

---

# **MPC-374**

## **User's Manual**

1.1

## Document history

Date	Version	Author	Remarks
04/04/2016	1.0	Sara Prior	User's Manual including new features such as M-Bus and FTP client configuration
18/05/2016	1.1	Sara Prior	Added Modbus TCP functionality

# Content

<b>DOCUMENT HISTORY</b>	<b>2</b>
<b>CONTENT</b>	<b>3</b>
<b>1 INTRODUCTION</b>	<b>5</b>
<b>2 SETTING UP CONNECTION TO THE DEVICE</b>	<b>6</b>
2.1 USB connection	7
2.2 ETHERNET Connection	8
2.3 GPRS connection	9
2.4 MPC-374 connection diagrams	10
<b>3 "START" TAB</b>	<b>11</b>
3.1 MPC-374 Basic Information	11
3.2 Configuration files	12
3.3 Status indicators	13
<b>4 "ANALOG INPUTS" TAB</b>	<b>14</b>
4.1 Current Output Sensors	14
4.2 Analog Inputs configuration	16
<b>5 "DISCRETE INPUTS" TAB</b>	<b>17</b>
<b>6 "DISCRETE OUTPUTS" TAB</b>	<b>18</b>
<b>7 "COMMUNICATION" TAB</b>	<b>20</b>
7.1 Communication > Ethernet	20
7.2 Communication > GPRS	21
7.3 Communication > UARTs	24
7.4 Communication > Virtual interfaces	25
7.5 Communication > Data transfer	27
7.6 Communication > Routing	30
7.7 Communication > Connected devices	33
7.8 Communication > Modbus devices	38

---

<b>8</b>	<b>"ARCHIVES" TAB</b>	<b>41</b>
<b>9</b>	<b>"TIME PARAMETERS" TAB</b>	<b>43</b>
<b>10</b>	<b>ABBREVIATIONS AND EXPLANATIONS</b>	<b>44</b>
<b>11</b>	<b>SAFETY INSTRUCTIONS</b>	<b>45</b>
<b>12</b>	<b>MANUFACTURER'S WARRANTY</b>	<b>46</b>
<b>13</b>	<b>TECHNICAL DATA</b>	<b>46</b>
<b>13.1</b>	<b>Communication interfaces</b>	<b>46</b>
<b>13.2</b>	<b>Galvanic insulation</b>	<b>47</b>
<b>13.3</b>	<b>Indication</b>	<b>47</b>
<b>13.4</b>	<b>Power supply</b>	<b>47</b>
<b>13.5</b>	<b>Construction</b>	<b>47</b>
<b>13.6</b>	<b>Climate conditions</b>	<b>48</b>
<b>13.7</b>	<b>Safety parameters</b>	<b>48</b>
<b>13.8</b>	<b>Other parameters</b>	<b>48</b>

# 1 Introduction

This Manual will show you how to configure the MPC-374 datalogger/controller by using the MPC-374 configurator software.

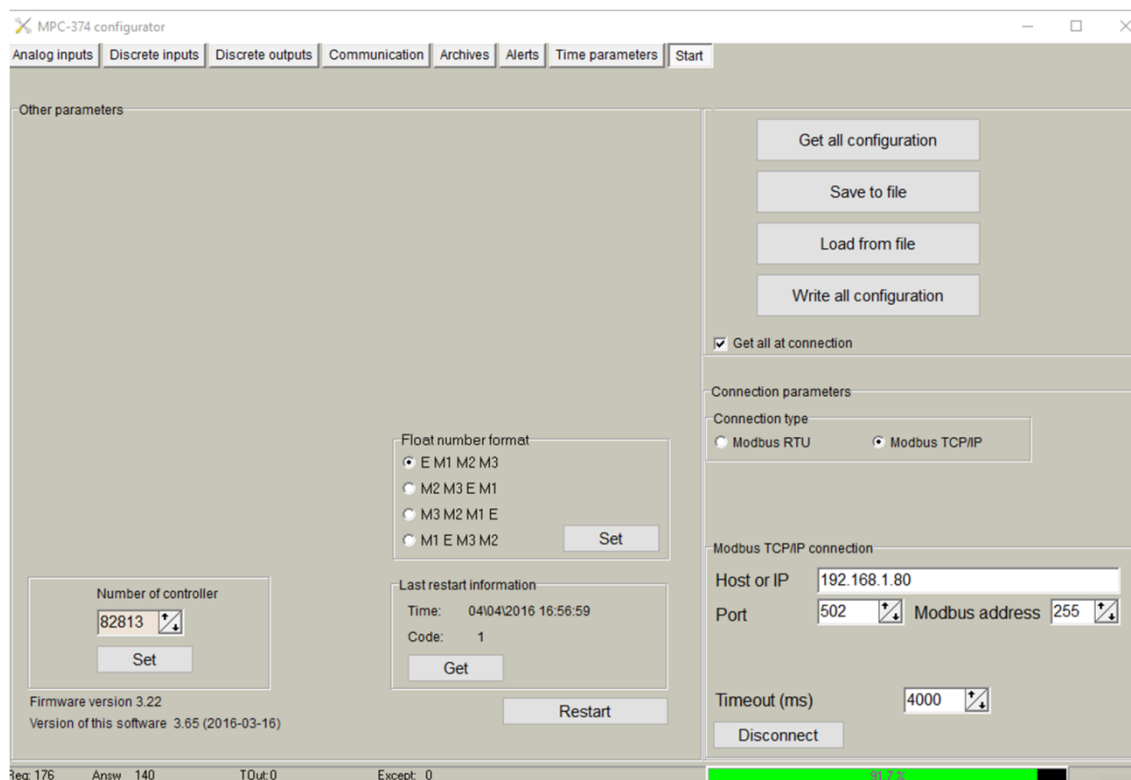
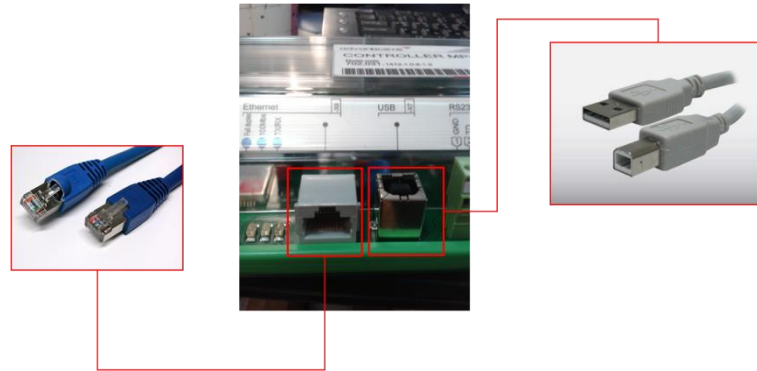


Figure 1. MPC-374 Configurator

## 2 Setting up connection to the device

In order to configure the controller, user must connect its PC to the device by using any of the following interfaces:

1. USB port
2. ETHERNET interface
3. Through a GPRS connection (only accessible after configuring GPRS APN, user and password inside the controller)



**Figure 2. MPC-374 connection interfaces**

**NOTE:** Not all the models support above interfaces. Check your ordering code first.

## 2.1 USB connection

Steps to be followed:

- Connect an USB cable class B to the correspondent port
- Open MPC-374 Configuration Tool
- Set up "**Connection Parameters**" frame
  - o Select "Modbus RTU" option under "Connection type"
  - o Configure "Bode" and "Parity" parameters; default values are:"19200" , "none"
  - o Select COM port number assigned by your PC to the USB port
  - o Click on "**Get all**" to establish connection with controller.

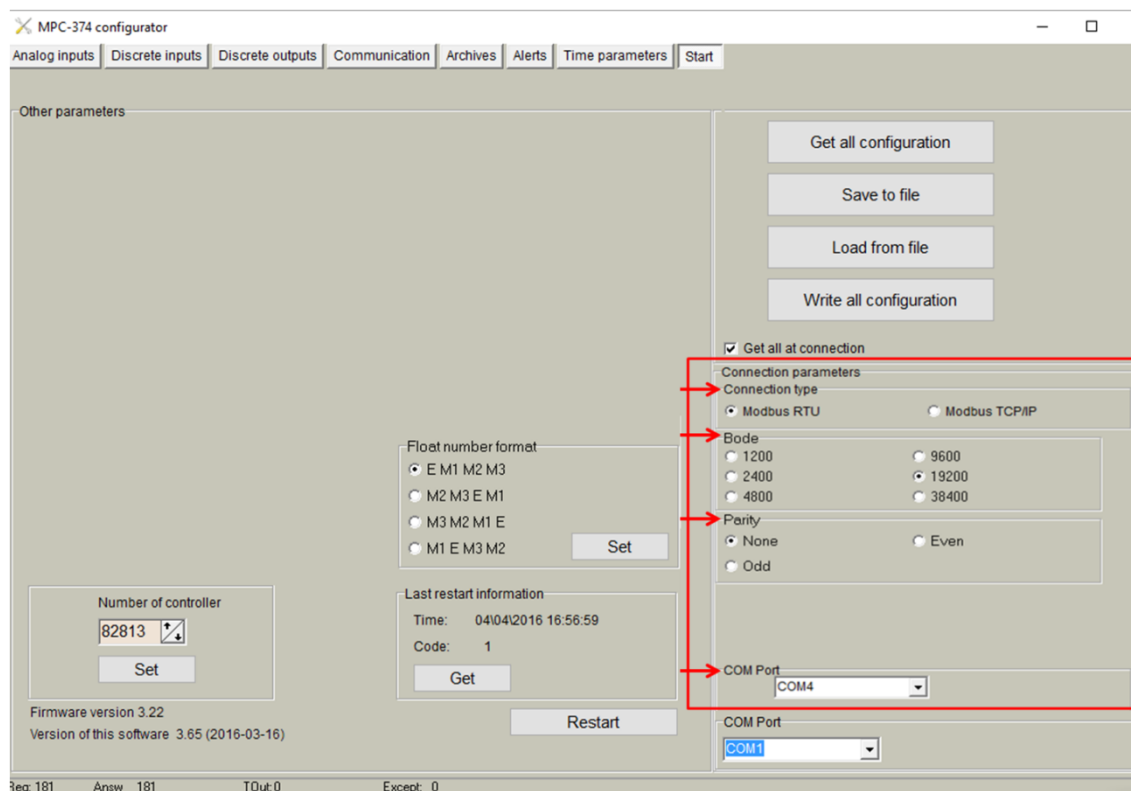


Figure 3. USB connection set up

## 2.2 ETHERNET Connection

Steps to be followed:

- Connect a RJ45 Ethernet cable to the correspondent port.
- Open MPC-374 Configuration Tool
- Set up "**Connection Parameters**" frame
  - o Select "Modbus TCP/IP" option under "Connection type"
  - o Add controller IP address and TCP port in "Host or IP" and "Port" text boxes, respectively
  - o Select "Ethernet" option under "Connection through"
  - o Click on "**Connect**" to establish connection with controller.

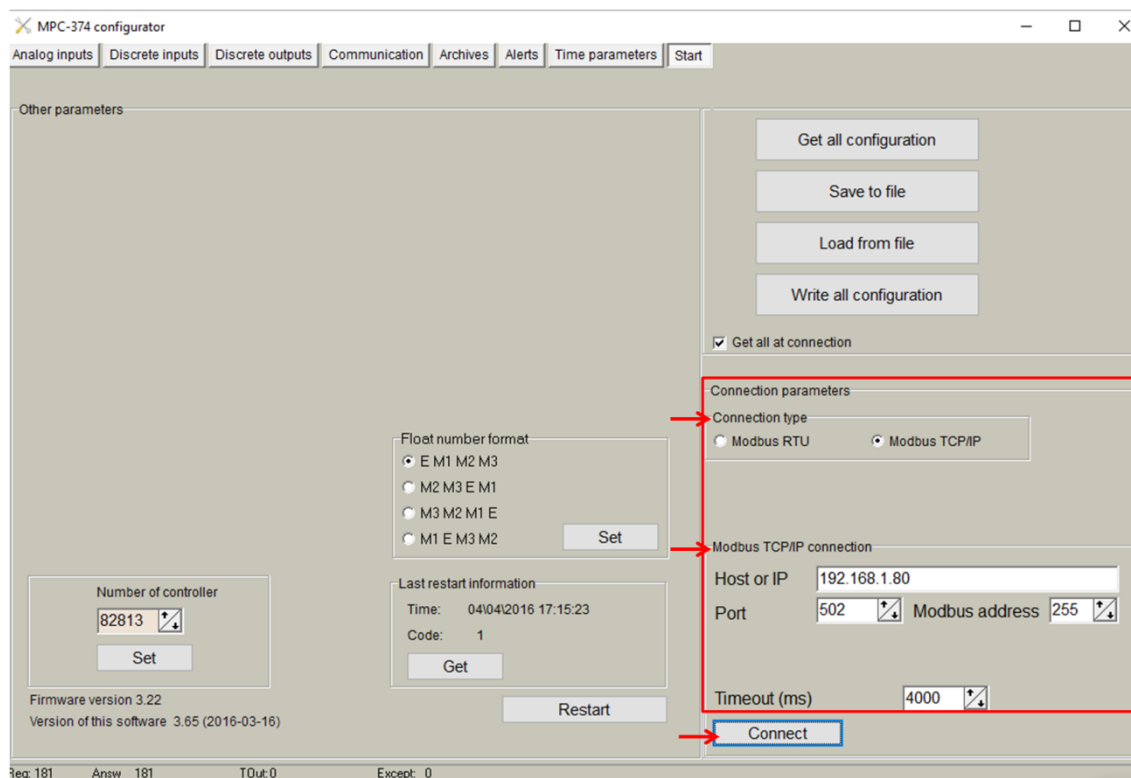


Figure 4. Ethernet connection set up



## 2.3 GPRS connection

Steps to be followed:

- Check GPRS antenna is properly connected to the device.



Figure 5. GPRS antenna connection

- Open MPC-374 Configuration Tool
- Set up "**Connection Parameters**" frame
  - o Select "ModBus TCP/IP" option under "Connection type"
  - o Add controller IP address and TCP port in "Host or IP" and "Port" text boxes, respectively
  - o Select "HOST or IP" option under "Connection through"
  - o Click on "**Connect**" to establish connection with controller.

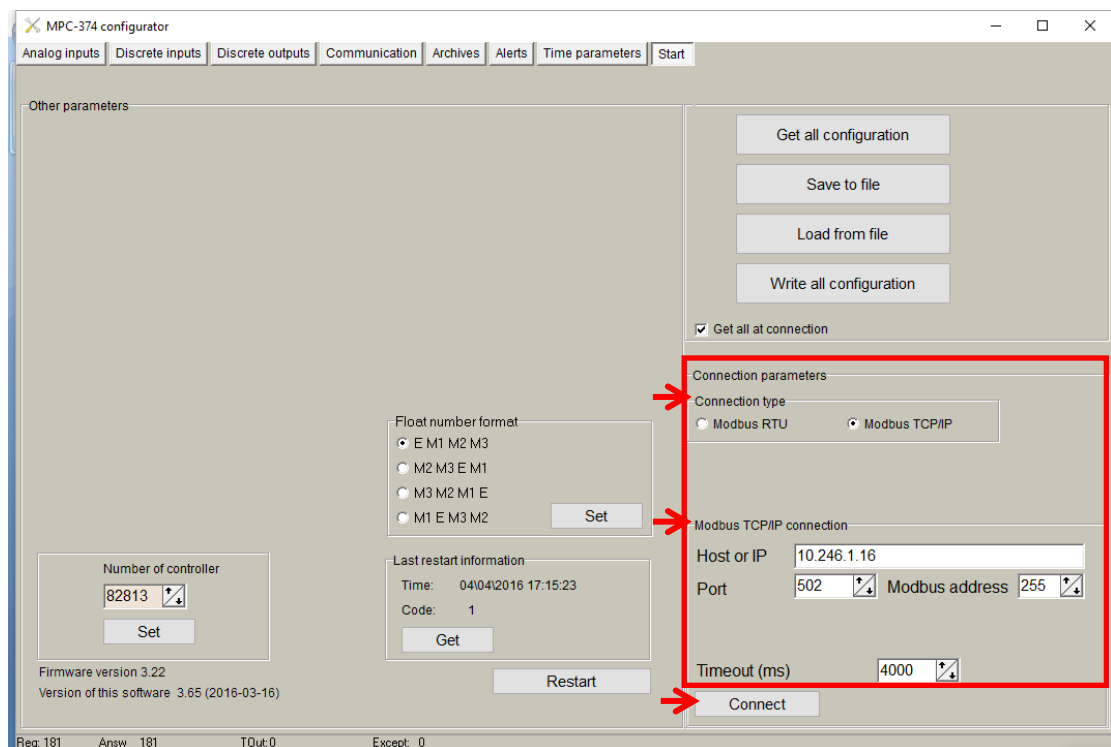


Figure 6. GPRS connection set up

## 2.4 MPC-374 connection diagrams

In the following Figure, the most typical connection schemes of MPC-374 with PC are shown. The Configuration Tool described in this document or any other Modbus compliant software can establish a communication link making use of **Modbus RTU** and/or **Modbus TCP** protocols.

