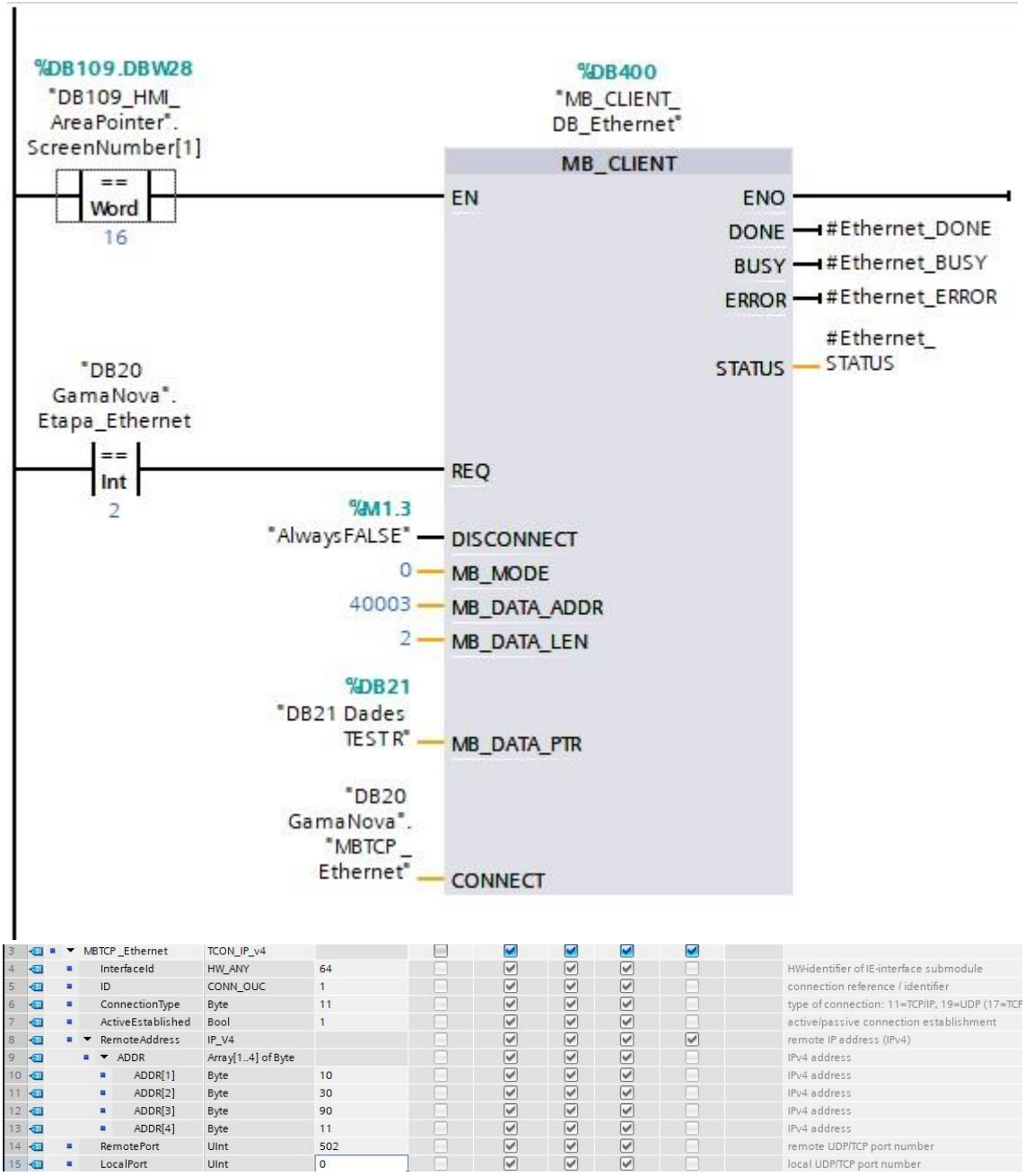
Fig. 40: Reading the registers of the response.