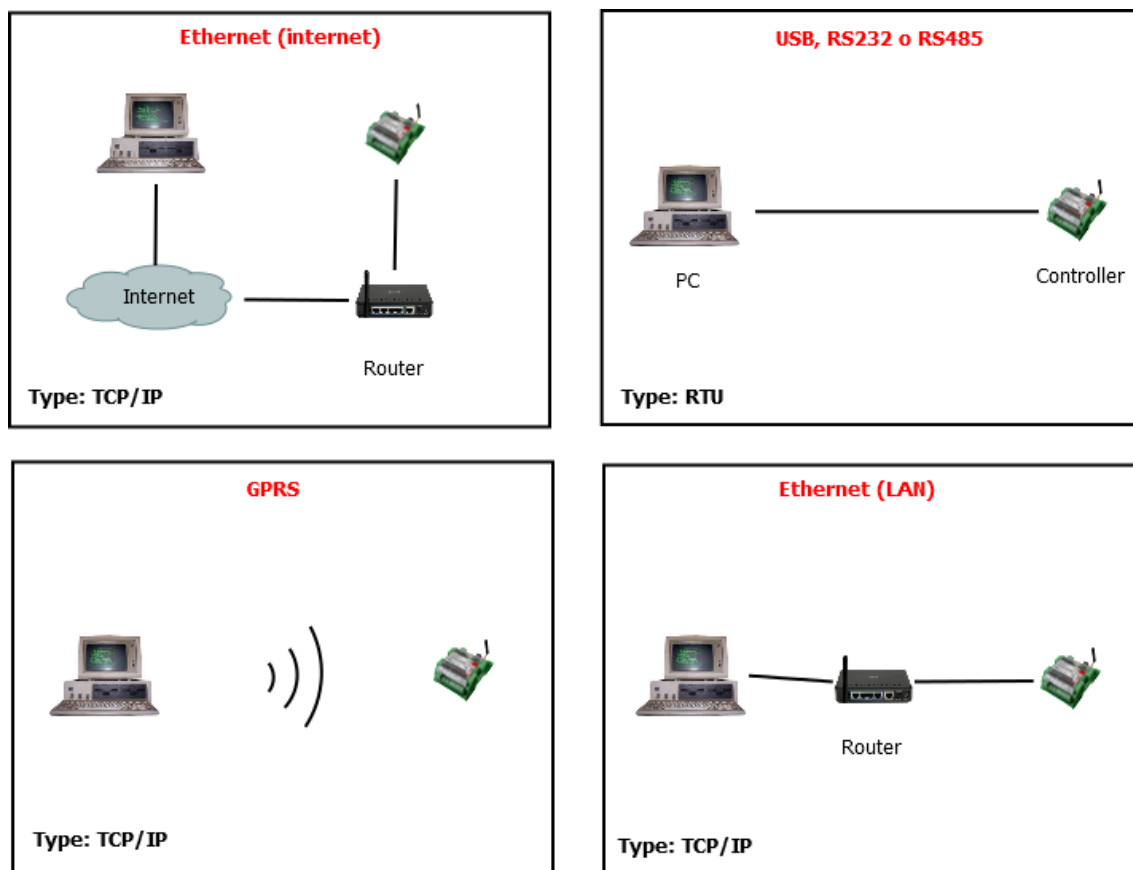


Figure 7. MPC-374 typical connection diagrams

### 3 "Start" tab

#### 3.1 MPC-374 Basic Information

Once user has established communication with MPC-374, basic information such as, "Serial number", "Last restart", "Firmware version", "Software version", etc. is shown in the left frame

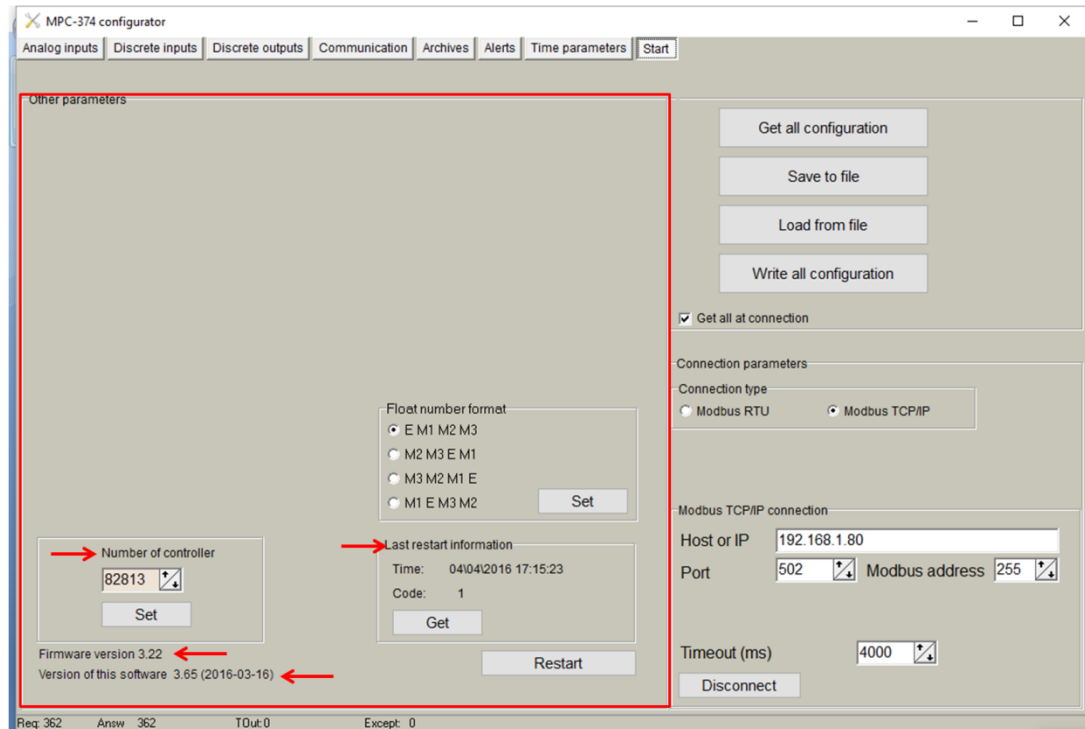


Figure 8. "Start" tab. MPC-374 basic information

## 3.2 Configuration files

This feature enables user to save and load configuration files so that programming a number of controllers with the same configuration becomes an easy process. Steps to be followed are:

1. Set up all the configuration parameters making use of MPC-374 Configuration Tool.
2. Then, under “Start” tab, click on **"Save configuration"** button. A dialog will be shown requesting user to select folder destination.
3. Once the file has been stored, connect a new controller to the PC and then click on **"Open configuration"** and select the file previously stored.
4. Then, click on **"Write configuration"** button to load such configuration into the new controller.
5. A restart will be needed so that controller can start using the loaded configuration.
6. Repeat from step 3 with all the controllers that need the same configuration.

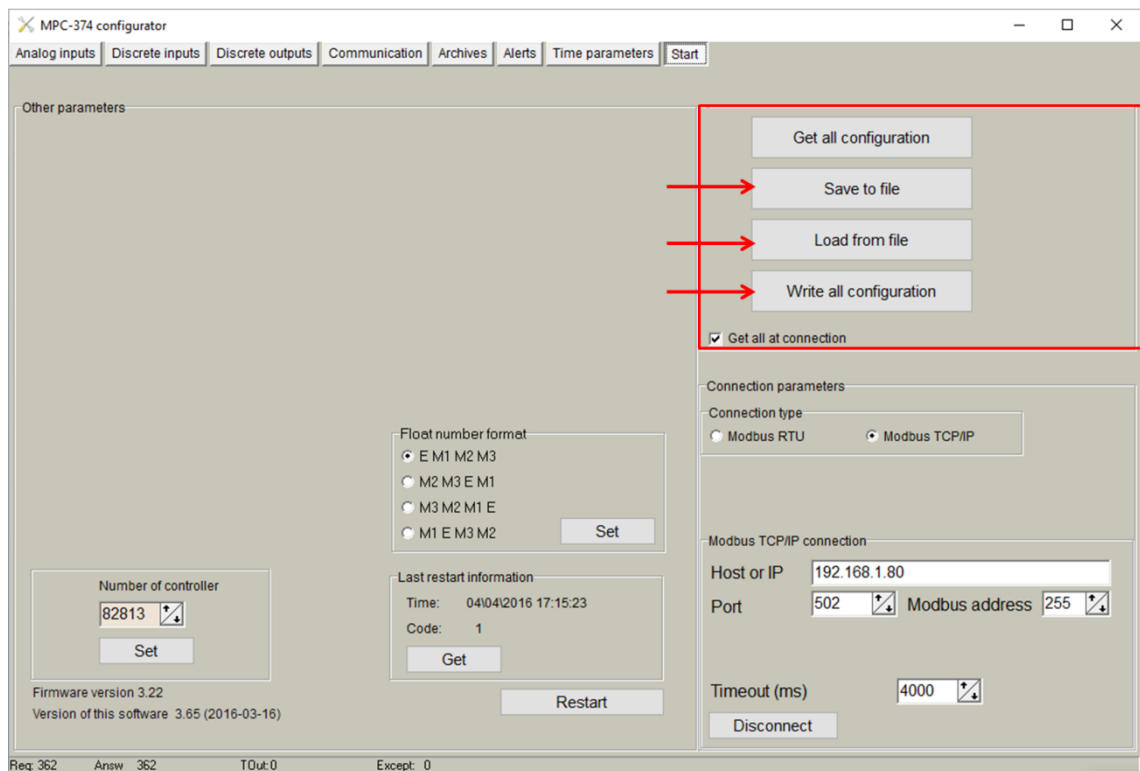


Figure 9. “Start” tab. Configuration files management

### 3.3 Status indicators

Several status indicators are shown in the MPC-374 configuration tool in order to inform user about current performance of Modbus communication:

1. Req: number of Modbus requests performed.
2. Answ: Number of Modbus answers received.
3. Tout: number of Modbus requests not answered (time outs raised).
4. Except: number of Modbus errors.

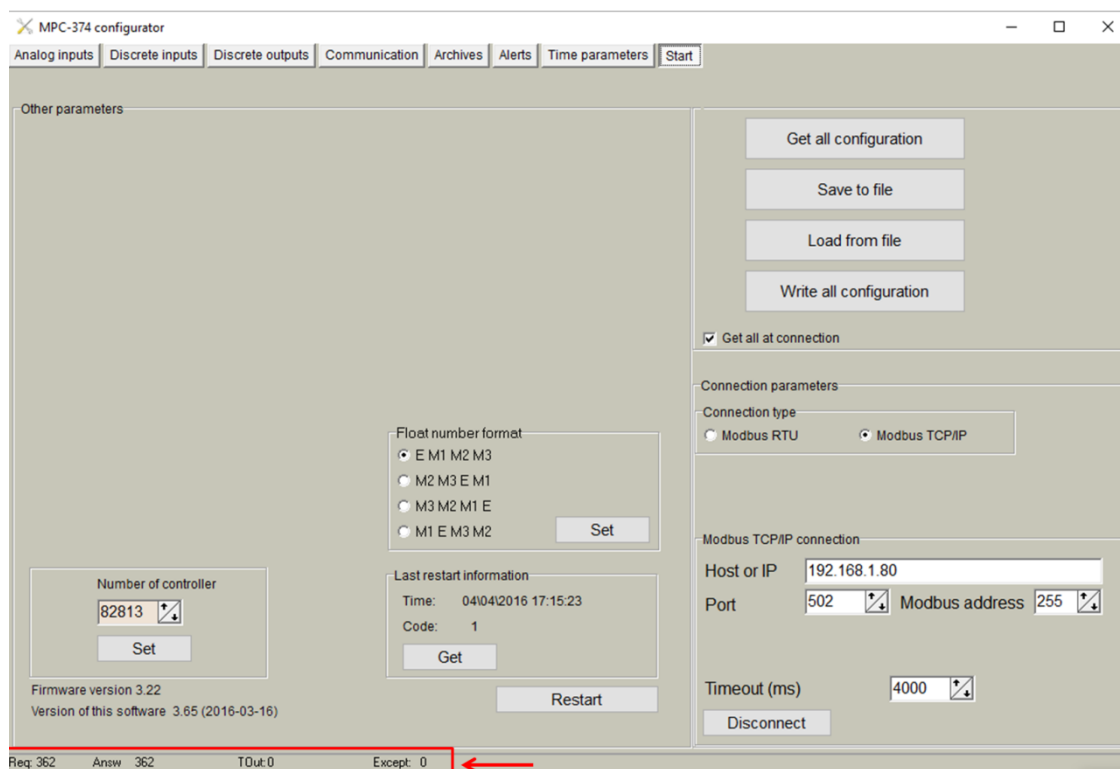


Figure 10. “Start” tab. Status indicators

## 4 “Analog inputs” tab

Up to 8 analog inputs can be included in the MPC-374, depending on its reference. In the case of the MPC-374 all analog inputs are 'Current' type, which means that just current-output sensors can be attached to the MPC-374.

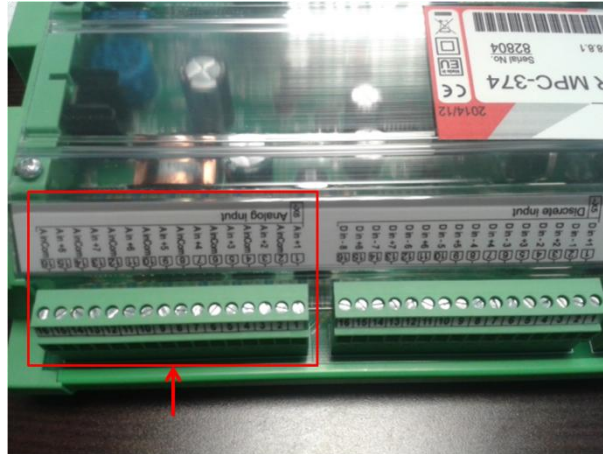


Figure 11. Analog inputs Port

### 4.1 Current Output Sensors

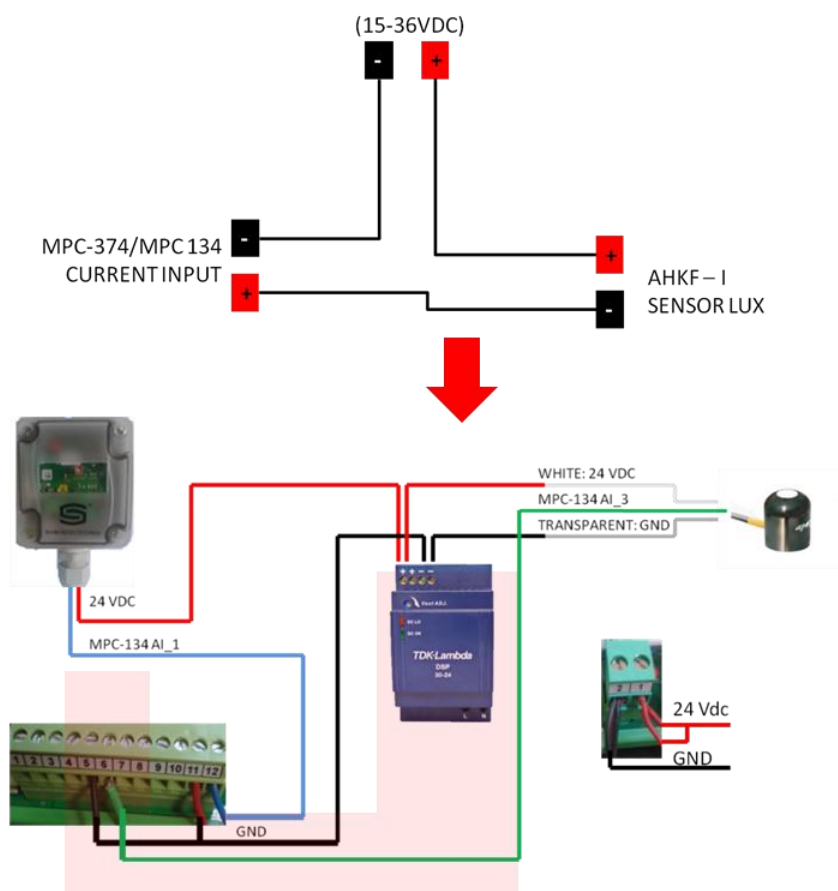
There exists two kind of current sensor types which can be attached to the MPC-374.

- Passive sensors. This kind of sensors don't need to be powered for generating the Current output. In this case, the connection to the MPC-374 must be in 'Passive mode' (by connecting a standard power source in between the sensor and the analog input).
  - Example: 'Light Sensor' > [Photasgard AHFK-I](#).
    - Power supply: 15-36 Vdc
    - Output : 4-20 mA (0-20000lux)



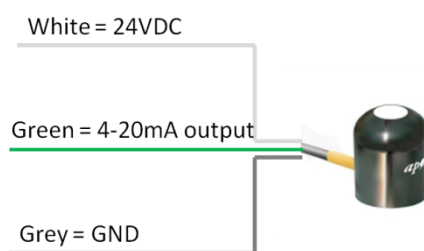
Figure 12. Photasgard AHFK-I

Due to this sensor is a 'Passive Sensor' the connection with the MPC-374 will be the following:



**Figure 13. AHFK- MPC374 wiring example**

- Active sensors. This kind of sensors needs to be powered for generating its current output.
  - Example: 'PAR Light Sensor'> APOGEE\_SQI225
    - Power supply: 5-24 Vdc
    - Output : 4-20 mA ( $0.5 \text{ umol m}^{-2} \text{ s}^{-1} \times \text{mV}$ )



**Figure 14. Apogee SQI225**

In this case, the connection to the MPC-374 can be done directly connecting the 'sensor output' to the 'analog input(+)' and the 'sensor GND' to the 'analog input(-)'

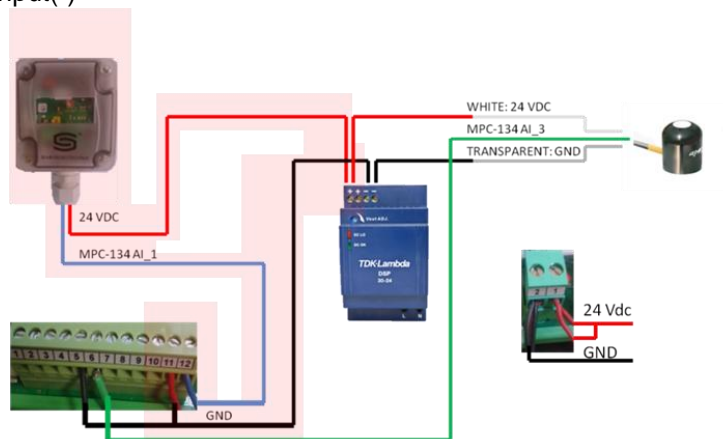


Figure 15. APOGEE-MPC374 wiring example

## 4.2 Analog Inputs configuration

In order to configure the Analog inputs go to *Analog inputs>Configuration* and make the following steps:

- Enable the Analog inputs.
- Select the sensor's current type "Range"
- Fill the Low and high conversion values for the sensor in "MIN. value" y "MAX value"
- In case of "Alert" fill the Alert limits in "Low alarm limit" y "High alarm limit" and also configure which kind of event will be associated to the alert event. y in "Alarm Case".
- Once everything is configured, click in 'SET'

In the "Current value" column it is possible to check the current analog input value once the configuration has been done, including its conversion according to the scale range configured. In case that it is not needed to load a conversion in the Analog input channel, just fill in "MIN. value= 4mA" y "MAX value = 20mA" and the output will be shown in mA.

MPC-374 configurator

Analog inputs | Discrete inputs | Discrete outputs | Communication | Archives | Alerts | Time parameters | Start

Configuration | Calibration | Test

Analog inputs

AIn	Enabled	Range	Min. value	Max. value	Filter averaging duration	Alarm filt. time (sec.)	Lower alarm limit	High alarm limit	Alarm case	Current value
1	<input checked="" type="checkbox"/>	4-20 mA	0.000	20000.000	4	3	10.000	18000.000	Out of range Out & return	226,5
2	<input checked="" type="checkbox"/>	4-20 mA	0.000	1000.000	1	3	10.000	980.000	Out of range Out & return	10,9
3	<input checked="" type="checkbox"/>	4-20 mA	0.000	20.000	1	3	0.000	20.000	Out of range Out & return	-1000000.000
4	<input checked="" type="checkbox"/>	4-20 mA	0.000	20.000	1	3	0.000	20.000	Out of range Out & return	-1000000.000
5	<input type="checkbox"/>	4-20 mA	0.000	20.000	1	3	0.000	20.000	Out of range Out & return	-1000000.000
6	<input type="checkbox"/>	4-20 mA	0.000	20.000	1	3	0.000	20.000	Out of range Out & return	0.000
7	<input type="checkbox"/>	4-20 mA	0.000	20.000	1	3	0.000	20.000	Out of range Out & return	0.000
8	<input type="checkbox"/>	4-20 mA	0.000	20.000	1	3	0.000	20.000	Out of range Out & return	0.000

Set Get

Figure 16. Analog input configuration.



## 5 “Discrete inputs” tab

Discrete inputs can be configured within this tab. They are shown organized in columns and configuration parameters are shown in rows as follows:

- "Current State": Graphical representation of current status (open/closed).
- "Impulse quantity": Number of pulses counted from last reset.
- "Value": Calculated from multiplying number of pulses counted (Impulse quantity) and user-defined multiplier (Impulse multiplier).
- "Enabled": user enable/disable discrete inputs to be used. In order to save memory, it is recommended to only enable inputs in use.
- "Mode":
  - "Discrete Input": input works as a standard discrete input detecting its current state (1 / 0)
  - "Impulse counter": input is used as a standard pulse counter, storing the amount of pulses raised.
  - "Alarm signal": input is used as an alarm input.
- "Alarm state": If "Alarm signal" option is selected, controller will raised an alarm when one of the following state is reached: "Open", "Close", "Both cases".
- "Enable archiving": If enabled, events produced by discrete inputs are stored in the datalogging archive.
- "Debounce time": only pulses received within one or more “Debounce time” intervals are registered.
- "Impulse multiplier": user-defined multiplier. Values can be set from 1 to 1000.
- "Set initial amount of pulses": initial value for pulse counter can be set here.

NOTE: After configuration is completed, click on "Set" button in order to save changes.

MPC-374 configurator

Analogue inputs | **Discrete inputs** | Discrete outputs | Communication | Archives | Alerts | Time parameters | Start

	Chan. 1	Chan. 2	Chan. 3	Chan. 4	Chan. 5	Chan. 6	Chan. 7	Chan. 8
Current state								
Impulse quantity	15	0	0	0	0	0		
Value	15,000	0,000	0,000	0,000	0,000	0,000		
Enabled	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Mode	<input checked="" type="radio"/> Discrete inp. <input type="radio"/> Impulse count <input type="radio"/> Alarm signal	<input checked="" type="radio"/> Discrete inp. <input type="radio"/> Impulse count <input type="radio"/> Alarm signal	<input checked="" type="radio"/> Discrete inp. <input type="radio"/> Impulse count <input type="radio"/> Alarm signal	<input checked="" type="radio"/> Discrete inp. <input type="radio"/> Impulse count <input type="radio"/> Alarm signal	<input checked="" type="radio"/> Discrete inp. <input type="radio"/> Impulse count <input type="radio"/> Alarm signal	<input checked="" type="radio"/> Discrete inp. <input type="radio"/> Impulse count <input type="radio"/> Alarm signal	<input checked="" type="radio"/> Discrete inp. <input type="radio"/> Impulse count <input type="radio"/> Alarm signal	<input checked="" type="radio"/> Discrete inp. <input type="radio"/> Impulse count <input type="radio"/> Alarm signal
Alarm state	<input checked="" type="radio"/> Open <input type="radio"/> Close <input type="radio"/> Both cases	<input checked="" type="radio"/> Open <input type="radio"/> Close <input type="radio"/> Both cases	<input checked="" type="radio"/> Open <input type="radio"/> Close <input type="radio"/> Both cases	<input checked="" type="radio"/> Open <input type="radio"/> Close <input type="radio"/> Both cases	<input checked="" type="radio"/> Open <input type="radio"/> Close <input type="radio"/> Both cases	<input checked="" type="radio"/> Open <input type="radio"/> Close <input type="radio"/> Both cases	<input checked="" type="radio"/> Open <input type="radio"/> Close <input type="radio"/> Both cases	<input checked="" type="radio"/> Open <input type="radio"/> Close <input type="radio"/> Both cases
Enable archiving	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
State to archive	<input checked="" type="radio"/> Open <input type="radio"/> Close <input type="radio"/> Both cases	<input checked="" type="radio"/> Open <input type="radio"/> Close <input type="radio"/> Both cases	<input checked="" type="radio"/> Open <input type="radio"/> Close <input type="radio"/> Both cases	<input checked="" type="radio"/> Open <input type="radio"/> Close <input type="radio"/> Both cases	<input checked="" type="radio"/> Open <input type="radio"/> Close <input type="radio"/> Both cases	<input checked="" type="radio"/> Open <input type="radio"/> Close <input type="radio"/> Both cases	<input checked="" type="radio"/> Open <input type="radio"/> Close <input type="radio"/> Both cases	<input checked="" type="radio"/> Open <input type="radio"/> Close <input type="radio"/> Both cases
Debounce time (ms)	50	50	50	50	257	257	257	257
Impulse multiplier	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000

Set Get

Set the initial amount of impulses

	D In 1	D In 2	D In 3	D In 4	D In 5	D In 6	D In 7	D In 8
Impulse quantity								
	Set	Set	Set	Set	Set	Set	Set	Set

Clear

Req: 796 Answ: 788 TOut: 0 Except: 0

Figure 11. “Discrete inputs” tab

## 6 “Discrete Outputs” tab

The MPC-374 has eight discrete relay outputs that can perform in 2 different working modes:

- Manual controls. It is possible to enable/disable the relay status by using the Manual controls that appear in *Discrete outputs>Configuration*.
- Manual test. It is possible to test manually each DO by sending a configurable pulse. It is possible to configure the 'Time of impulse (milliseconds)' in the 'programmed controls panel'

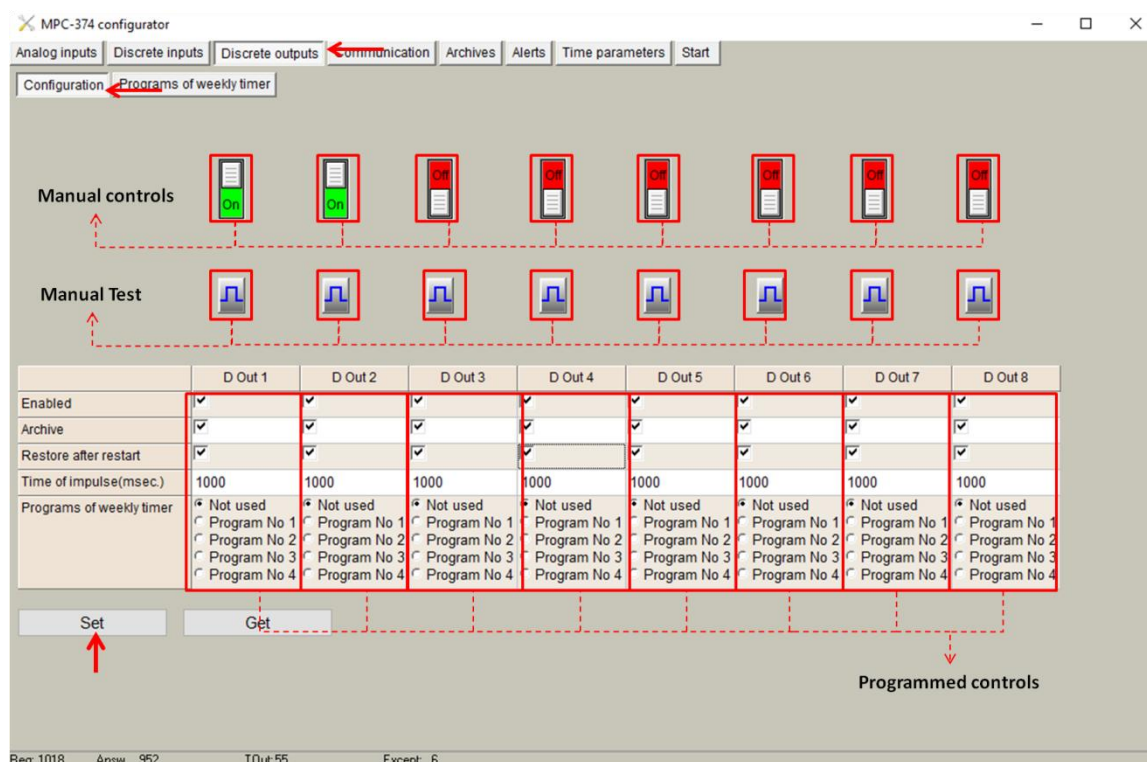


Figure 12. “Discrete outputs” programmer

- Also it is possible to manage the state of the relays by using the MPC-374 register list with Modbus function '3' for reading or '16' for writing:

Modbus Register list - Values										
Description	DO_1	DO_2	DO_3	DO_4	DO_5	DO_6	DO_7	DO_8	Data format	Type
Discrete Output Use (0-disabled, 1 – enabled)	1	1	1	1	1	1	1	1	Char	R/W
Discrete OutputCurrent State (0-OFF, 1 – ON)	1	0	0	0	0	0	0	0	Char	R/W

(ALL ENABLED)

(DO1=ON, REST=OFF)

Figure19. Discrete Outputs MPC-374's Register list

- Programmed: In case of the 'program mode', it is possible to program 4 different working calendars related to the discrete outputs. For this purpose, go to 'Discrete outputs> programs of weekly timer' and load them to the different discrete outputs:

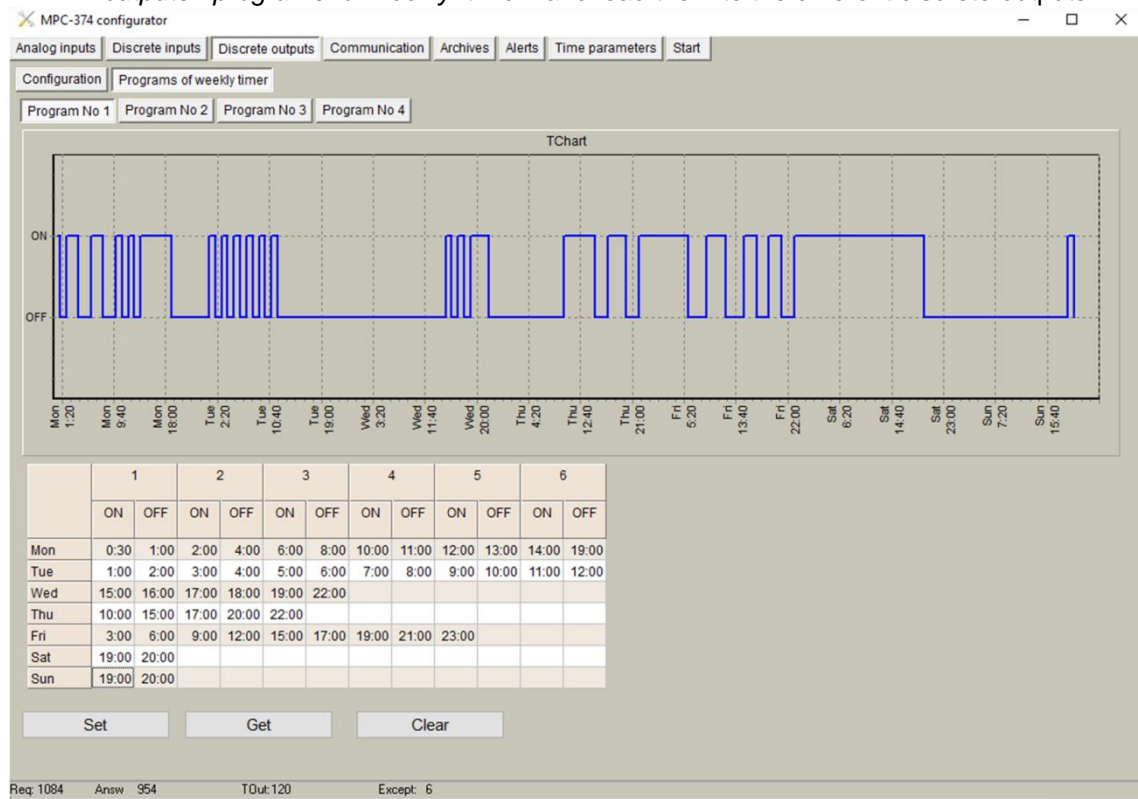


Figure 20. "Discrete outputs" Program No1

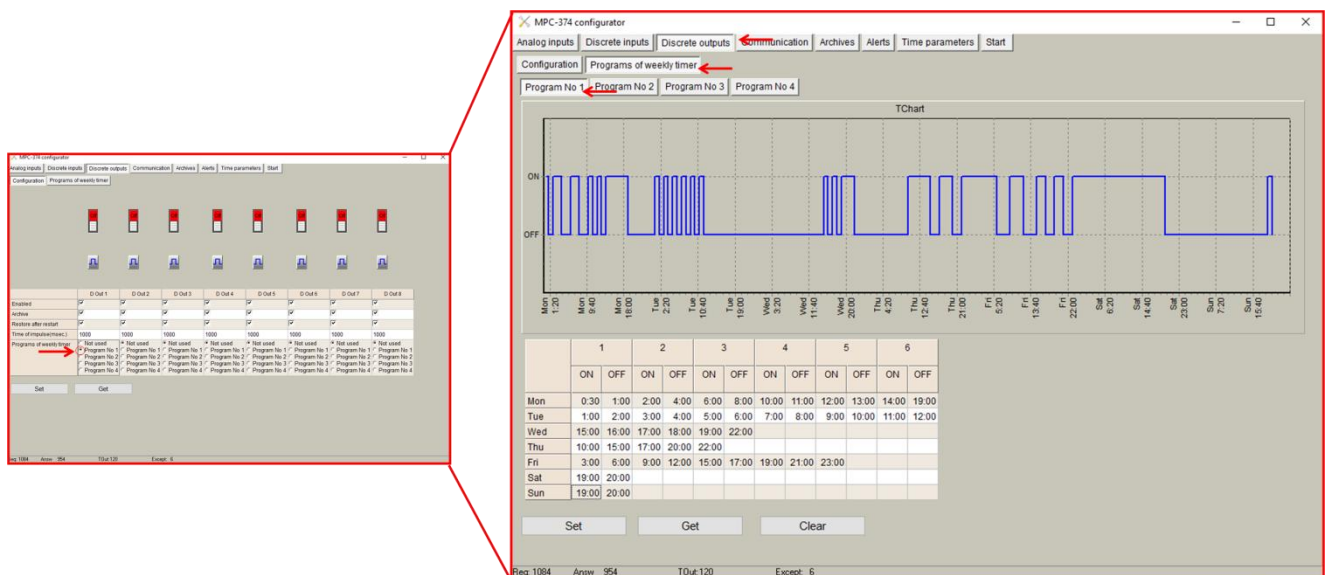


Figure 21. "Discrete outputs" Program No1 . Example

## 7 “Communication” tab

### 7.1 Communication > Ethernet

Ethernet interface parameters can be configured within this tab:

- "MAC number": Media Access Control address
- "IP address"
- "Gateway IP"
- "Mask"

NOTE: After configuration is completed, click on "**Set**" button in order to save changes.