The block is the same "Modbus_Master". By changing the input values, you configure it to be a write or read message, as well as the number of registers or their location.

IMPORTANT: When configuring the "Modbus_Master" you must consult the documents of the same so as not to make mistakes in any of the entries of the block. Depending on the MODBUS function used and its content, it will be necessary to modify the entries to adapt them to the needs of each information submission.

MODBUS_TCP: The modules for MODBUS_TCP "Modbus_master" are the same as for RTU.

These blocks initiate communication via MODBUS_TCP. The "MBTCP:Ethernet" variable needs to be set correctly.



3		MBTCP_Ethernet	TCON_IP_v4		☐	☑	☑	☑	☑	
4		InterfaceId	HW_ANY	64	☐	☑	☑	☑	☐	HW-identifier of IE-interface submodule
5		ID	CONN_OUC	1	☐	☑	☑	☑	☐	connection reference / identifier
6		ConnectionType	Byte	11	☐	☑	☑	☑	☐	type of connection: 11=TCP/IP, 19=UDP (17=TCP
7		ActiveEstablished	Bool	1	☐	☑	☑	☑	☐	active/passive connection establishment
8		RemoteAddress	IP_V4		☐	☑	☑	☑	☑	remote IP address (IPv4)
9		ADDR	Array[1..4] of Byte		☐	☑	☑	☑	☐	IPv4 address
10		ADDR[1]	Byte	10	☐	☑	☑	☑	☐	IPv4 address
11		ADDR[2]	Byte	30	☐	☑	☑	☑	☐	IPv4 address
12		ADDR[3]	Byte	90	☐	☑	☑	☑	☐	IPv4 address
13		ADDR[4]	Byte	11	☐	☑	☑	☑	☐	IPv4 address
14		RemotePort	UInt	502	☐	☑	☑	☑	☐	remote UDP/TCP port number
15		LocalPort	UInt	0	☐	☑	☑	☑	☐	local UDP/TCP port number

Fig. 41: Setting up communication in MODBUS_TCP.

The IP must be configured according to the value assigned in the display's web server. To do this, generate the variable "MBTCP_Ethernet" and write in its type "TCON_IP_v4", in this way all the fields are developed automatically. Remember to set the desired IP and "ConnectionType" = 11 (TCP/IP).

The entry "MB_MODE" (1 or 0) indicates whether the communication is read or write registers.

UDP: Blocks downloaded from the Siemens website are used to communicate in UDP. Specifically "S7-1200/S7-1500" (LOpenUserComm_Udp).