MPC-374 configurator

Communication

Ethernet

MAC number: 00004C01437D Set unique MAC

IP address: 192.168.1.80

Gateways IP: 192.168.1.1

Mask: 255.255.255.0

Incoming IP filter

Filter enabled: ☐

IP 1: 28.32.255.255

IP 2: 255.255.255.255

IP 3: 255.255.255.255

IP 4: 255.255.255.255

IP 5: 255.255.255.255

Ping data

Enabled: ☐

Send to address: 255.255.255.255

Time interval(sec.): 60

Time before restart if no packets receive: 600

Set Get

Req: 532 Answ: 528 TOut: 2 Except: 0

Figure 13. “Ethernet” configuration tab

## 7.2 Communication > GPRS

As previously described, MPC-374 can be configured through a GPRS link. In order to do so, user must enable GPRS connection as shown below. Then, click on **"Set"** button in order to save changes.

The screenshot shows the MPC-374 configurator interface. The 'GPRS' tab is selected, and the 'Enabled' checkbox is checked. The 'GPRS-GSM mode' section has 'GPRS' selected. The 'Signal level measurement' section has 'Periodically' selected. The 'Set' button is highlighted with a red arrow. The 'GPRS' sub-tab is active, showing 'DDNS setting' and 'Incoming IP filter' sections. The 'APN' is set to 'movistar'. The 'GPRS login' section shows 'Enabled' checked, 'User name' as 'guest', and 'Password' as 'guest'. The 'Status of GPRS connection' section shows 'Status' as 'Assigned IP: 0.0', 'Connecting to GPRS time: 0.0', 'Disconnecting from GPRS time: 0.0', and 'Connections' as 0. A table with 10 rows for 'Socket', 'Port', and 'IP' is also visible.

**Figure 14. "GPRS" Tab. Enabling communication**

Within this tab, different services can be configured or checked.

1. GPRS connection parameters:
  - "APN": access point name provided by Internet Service Provider (ISP).

If needed by ISP, credentials can also be configured here:

- "Username": provided by ISP.
- "Password": provided by ISP.

NOTE: After configuration is completed, click on **"Set"** button in order to save changes.

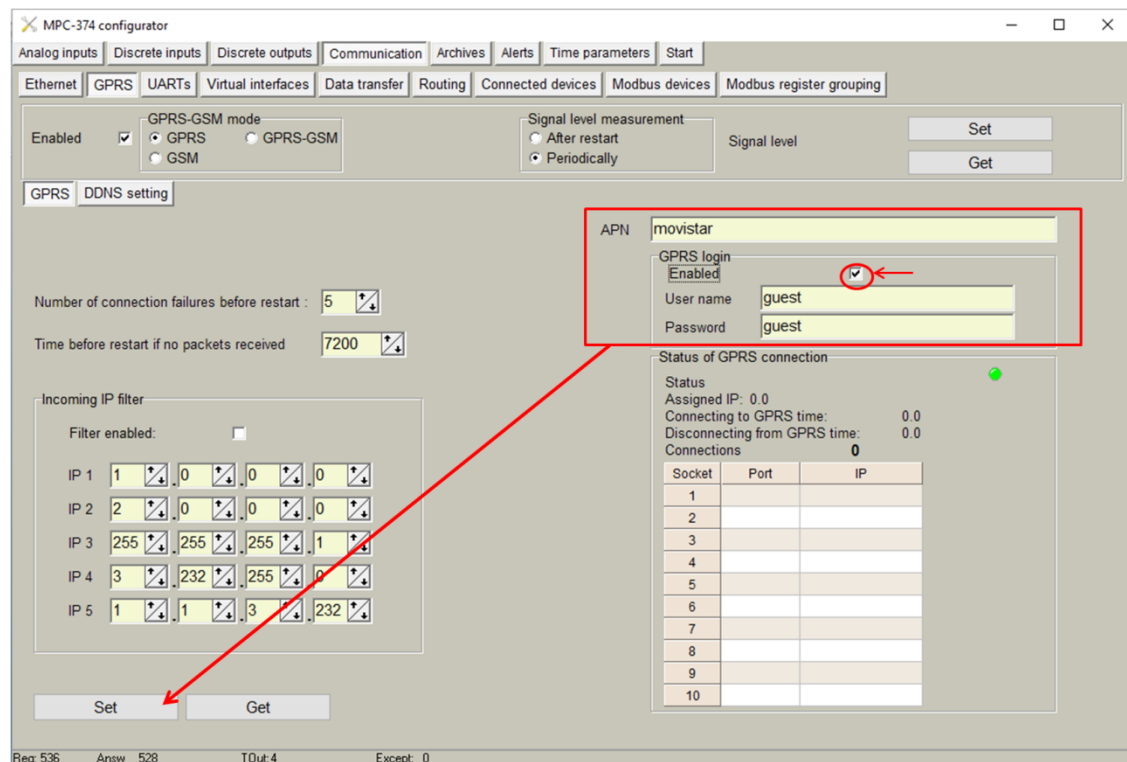


Figure 15. “GPRS” tab. Internet Service Provider access data configuration

2. Connection status. Once GPRS connection is established, status will be shown as below:

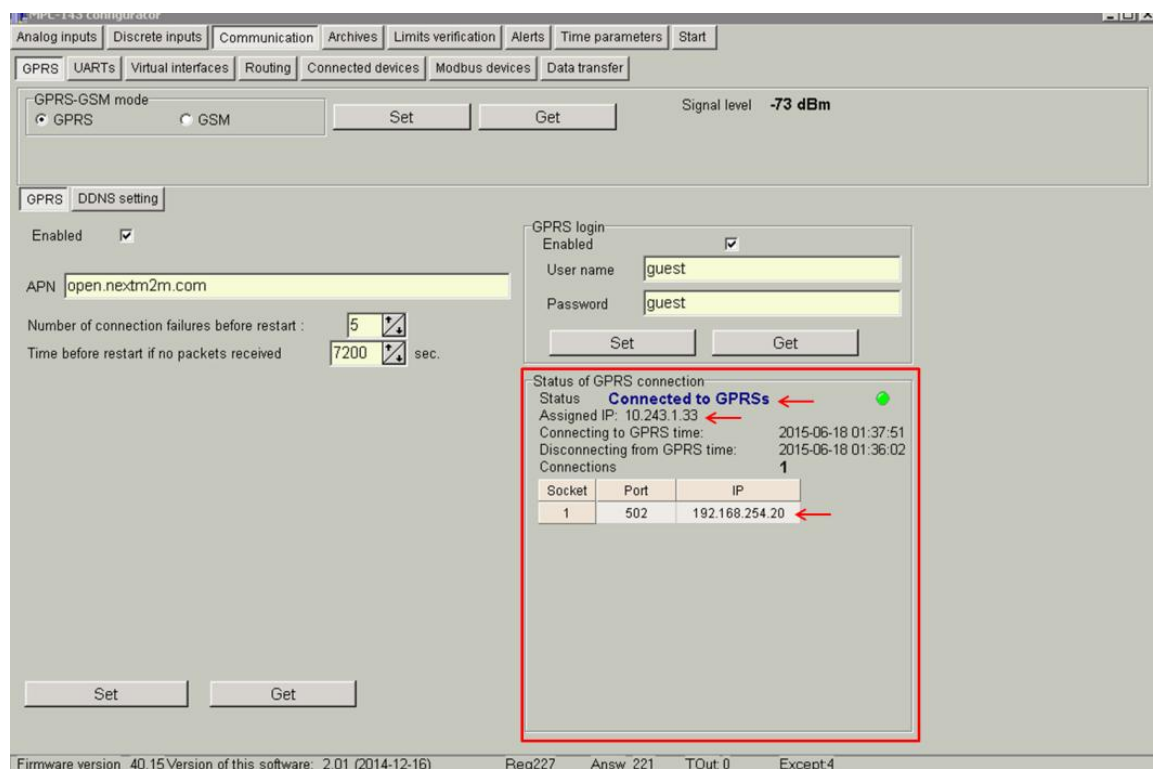


Figure 16. “GPRS” tab. Connection status



3. DynDNS account configuration: in order to avoid the need to have a fixed IP address to be able to remotely access the device, it is possible to set up a DynDNS account under the "DDNS setting". Steps to be followed are:
- Select "Enable"
  - Fill in information according to your DynDNS account
  - Click on "Load from file" and select "dyndns.org\_Register\_String.txt" which contains the connection string to be sent to DynDNS servers.

The screenshot displays the 'GPRS' configuration tab with the 'DDNS setting' sub-tab selected. The 'Enabled' checkbox is checked. The 'Server URL' is set to 'members.dyndns.com', 'Client Host Name' is 'YourDomainName.dyndns.org', 'User Name' is 'YourUserName', and 'User Password' is 'YourPass'. The 'Load from file' button is highlighted with a red arrow. The 'Set' button at the bottom left is also highlighted with a red arrow. The 'Registration string' field contains the following text:

```
Registration string
GET /nic/update?hostname=advanticedemo.dyndns.org&myip=<ipaddress>
Host: members.dyndns.org
Authorization: Basic <authorization>
User-Agent: AdvanticsSYS - MPC333 Controller - 2.46
```

**Figure 17. "GPRS" tab. DynDNS configuration**

NOTE: After configuration is completed, click on "Set" button in order to save changes.

NOTE: It is recommended to disable GPRS connection if not used.

## 7.3 Communication > UARTs

Each UART can be configured individually. It is recommended to check peripheral devices UART constraints before setting up parameters in the controller configuration tool.

NOTE: All devices connected to the same MPC-374 UART must have the same communication parameters.

- "Bode": Transmission rate (bauds per second).
- "Parity": Communication Parity.
- "Data bits": Number of data bits within the communication packet.
- "Stop bits": Number of stop bits within the communication packet.
- "Packetization":
  - "time": One packet time has been reached, it is considered that the packet has been properly sent.
  - "Symbol". Every time a "packet symbol" is received, it is considered that the packet has been properly sent.
  - "Length". Once the "packet byte count" has been reached, it is considered that the packet has been properly sent.
- "Packet time": Transmission duration.
- "Packet symbol": Symbol which defines packet ending.
- "Packet byte count": Length which defines packet size
- "Mode": Full duplex or half duplex
- "Destination of DTR": bit which defines the "Data ready" state has been reached. It must be configured following peripheral devices connected to MPC-374 UARTs vendor information. By default, if this information is not available, it is recommended to select "OFF when send" option.

NOTE: After configuration is completed, click on **"Set"** button in order to save changes.

The screenshot shows the 'MPC-374 configurator' window with the 'UARTs' tab selected. The window has a top menu bar with options: Analog inputs, Discrete inputs, Discrete outputs, Communication, Archives, Alerts, Time parameters, and Start. Below this is a sub-menu bar with: Ethernet, GPRS, UARTs (selected), Virtual interfaces, Data transfer, Routing, Connected devices, Modbus devices, and Modbus register grouping. The main area is a table with 8 columns for UART 1 through UART 8 and rows for various parameters. A red box highlights the 'UARTs' tab and the table. At the bottom, there are 'Set' and 'Get' buttons, with a red arrow pointing to the 'Set' button. The status bar at the very bottom shows 'req: 536', 'Answ: 528', 'TOut: 4', and 'Except: 0'.

	UART 1	UART 2	UART 3	UART 4	UART 5	UART 6	UART 7	UART 8
Bode	<input type="radio"/> 300 <input type="radio"/> 600 <input type="radio"/> 1200 <input type="radio"/> 2400 <input type="radio"/> 4800 <input checked="" type="radio"/> 9600 <input type="radio"/> 19200 <input type="radio"/> 38400 <input type="radio"/> 57600	<input type="radio"/> 300 <input type="radio"/> 600 <input type="radio"/> 1200 <input type="radio"/> 2400 <input type="radio"/> 4800 <input checked="" type="radio"/> 9600 <input type="radio"/> 19200 <input type="radio"/> 38400 <input type="radio"/> 57600	<input type="radio"/> 300 <input type="radio"/> 600 <input type="radio"/> 1200 <input type="radio"/> 2400 <input type="radio"/> 4800 <input checked="" type="radio"/> 9600 <input type="radio"/> 19200 <input type="radio"/> 38400 <input type="radio"/> 57600	<input type="radio"/> 300 <input type="radio"/> 600 <input type="radio"/> 1200 <input type="radio"/> 2400 <input type="radio"/> 4800 <input checked="" type="radio"/> 9600 <input type="radio"/> 19200 <input type="radio"/> 38400 <input type="radio"/> 57600	<input type="radio"/> 300 <input type="radio"/> 600 <input type="radio"/> 1200 <input type="radio"/> 2400 <input type="radio"/> 4800 <input checked="" type="radio"/> 9600 <input type="radio"/> 19200 <input type="radio"/> 38400 <input type="radio"/> 57600	<input type="radio"/> 300 <input type="radio"/> 600 <input type="radio"/> 1200 <input type="radio"/> 2400 <input type="radio"/> 4800 <input checked="" type="radio"/> 9600 <input type="radio"/> 19200 <input type="radio"/> 38400 <input type="radio"/> 57600	<input type="radio"/> 300 <input type="radio"/> 600 <input type="radio"/> 1200 <input type="radio"/> 2400 <input type="radio"/> 4800 <input checked="" type="radio"/> 9600 <input type="radio"/> 19200 <input type="radio"/> 38400 <input type="radio"/> 57600	<input type="radio"/> 300 <input type="radio"/> 600 <input type="radio"/> 1200 <input type="radio"/> 2400 <input type="radio"/> 4800 <input checked="" type="radio"/> 9600 <input type="radio"/> 19200 <input type="radio"/> 38400 <input type="radio"/> 57600
Parity	<input type="radio"/> Even <input type="radio"/> Odd <input type="radio"/> Mark <input type="radio"/> Space <input checked="" type="radio"/> None	<input type="radio"/> Even <input type="radio"/> Odd <input type="radio"/> Mark <input type="radio"/> Space <input checked="" type="radio"/> None	<input type="radio"/> Even <input type="radio"/> Odd <input type="radio"/> Mark <input type="radio"/> Space <input checked="" type="radio"/> None	<input type="radio"/> Even <input type="radio"/> Odd <input type="radio"/> Mark <input type="radio"/> Space <input checked="" type="radio"/> None	<input type="radio"/> Even <input type="radio"/> Odd <input type="radio"/> Mark <input type="radio"/> Space <input checked="" type="radio"/> None	<input type="radio"/> Even <input type="radio"/> Odd <input type="radio"/> Mark <input type="radio"/> Space <input checked="" type="radio"/> None	<input type="radio"/> Even <input type="radio"/> Odd <input type="radio"/> Mark <input type="radio"/> Space <input checked="" type="radio"/> None	<input type="radio"/> Even <input type="radio"/> Odd <input type="radio"/> Mark <input type="radio"/> Space <input checked="" type="radio"/> None
Data bits	<input type="radio"/> 5 <input type="radio"/> 6 <input type="radio"/> 7 <input checked="" type="radio"/> 8	<input type="radio"/> 5 <input type="radio"/> 6 <input type="radio"/> 7 <input checked="" type="radio"/> 8	<input type="radio"/> 5 <input type="radio"/> 6 <input type="radio"/> 7 <input checked="" type="radio"/> 8	<input type="radio"/> 5 <input type="radio"/> 6 <input type="radio"/> 7 <input checked="" type="radio"/> 8	<input type="radio"/> 5 <input type="radio"/> 6 <input type="radio"/> 7 <input checked="" type="radio"/> 8	<input type="radio"/> 5 <input type="radio"/> 6 <input type="radio"/> 7 <input checked="" type="radio"/> 8	<input type="radio"/> 5 <input type="radio"/> 6 <input type="radio"/> 7 <input checked="" type="radio"/> 8	<input type="radio"/> 5 <input type="radio"/> 6 <input type="radio"/> 7 <input checked="" type="radio"/> 8
Stop bits	<input checked="" type="radio"/> 1 <input type="radio"/> 1.5 <input type="radio"/> 2	<input checked="" type="radio"/> 1 <input type="radio"/> 1.5 <input type="radio"/> 2	<input checked="" type="radio"/> 1 <input type="radio"/> 1.5 <input type="radio"/> 2	<input checked="" type="radio"/> 1 <input type="radio"/> 1.5 <input type="radio"/> 2	<input checked="" type="radio"/> 1 <input type="radio"/> 1.5 <input type="radio"/> 2	<input checked="" type="radio"/> 1 <input type="radio"/> 1.5 <input type="radio"/> 2	<input checked="" type="radio"/> 1 <input type="radio"/> 1.5 <input type="radio"/> 2	<input checked="" type="radio"/> 1 <input type="radio"/> 1.5 <input type="radio"/> 2
Packetization	<input checked="" type="radio"/> Time <input type="radio"/> Symbol <input type="radio"/> Length <input type="radio"/> >1b+time	<input checked="" type="radio"/> Time <input type="radio"/> Symbol <input type="radio"/> Length <input type="radio"/> >1b+time	<input checked="" type="radio"/> Time <input type="radio"/> Symbol <input type="radio"/> Length <input type="radio"/> >1b+time	<input checked="" type="radio"/> Time <input type="radio"/> Symbol <input type="radio"/> Length <input type="radio"/> >1b+time	<input checked="" type="radio"/> Time <input type="radio"/> Symbol <input type="radio"/> Length <input type="radio"/> >1b+time	<input checked="" type="radio"/> Time <input type="radio"/> Symbol <input type="radio"/> Length <input type="radio"/> >1b+time	<input checked="" type="radio"/> Time <input type="radio"/> Symbol <input type="radio"/> Length <input type="radio"/> >1b+time	<input checked="" type="radio"/> Time <input type="radio"/> Symbol <input type="radio"/> Length <input type="radio"/> >1b+time
Packet time (msec.)	100	100	80	100	200	150	120	80
Packet symbol(Hex)	00	00	00	00	FF	0D	FF	0D
Packet byte count	10	1	10	10	10	10	10	1
Mode	<input type="radio"/> Full duplex <input checked="" type="radio"/> Half duplex	<input type="radio"/> Full duplex <input checked="" type="radio"/> Half duplex	<input type="radio"/> Full duplex <input checked="" type="radio"/> Half duplex	<input type="radio"/> Full duplex <input checked="" type="radio"/> Half duplex	<input type="radio"/> Full duplex <input checked="" type="radio"/> Half duplex	<input type="radio"/> Full duplex <input checked="" type="radio"/> Half duplex	<input type="radio"/> Full duplex <input checked="" type="radio"/> Half duplex	<input type="radio"/> Full duplex <input checked="" type="radio"/> Half duplex
Destination of DTR	<input type="radio"/> Always OFF <input type="radio"/> Always ON <input checked="" type="radio"/> OFF when send <input type="radio"/> ON when send	<input type="radio"/> Always OFF <input type="radio"/> Always ON <input checked="" type="radio"/> OFF when send <input type="radio"/> ON when send	<input type="radio"/> Always OFF <input type="radio"/> Always ON <input checked="" type="radio"/> OFF when send <input type="radio"/> ON when send	<input type="radio"/> Always OFF <input type="radio"/> Always ON <input checked="" type="radio"/> OFF when send <input type="radio"/> ON when send	<input type="radio"/> Always OFF <input type="radio"/> Always ON <input checked="" type="radio"/> OFF when send <input type="radio"/> ON when send	<input type="radio"/> Always OFF <input type="radio"/> Always ON <input checked="" type="radio"/> OFF when send <input type="radio"/> ON when send	<input type="radio"/> Always OFF <input type="radio"/> Always ON <input checked="" type="radio"/> OFF when send <input type="radio"/> ON when send	<input type="radio"/> Always OFF <input type="radio"/> Always ON <input checked="" type="radio"/> OFF when send <input type="radio"/> ON when send

Figure 18. "UARTs" tab



## 7.4 Communication > Virtual interfaces

The MPC-374 can perform as communication gateway Modbus server/client and datalogger simultaneously. In order to set up these options, different virtual interfaces are needed as shown below

MPC-374 configurator

Analogue inputs | Discrete inputs | Discrete outputs | **Communication** | Archives | Alerts | Time parameters | Start

Ethernet | GPRS | UARTs | **Virtual interfaces** | Data transfer | Routing | Connected devices | Modbus devices | Modbus register grouping

**COM Client**

	COM 1	COM 2	COM 3	COM 4	COM 5	COM 6
Enabled	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
UART	UART 2	UART 8	UART 5	UART 4	UART 7	UART 6
Stack depth	1	1	1	1	1	1
Timeout (msec.)	2000	1000	1000	1000	1000	1000
Number of repeats	2	2	2	2	2	2

Set ← Get

**Modbus RTU Client**

	Modbus 1	Modbus 2
Enabled	<input checked="" type="checkbox"/>	<input type="checkbox"/>
UART	UART 1	UART 2
Stack depth	1	1
Timeout (msec.)	1000	1000

Set ← Get

**Modbus RTU Server**

	Modbus 1	Modbus 2
Enabled	<input type="checkbox"/>	<input type="checkbox"/>
UART	UART 1	UART 3
Address	4	4

Set ← Get

UART 1: First Modbus RTU Client  
 UART 2: First COM Client  
 UART 3: Not used  
 UART 4: Not used  
 UART 5: Not used  
 UART 6: Not used  
 UART 7: Not used  
 UART 8: GPRS modem

Check

Req: 536   Answ: 528   TOut: 4   Except: 0

**Figure 19. "Virtual interfaces" tab**

- **"COM Client"**. It is needed to configure a COM Client every time a "transparent" communication mode or M-Bus compatibility is needed. This is mandatory in order to read devices not compatible with Modbus RTU/TCP protocols. Configuration parameters are:
  - "Enable": Do not enable any COM client if not needed.
  - "UART": Select the UART number in which a COM Client will be configured.  
NOTE: Once a given UART is selected to be a COM client, it cannot be used with any other configuration such as Modbus client/server.
  - "Stack depth": number of parallel executions that can be done in parallel by the COM client.
  - "Timeout": it defines the amount of milliseconds defined as time out.
  - "Number of repeats": number of retries defined in case of transmission error.

NOTE: COM Client must always be used when peripheral devices connected to a given UART use protocols such as M-Bus, IEC-102, DLMS, etc.

NOTE: After configuration is completed, click on **"Set"** button in order to save changes.

- "Modbus RTU Client". This mode working mode must be used when the MPC-374 behaves as master in a Modbus RTU network (it sends Modbus requests to the slaves connected to the Modbus RTU network). These requests will be defined under the "Modbus devices" tab. Parameters to be configured are:
  - "Enable": Do not enable any Modbus RTU client if not needed.
  - "UART": Select the UART to be associated to the Modbus RTU client.  
NOTE: Once a given UART is selected to be a Modbus RTU client, it cannot be used with any other configuration such as COM client.
  - "Stack depth": number of requests that can be stacked in the UART buffer. It is recommended to set this value to 1 by default.
  - "Timeout": it defines the amount of milliseconds defined as time out.
  - "Number of repeats": number of retries defined in case of transmission error.
  - "Delay before next request": In case peripheral devices need some "extra time" to answer after a received request  
NOTE: After configuration is completed, click on **"Set"** button in order to save changes.
- "Modbus RTU Server". This mode is used when the MPC-374 will be used as Modbus slave in a network. Requests from a Modbus server will be received and answered.
  - "Enable": Do not enable any Modbus RTU server if not needed.
  - "UART": Select the UART to be associated to the Modbus RTU server.  
NOTE: Once a given UART is selected to be a Modbus RTU server, it cannot be used with any other configuration such as COM client.
  - "Address": Define Modbus address for MPC-374. Modbus master must send requests to the given address.  
NOTE: After configuration is completed, click on **"Set"** button in order to save changes.

## 7.5 Communication >Data transfer

Data collected by MPC-374 can be sent in a CSV file through the configuration of an FTP client inside the controller. This file is, thus, sent to a remote FTP server properly defined as shown below. FTP Configuration must be done in two steps:

1. Define CSV file structure: Go to *Communication>Data transfer>Common parameters>* and configure the following parameters:
  - "Amount of group of registers": Number of registers or group of registers to be stored in the CSV file.
  - "Set user archive". If selected, it enables automatic storage of selected registers in the internal memory. It is recommended to always select this option so that in case of file transmission failure, data can be retrieved and resent when communication is reestablished.
  - "Parameter". It is filled in automatically.
  - "Register/Coil" Initial Modbus register address.
  - "Amount of registers". Number of registers to be stored from the initial Modbus registers address.
  - "Format": Data type of registered parameter.
  - "Header": Fixed column header to be written in CSV file.
  - "Dimension": Extra column in which data units can be stored.

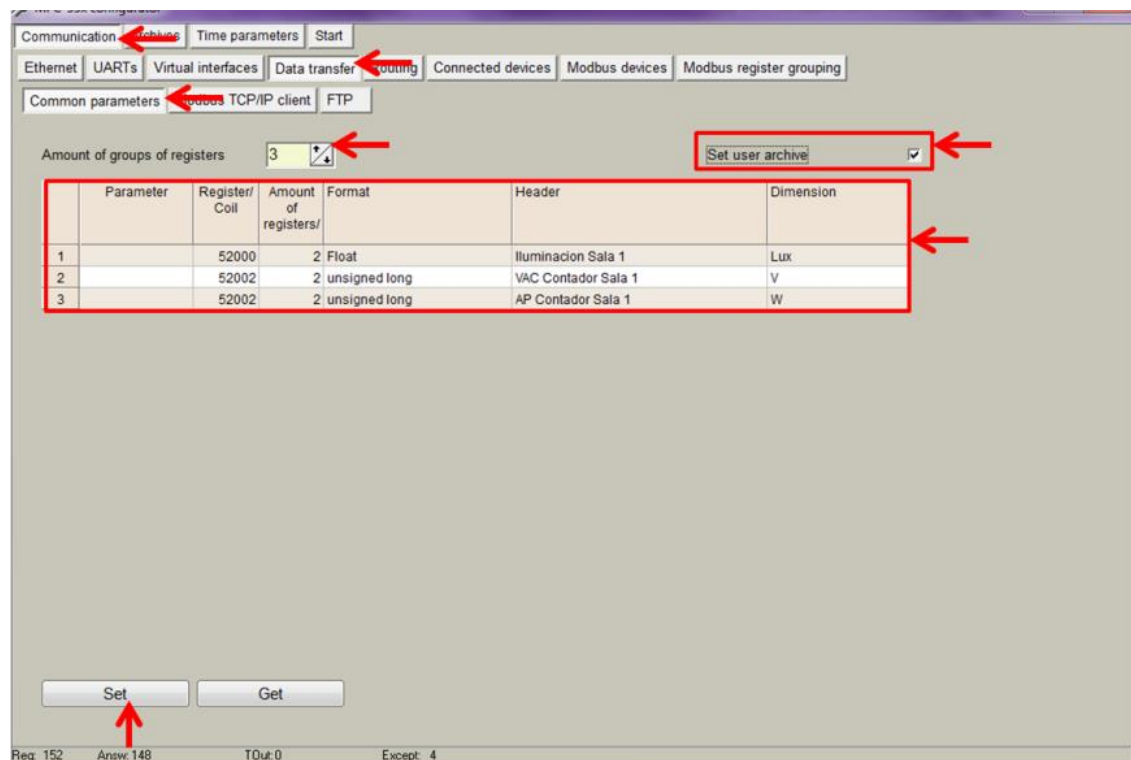
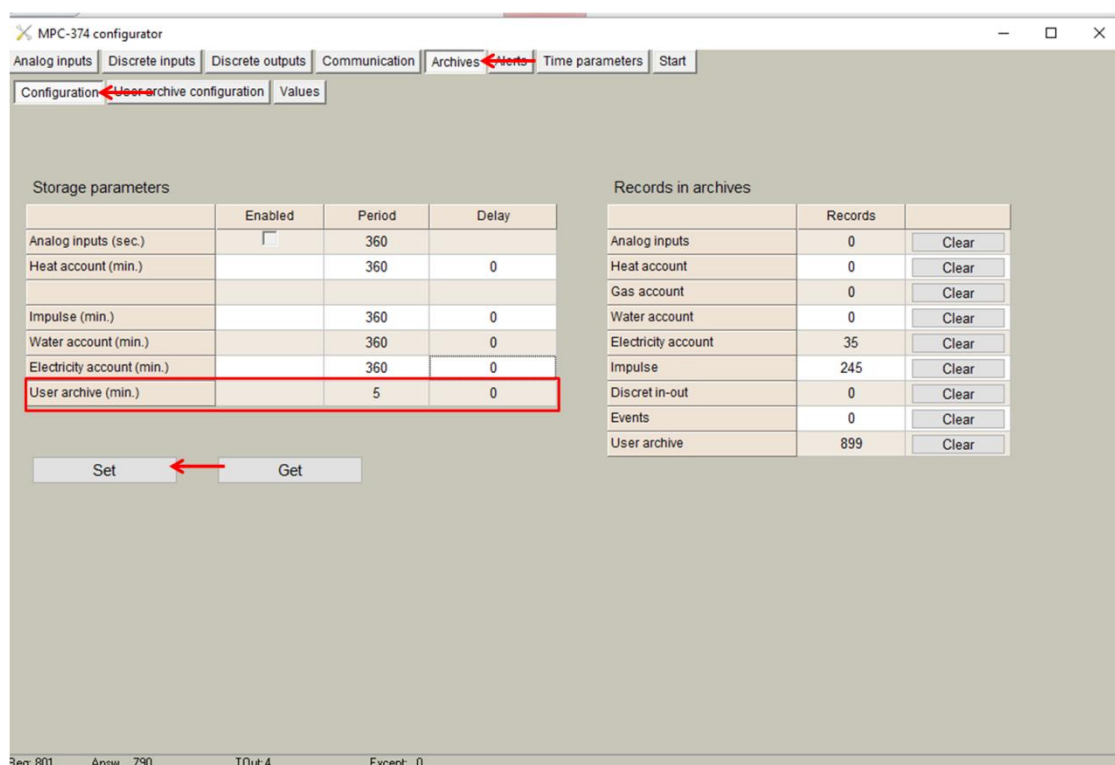


Figure 20. "Data transfer" tab. CSV file structure definition

NOTE: Go to *Archives>Configuration* and specify 'User archive (min)' value to set the datalogging period as shown in the following figure.



**Figure 21. Setting up Datalogging period**

NOTE: After configuration is completed, click on "**Set**" button in order to save changes.

2. Configure FTP client: Go to *Communication>Data transfer>FTP>* and configure the following parameters:
  - "FTP server": Up to 4 different FTP connections to remote FTP servers can be configured.
    - "Enabled": Select it to enable an FTP connection
    - "FTP Port": By default, 21 but can be changed.
    - "Transfer period": It defines interval between CSV file sending tasks.
    - "Number of last records to be read": It defines maximum number of previous data stored and not sent due to communication error. These data will be sent in a CSV file when communication is restored.
    - "Transmission channel": It can be Ethernet or GPRS
  - "URL and directory of FTP servers": It defines the complete FTP server URL where CSV file will be hosted.
  - Username and password of FTP server: to be configured in case credentials are requested by FTP server.

The screenshot shows the 'Data transfer' configuration window. The 'FTP' sub-tab is selected. The configuration is organized into four columns for FTP server 1 through 4. Red arrows point to the 'FTP' sub-tab, the 'FTP server 1' column, and the 'FTP port' and 'Transmission channel' fields for server 1.

	FTP server 1	FTP server 2	FTP server 3	FTP server 4
Enabled	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
FTP port	21	21	21	21
Transfer period (min.)	5	0	0	0
Max number of last records to be	10	10	10	10
Transmission channel	<input checked="" type="radio"/> Ethernet <input checked="" type="radio"/> GPRS	<input checked="" type="radio"/> Ethernet <input checked="" type="radio"/> GPRS	<input checked="" type="radio"/> Ethernet <input checked="" type="radio"/> GPRS	<input checked="" type="radio"/> Ethernet <input checked="" type="radio"/> GPRS

URL and directory of FTP servers

FTP server	URL and directory
1	192.168.1.99/
2	
3	
4	

Username of FTP server

FTP server	User name
1	usuario1
2	
3	
4	

Password of FTP server

FTP server	Password
1	user1
2	
3	
4	

Buttons: Set, Get

FTP server	Status	Successful/attempt/last record transfer times
1	Not active	01/01/1998 00:00:00 01/01/1998 00:00:00 01/01/1998 00:00:00
2	Not active	01/01/1998 00:00:00 01/01/1998 00:00:00 01/01/1998 00:00:00
3	Not active	01/01/1998 00:00:00 01/01/1998 00:00:00 01/01/1998 00:00:00
4	Not active	01/01/1998 00:00:00 01/01/1998 00:00:00 01/01/1998 00:00:00

Req: 2862    Answ: 2858    TOut: 0    Except: 4

**Figure 22. "Data transfer" tab. FTP client configuration**

## 7.6 Communication > Routing

Since MPC-374 can work as a communication gateway, several routing modes can be implemented in it. When used as gateway, all the information received by MPC-374 is forwarded following the chosen configuration, not storing any of these data in its memory:

- **TCP/IP - serial (request)**

Making use of this mode, two devices can be connected between a TCP/IP port and a UART. In this case, the TCP/IP device must initiate communication and the RS485/RS232 will answer under "request". Parameters to be configured are:

- "Enabled": Mode enabling/disabling.
- "Socket live time": Duration in which socket is kept active. When the device initiating communication sends a request command to the slave device, communication will be shut down when this time is reached.
- Routing 1,2,3,4,5,6:
  - "TCP port": Routing is done through this port.
  - "COM client": Previously configured under "Communication > Virtual interfaces" tab

NOTE: After configuration is completed, click on "Set" button in order to save changes.