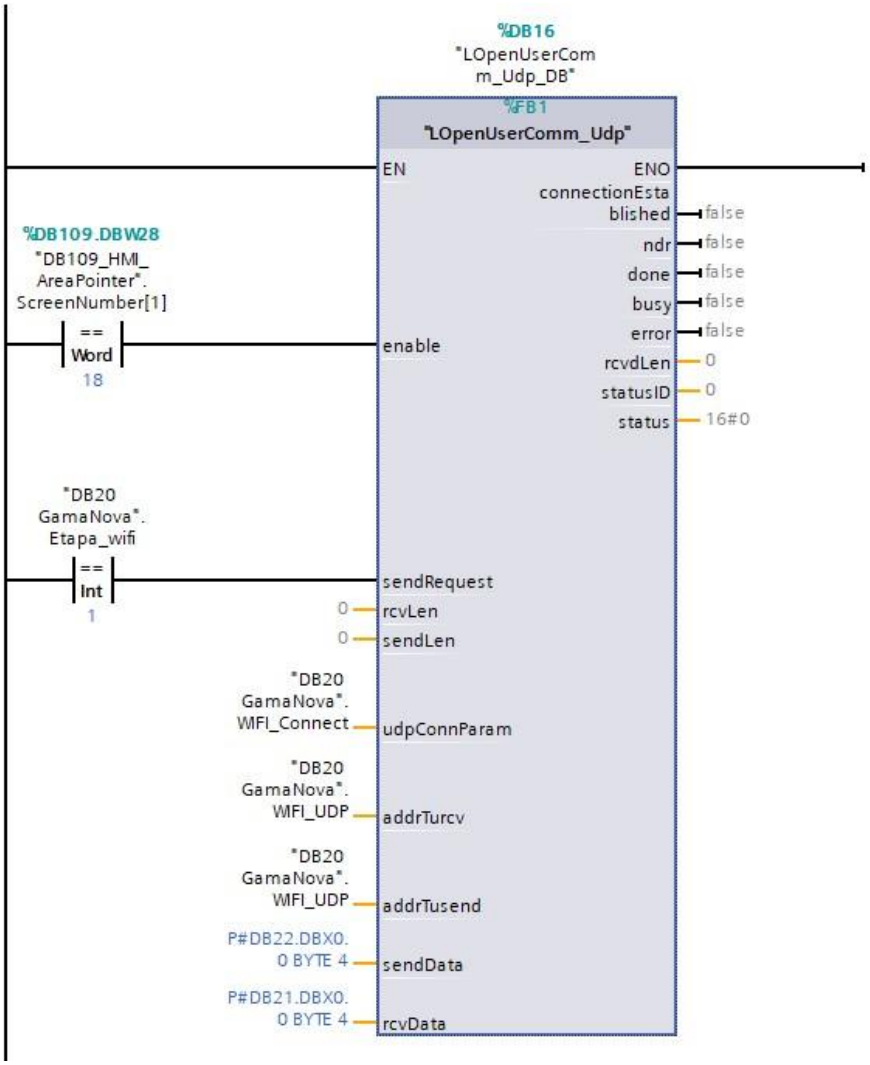
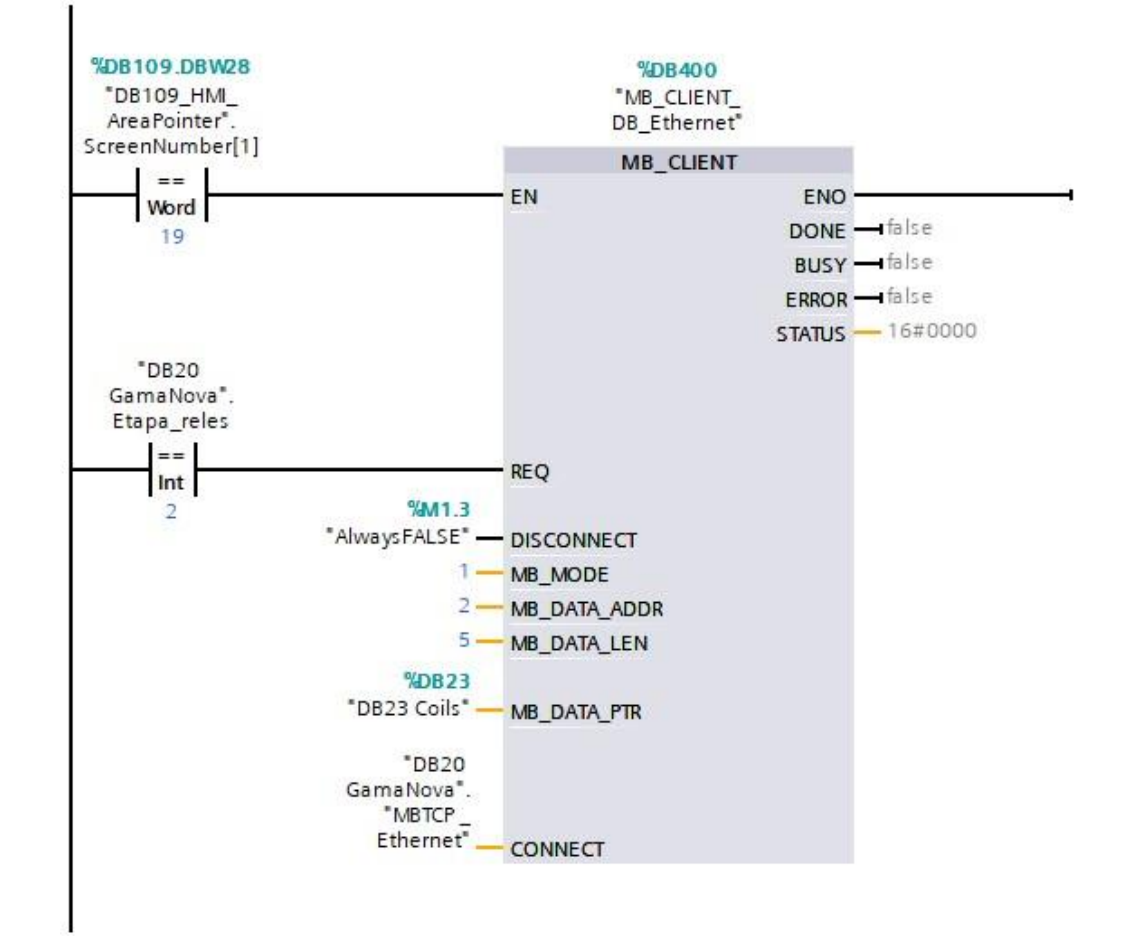