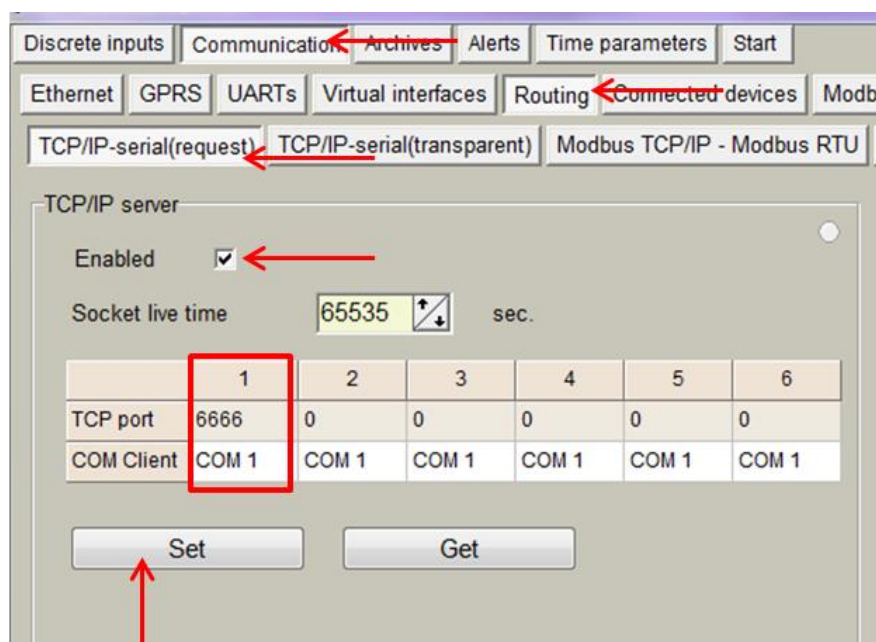


Figure 23. "Communication > Routing" tab. TCP/IP - serial (request) routing configuration

**USE CASE:** ZIV-branded electricity meter connected to a remote PC through a TCP/IP - serial (request) routing making use of ZIV commercial software.

- The meter is physically connected to UART 4 through RS-232 port.
- ZIV software is installed in a remote PC which is connected through Ethernet port to a MPC-374.
- ZIV software opens a socket requesting information to the meter making use of its own protocol. IP address and port of meter is provided by MPC-374 routing mode. In this particular case, it is 192.168.1.1:1005.

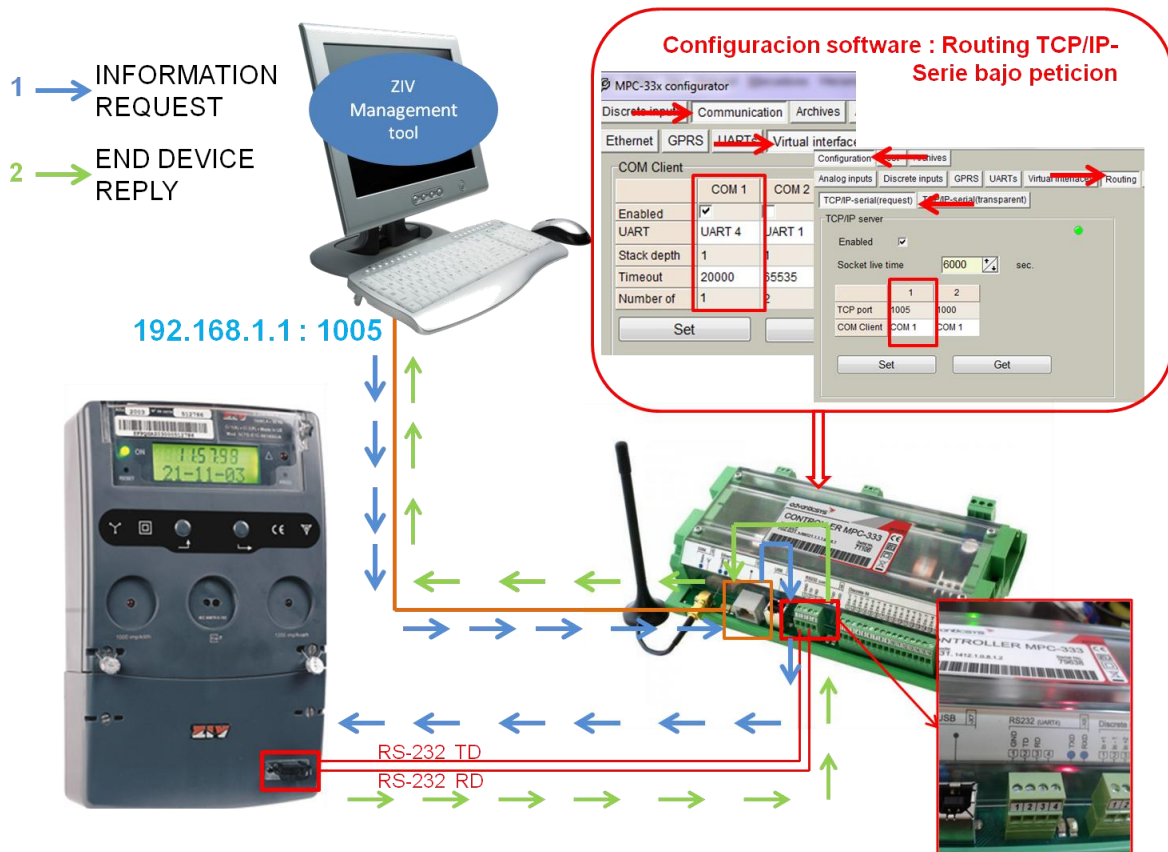


Figure 24. TCP/IP serial (request) routing use case



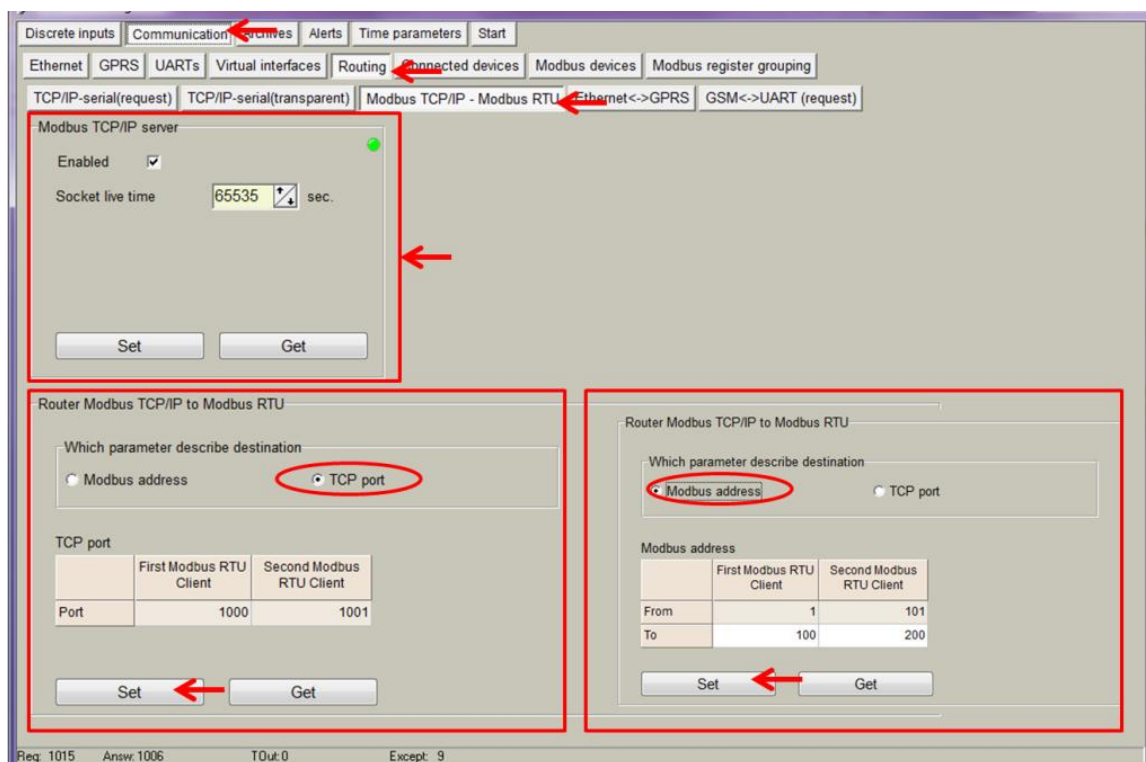
○ **Modbus TCP/IP - ModBus RTU**

The MPC-374 can also forward Modbus packets from a TCP port to an Modbus RTU client. In order to do so, Modbus TCP/IP server mode must be always enabled as shown in the figure. Also "Socket live time" parameter must be configured given that it is the duration in which socket is kept active. When the device initiating communication sends a request command to the slave device, communication will be shut down when this time is reached.

This routing mode can be done in two ways:

- Destination "TCP port": This mode enables assignment of a TCP port to each Modbus RTU device as previously configured under *Communication>Virtual interfaces>Modbus RTU Client* tab.
- Destination "Modbus address": Within this mode, user can identify Modbus addresses to each Modbus RTU device as previously configured under *Communication>Virtual interfaces>Modbus RTU Client* tab. With this configuration, Modbus commands can be sent to devices within the network making use of the specific device Modbus address and the MPC-374 IP address and TCP port 502.

NOTE: After configuration is completed, click on "Set" button in order to save changes.

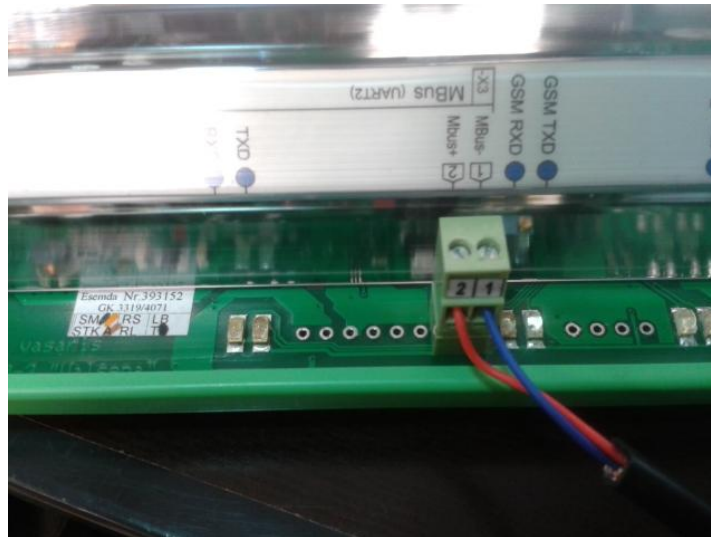


**Figure 25. "Communication > Routing" tab. Modbus TCP/IP – Modbus RTU routing configuration**



## 7.7 Communication > Connected devices

Within this section, M-Bus communication can be configured given that MPC-374 is able to collect data from up to 50 M-bus devices through the Mbus UARTs (in the example UART2) as shown in the following picture.



**Figure 26. Connecting M-bus devices to MPC-374 UART 2**

Although the MPC-374 automatically detects standard M-Bus devices, it is also possible to perform a manual configuration of devices which are not totally M-Bus compliant. To do so, the MPC-374 configuration tool organize different configuration parameters into different tabs according to the magnitude M-Bus meters are reading (heat, water, gas, electricity). In the following pages, a detailed description of each M-Bus meter configuration step is done:

- **Heat Meters**

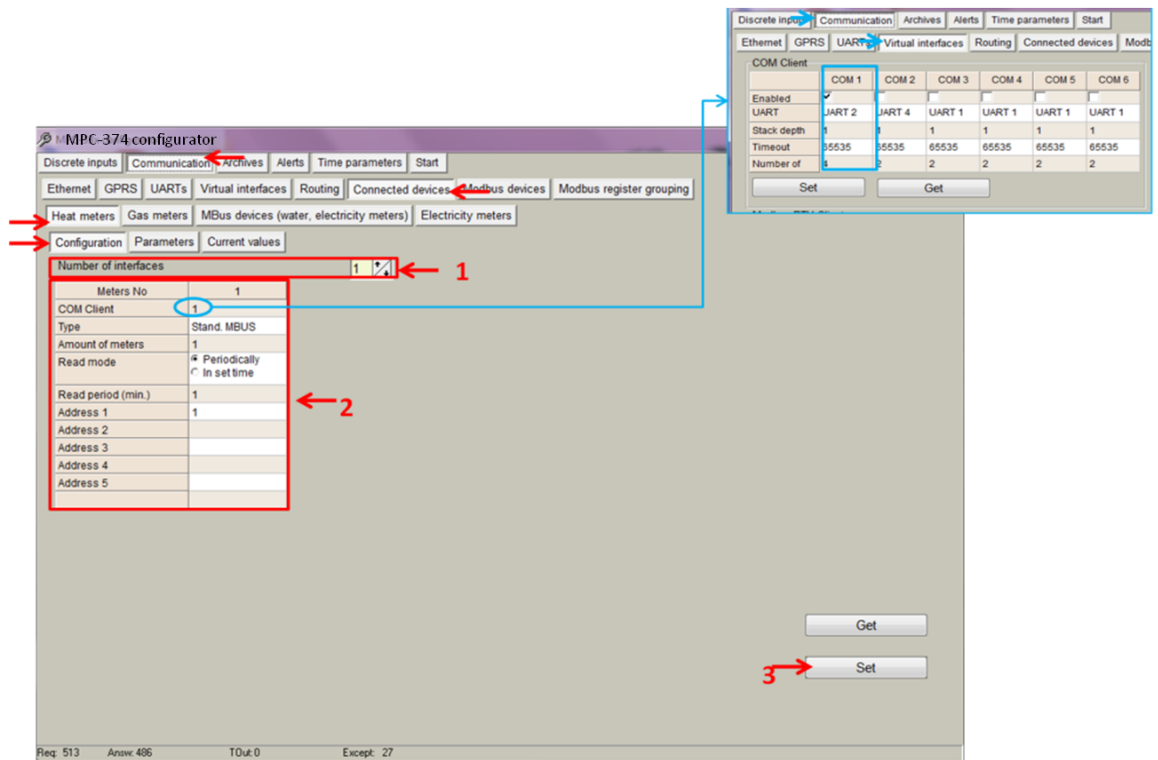
Configuration tool has already some preconfigured commercial devices easing the parameters organization process. These models are: SKS-3, SKU-01S, SKU-01M, SKU-1, SKM-1, Meterman-III, FP-93, enConvert, Elkora, Multical-601, Sitrans-F, Supertrol II and Compart DXF35. If your device is not among the previous list, you must click on **"Type"** and choose **"Stand. Mbus"**.

In order to read data from the device, two configuration steps must be followed: communication and parameters.

- Communication

- Go to *Communication > Connected devices > Heat Meters > Configuration*
    - Select number of M-Bus networks or M-Bus devices groups under "Number of interfaces" list.
    - "Com Client": Select COM Client previously configured under Communication>Virtual interfaces>COM Client tab.
    - "Amount of Meters": Indicate number of Mbus devices.
    - "Read mode":
      - "Periodically": data read requests are sent every given period.
      - "In set time": data read requests are sent following a programmed schedule
    - "Read Period": If periodic read mode selected, this parameter indicates reading period in minutes.
    - "Address": M-Bus devices addresses.

NOTE: After configuration is completed, click on **"Set"** button in order to save changes. Also, perform a hardware reset.

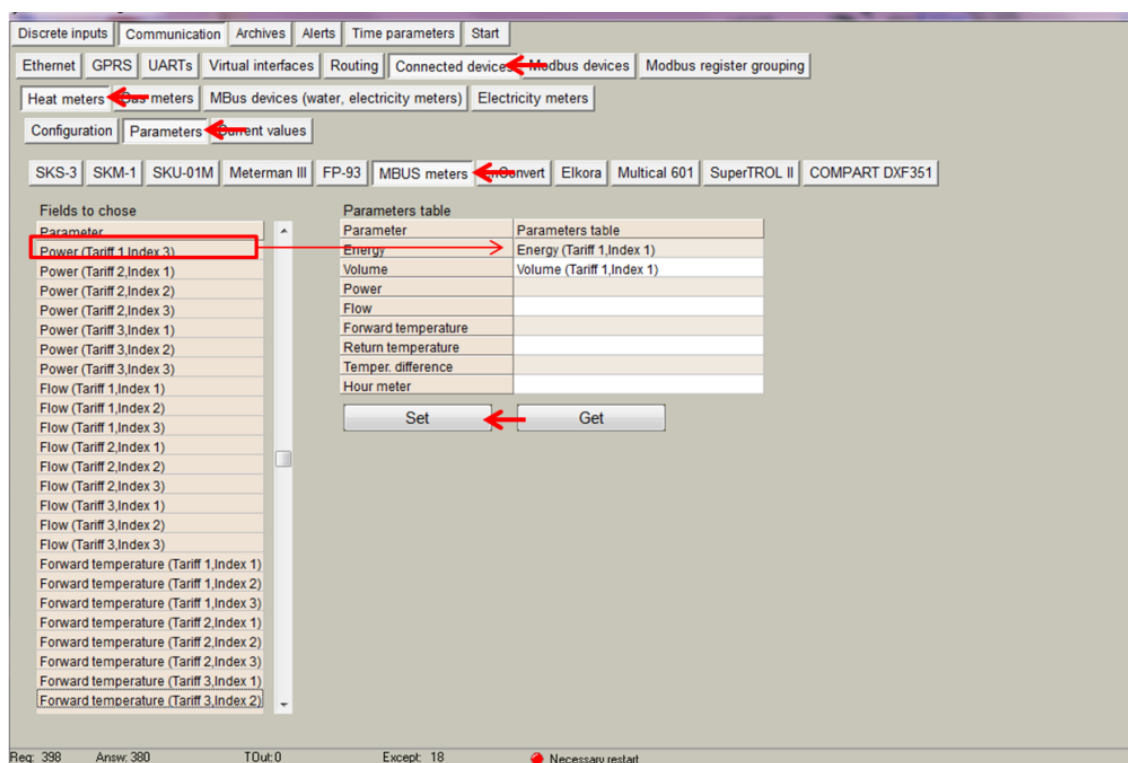


**Figure 36. "Heat meters" communication configuration**

- **Parameters**

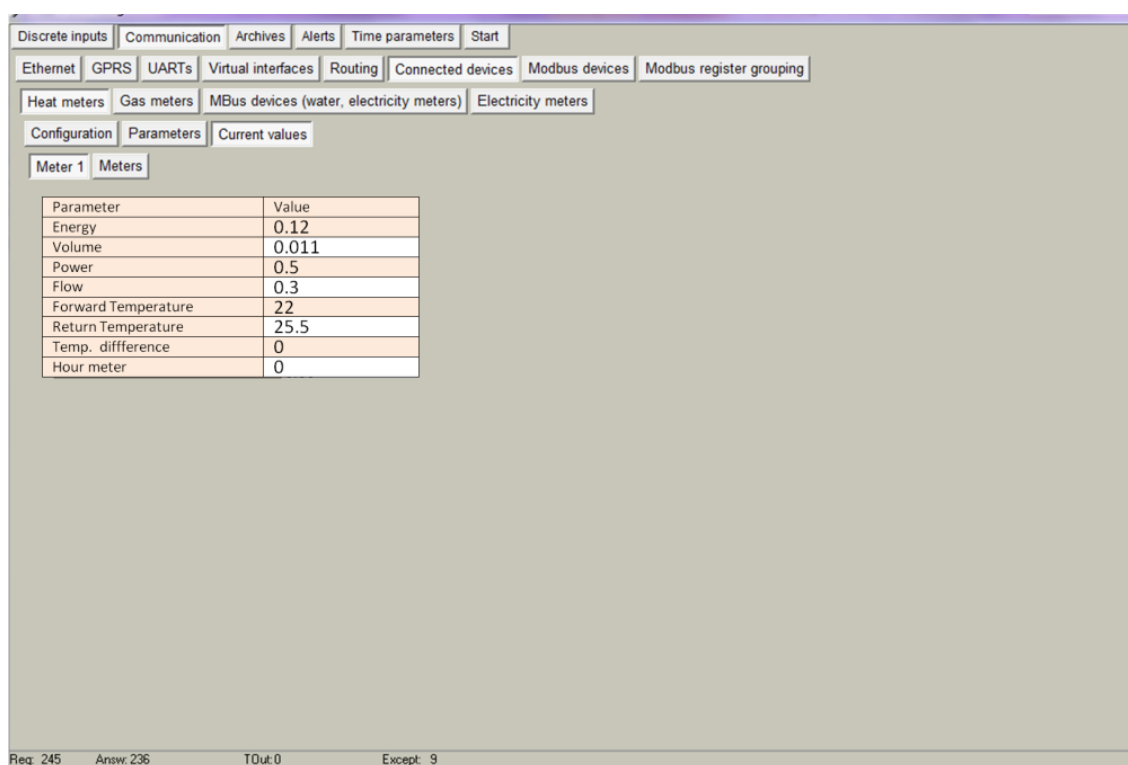
- Go to *Communication > Connected devices > Heat Meters > Parameters*
- If the device is predefined, select the correct option (e.g. if the device is a Kamstrup-Multical 601, go to *Communication > Connected devices > Heat Meters > Parameters > Multical 601*)
- If the device is not predefined, go to *Communication > Connected devices > Heat Meters > Parameters > Mbus meters*.
- "Parameters table" Drag and drop parameters to be read from the "Fields to choose" list.

NOTE: After configuration is completed, click on "**Set**" button in order to save changes. Also, perform a hardware reset.



**Figure 37. "Heat meters" parameters configuration**

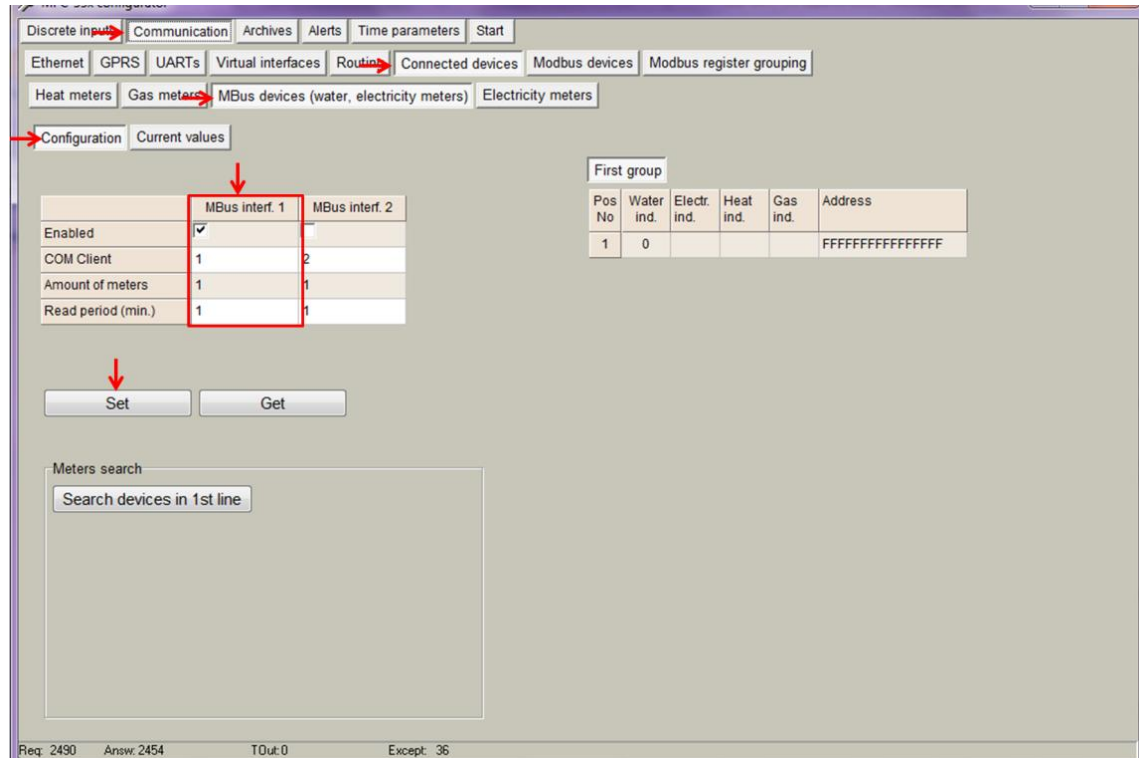
Meter readings can be checked under "Current Values" tab as shown below



**Figure 38. "Heat meters" Current Value**

- **Gas Meters**  
Select the appropriate tab and follow the same instructions as shown for heat meters.

- **Self-discovery.** Steps to be followed are:
  - Enable COM Client under *Communication>Virtual interfaces>COM Client* tab.
  - Go to *Communication > Connected devices > Mbus devices > Configuration* and configure the following parameters:
    - "Enabled": Select which MBus interface will be active.
    - "COM Client": Select COM Client previously configured.
    - "Amount of meters": Configure the number of M-Bus devices to be read.
    - "Read Period": Reading interval in minutes.



**Figure 39. "Mbus devices" tab. Communication configuration**

NOTE: After configuration is completed, click on "Set" button in order to save changes. Also, perform a hardware reset.

- Once MPC-374 has been reseted, go to *Communication>Connected devices>Mbus devices>Configuration* and click on "Search devices in 1st line" button. After a few seconds, a list including all the M-Bus devices connected to the network will be shown. Moreover, a list of its main parameters will be also identified automatically:

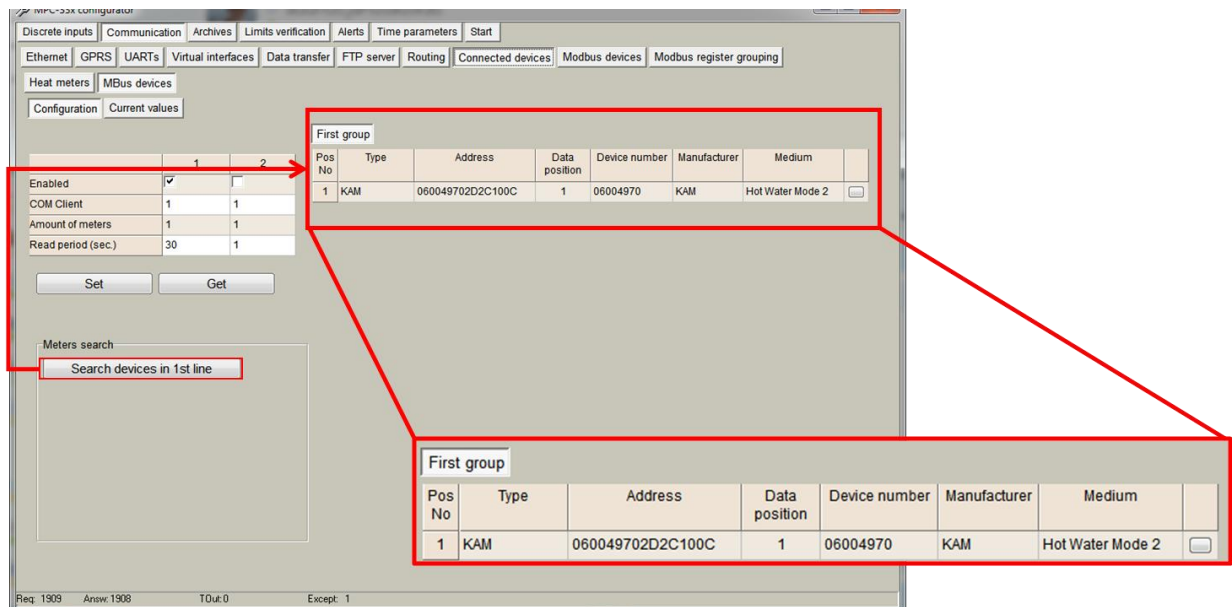


Figure 40. "MBus devices" tab. Self-discovery

- In order to check if data are being received from M-Bus devices, go to "Current values" tab. As an example, data received from a M-Bus MULTICAL 601 meter are shown in the following figure

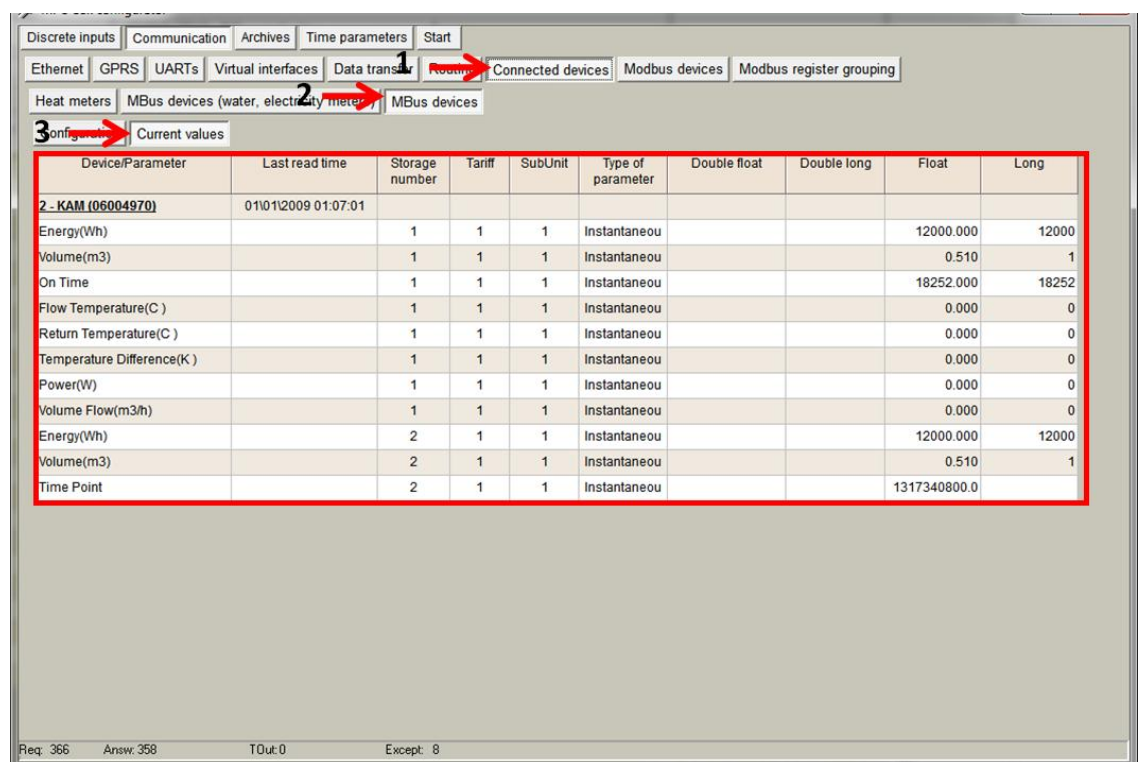


Figure 41 "MBus devices" tab. Current values

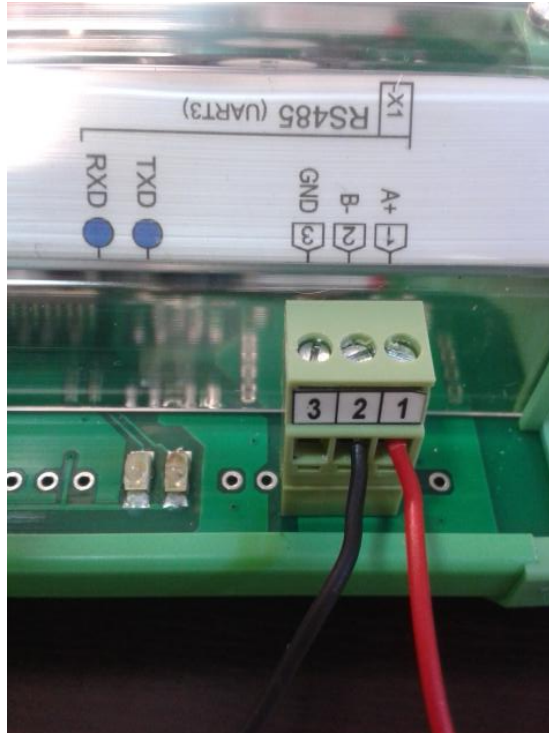
- **Electricity Meters**  
Select the appropriate tab and follow the same instructions as shown for heat meters.

## 7.8 Communication > Modbus devices

There are two methods for reading Modbus devices connected to the MPC-374.

- ✓ By using the MODBUS RTU client to request data from MODBUS RTU devices:  
*Communication>Modbus devices>Configuring Modbus devices> Modbus RTU client*

The MPC-374 can read up to 100 Modbus RTU devices through UART'x' as shown in the following figure.