Fig. 42: Block "LOpenUserComm_Udp" used to perform UDP communication.

The commented module takes care of performing the configuration automatically, once the inputs have been established in the desired way.

WIFI: An external module (TPLINK) is used for WIFI communication, so the PLC performs the communication as if it were a wired network.

RELAYS: To activate the relays or flashing, Modbus must be used. In our case we used the module shown above to Modbus_TCP. The difference is clearly visible, since in this case we work on the direction "MB_DATA_ADDR" = 2 and the length of the data is "MB_DATA_LEN" = 5, since there are 5 modifiable elements (4 relays + flashing).



DB23 Coils

	Nombre	Tipo de datos	Offset	Valor de arranq...	Remanen...	Accesible d...	Escrib...	Visible en ...	Valor de a...
1	Static								
2	Rele_1	Bool	0.0	false		☑	☑	☑	
3	Rele_2	Bool	0.1	false		☑	☑	☑	
4	Rele_3	Bool	0.2	false		☑	☑	☑	
5	Rele_4	Bool	0.3	false		☑	☑	☑	
6	Parpadeo	Bool	0.4	false		☑	☑	☑	

Fig. 43: Configuration and generation of the content to be sent to the "Coils". Relay and flicker management.

In "DB23 Coils", you type the state you want to activate or deactivate the (1 to 4) relays or flashing (5).

DECLARATION OF CONFORMITY

CE

Tetralec Electronica Industrial S.L.
c/ Severo Ochoa, 80
Font del Radium Industrial Estate
08403 Granollers

As a brand team builder **LARTET:**

Model: DN-107. V2 in all its versions.

We declare under our sole responsibility that the aforementioned
Product complies with the following European directives:

Directive: 2014/35/EU Low Voltage Directive.
UNE-EN 60204-1:2019 Machinery Safety Standard.
Electrical equipment of the machines.

Directive: EMC 2014/30 EU Electromagnetic Compatibility Directive.
UNE-EN 61000-6-2:2019 Generic standards. Immunity in Industrial environments.
UNE-EN 61000-4-4:2013 Rapid electrical transient immunity tests
in bursts.
UNE-EN 61000-4-2:2010 Electrostatic discharge immunity test.

Directive 2011/65/EU: Restrictions on the use of certain substances
in electrical and electronic devices.

Granollers, April 28, 2022