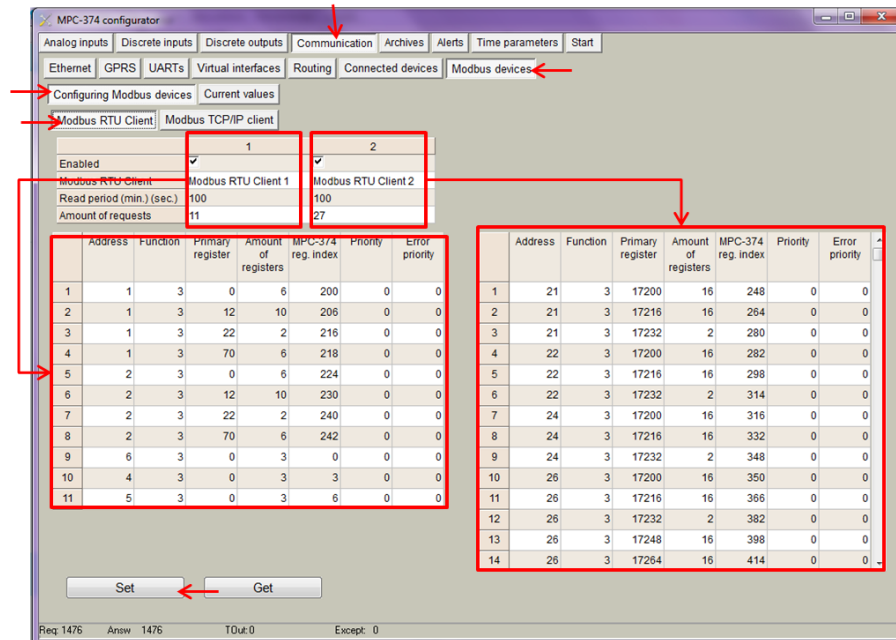


**Figure 27 Connecting Modbus RTU devices to the MPC-374**

In order to configure Modbus RTU client list, follow the next steps:

- Go to *Communication>Virtual interfaces>Modbus Devices*.
- Once the virtual interface has been enabled under Modbus RTU client- 'UART x' , the list of requests for the Modbus client must be configured. For this purpose go to *Communication>Modbus devices>Configuring modbus devices> Modbus RTU client* .It must be configured as much lines as Modbus RTU requests required. The number of requests can be configured in "Amount of requests", where:
  - "Address": Slave device Modbus Address.
  - "Function": Modbus RTU function.
  - "Registers/coils ":Register/s address to be requested.
  - "Amount of registers/coils": number of registers to be read from the register address configured in "Register/coils"
  - MPC- Reg. index: Internal MPC-374's RAM memory index where data is recorded. In case of MPC-374 it starts in the address 52000 (index =0 => RAM Modbus address=52000@DEC).
  - Priority: It configures the request management priority.





**Figure 28. “Modbus devices” tab. Modbus RTU client parameters list configuration**

NOTE: Maximum number of requests per each Modbus RTU Client is 50

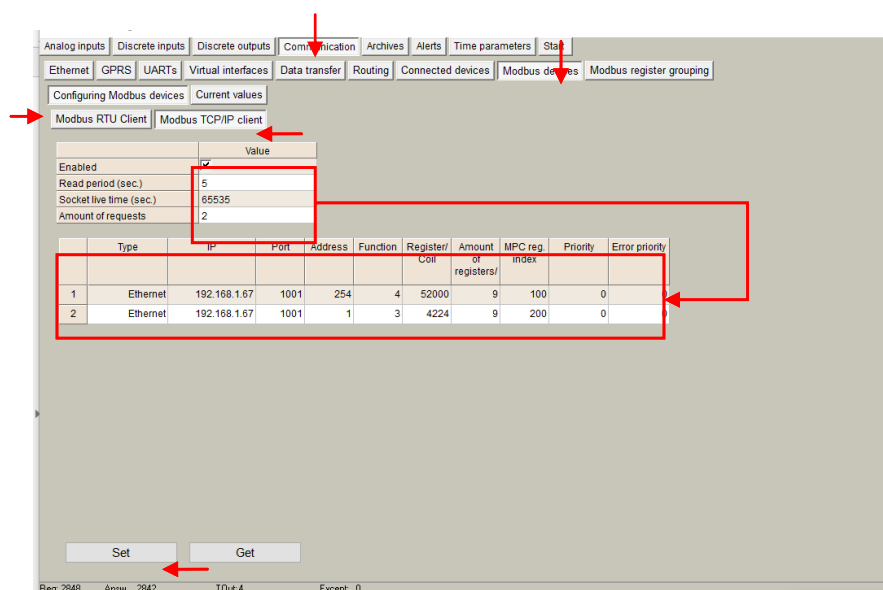
NOTE: There exist 2 Modbus RTU clients. In case that both must work together 2 different UARTs must be used, one per each Modbus RTU client.

✓ By using the Modbus TCP client to collect data from Modbus TCP devices:  
*Communication>Modbus devices>Configuring Modbus devices> Modbus TCP client*

In order to configure Modbus TCP client list, follow the next steps:

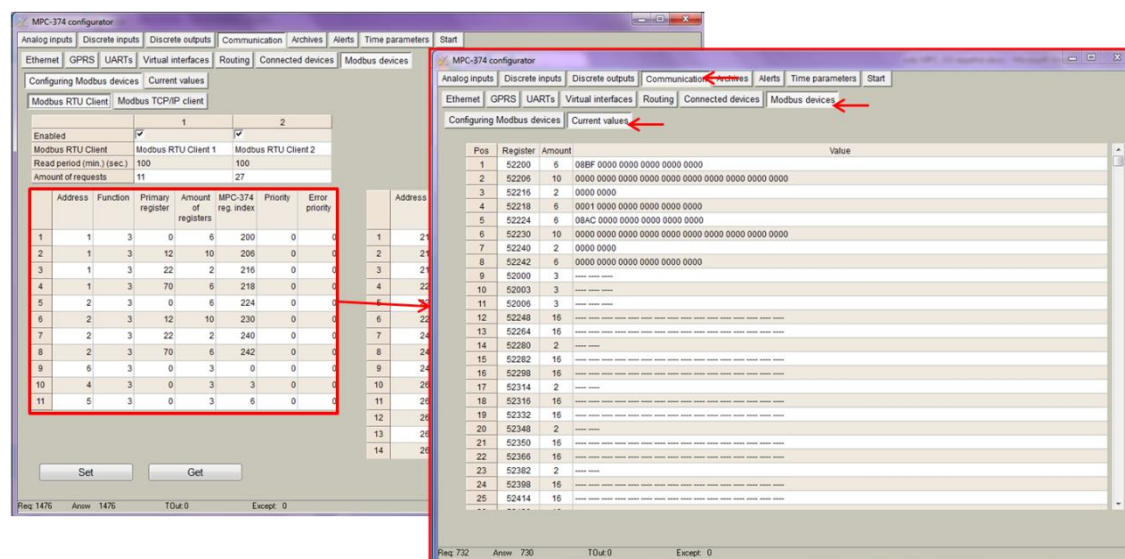
- Go to *Communication>Virtual interfaces>Modbus Devices>Configuring Modbus devices> Modbus TCP client*.
- User must configure as much lines as Modbus TCP requests required. The number of requests can be configured in "Amount of requests", where:
  - Type: Ethernet/GPRS
  - IP: Slave Modbus TCP IP Address
  - Port: Slave Modbus TCP port
  - "Address": Slave device Modbus Address.
  - "Function": Modbus function (3,4,16,...).
  - "Registers/coils ": Register/s address to be requested.
  - "Amount of registers/coils": number of registers to be read from the register address configured in "Register/coils"
  - MPC- Reg. index: Internal MPC-374's RAM memory index where data is recorded. In case of MPC-374 it starts in the address 52000 (index =0 => RAM Modbus address=52000@DEC).
  - Priority: It configures the request management priority.

NOTE: In order to remove any request previously set, write '0' in its "Address" column.



**Figure 29. “Modbus devices” tab. Modbus TCP client parameters list configuration**

After any of the previous method has been configured, it is possible to check the received information through the Modbus RTU/TCP client in “Current values” tab, also enabling correct performance testing.



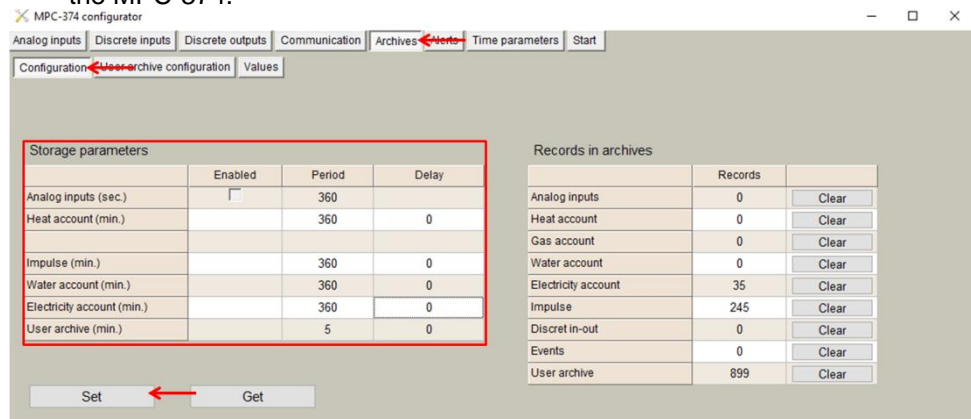
**Figure 30. “Modbus devices” tab. Current values**



## 8 "Archives" tab

The MPC-374 has an internal 8MB flash memory. In case, the device is used as datalogger, the following steps must be done:

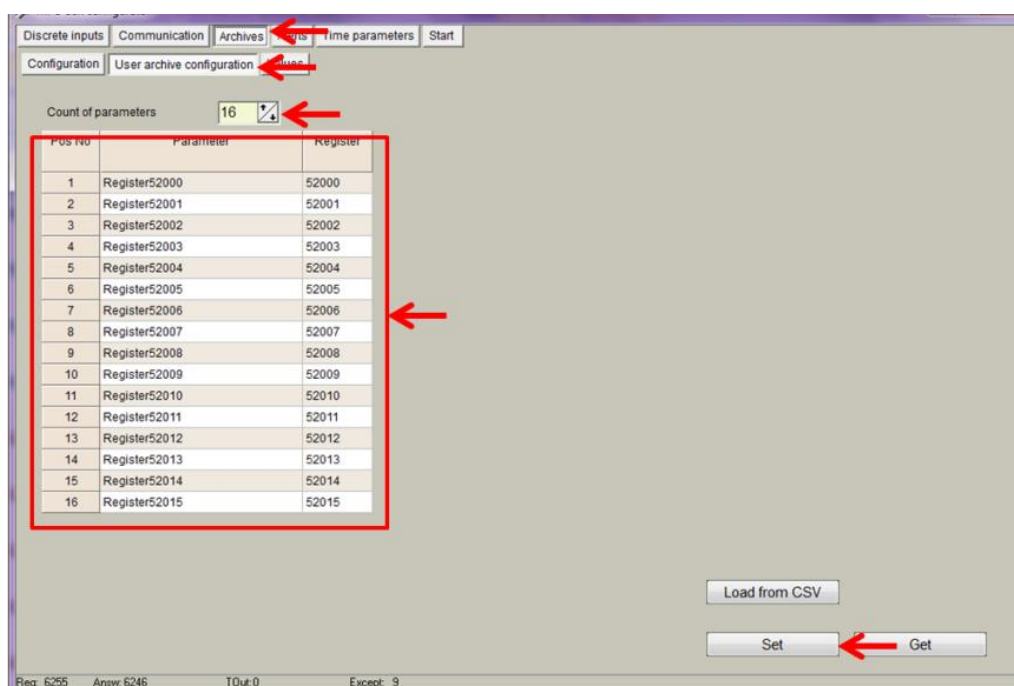
- Go to *Archives>Configuration*
- In "Storage parameters" frame, configure the following:
  - "*Period*": It defines storage interval. Internal memory is organized in different blocks depending on the devices nature which are connected to the MPC-374.



**Figure 45. "Archives" tab. Storage frequency configuration**

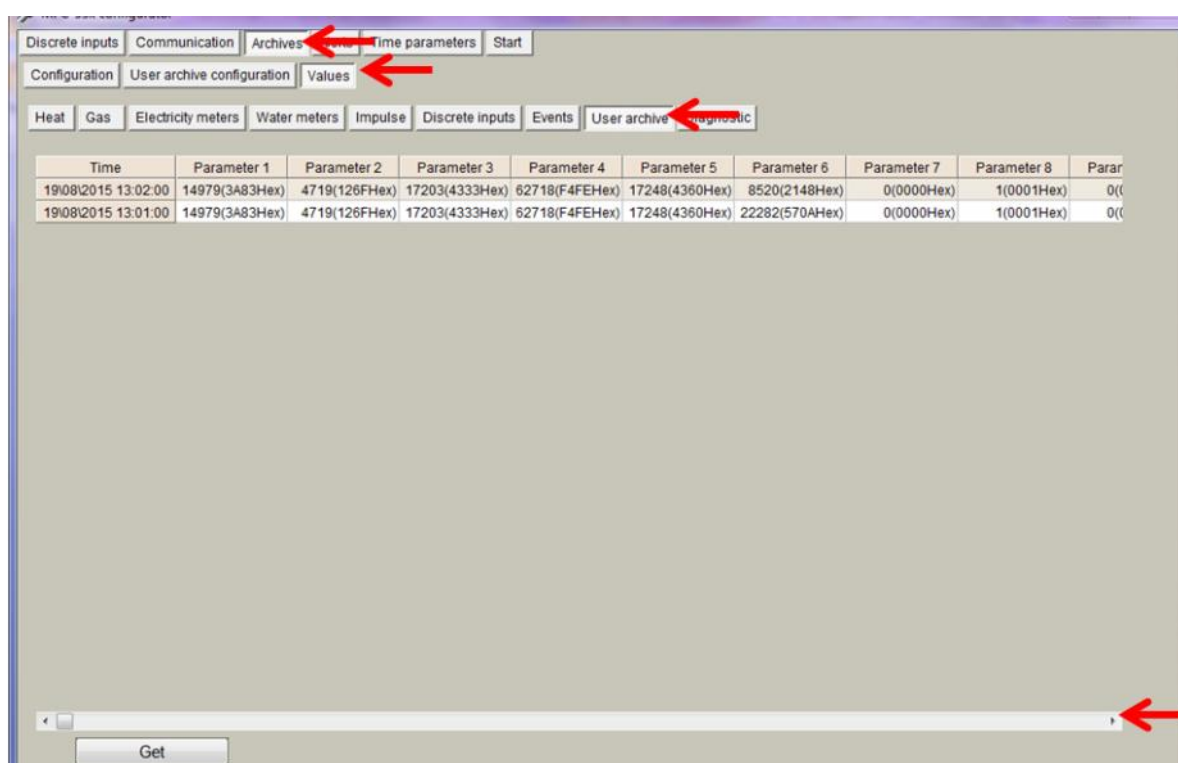
- In case user needs to customize storage blocks, signals acquisition must be configured accordingly. Memory block used will be defined as "User Archive" (see next step).
- Go to Archives>User Archive Configuration
- Under this tab, user can configure datalogging following his own requirements. In the next pages, a configuration example is given by setting the following parameters:
  - "*Count of parameters*": number of registers to be stored.
  - "*Register*": Specific register to be stored.

NOTE: Timestamp is registered automatically.



**Figure 46. “Archives” tab. User archive configuration**

In order to check current data logged in the internal memory, go to *Archives > Values > User archive* tab. This is only a test feature in order to ensure datalogging is performing correctly.



**Figure 31. “Archives” tab. User archive current values**

NOTE: Internal datalogging memory is organized as a file system accessible through the use of standard Modbus function 20 “File register”.

## 9 "Time parameters" tab

This tab enables users to set up time synchronization between MPC-374 Real Time Clock (RTC) and external time references. Several parameters can be configured under this tab:

- "Set PC time" It synchronizes internal RTC with PC time.

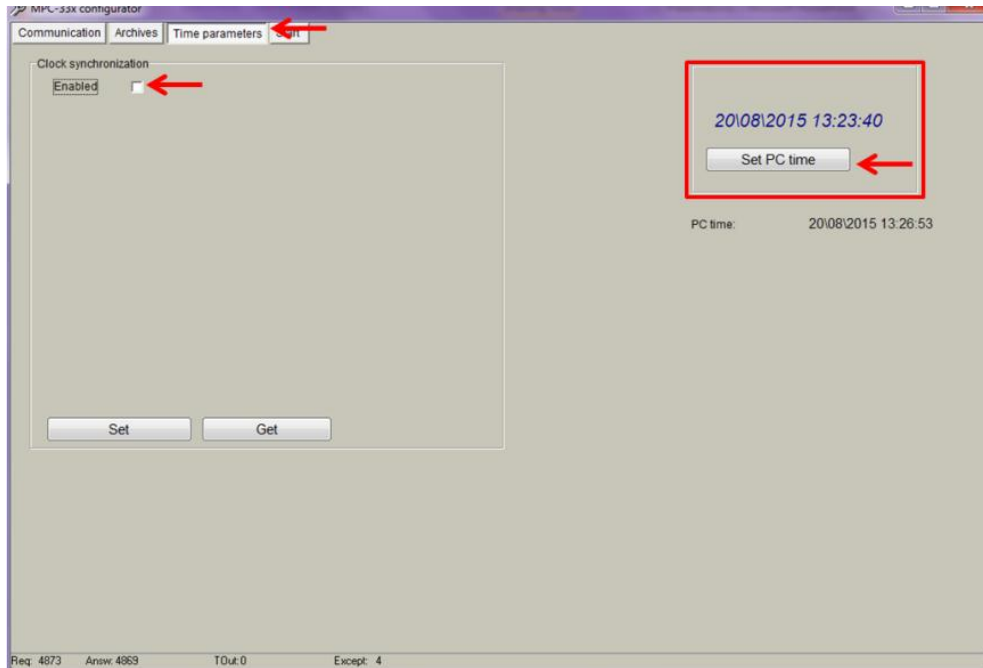


Figure 48. "Time Parameters" tab. Set PC time

- "Clock synchronization": It must be enabled if synchronization between MPC-374 and any Network Time Protocol (NTP) server is requested. This option is particularly accurate since, every given period ("Synchronization period" parameter), device connects to an NTP server to get current UTC time. Also automatic summertime is adjusted.

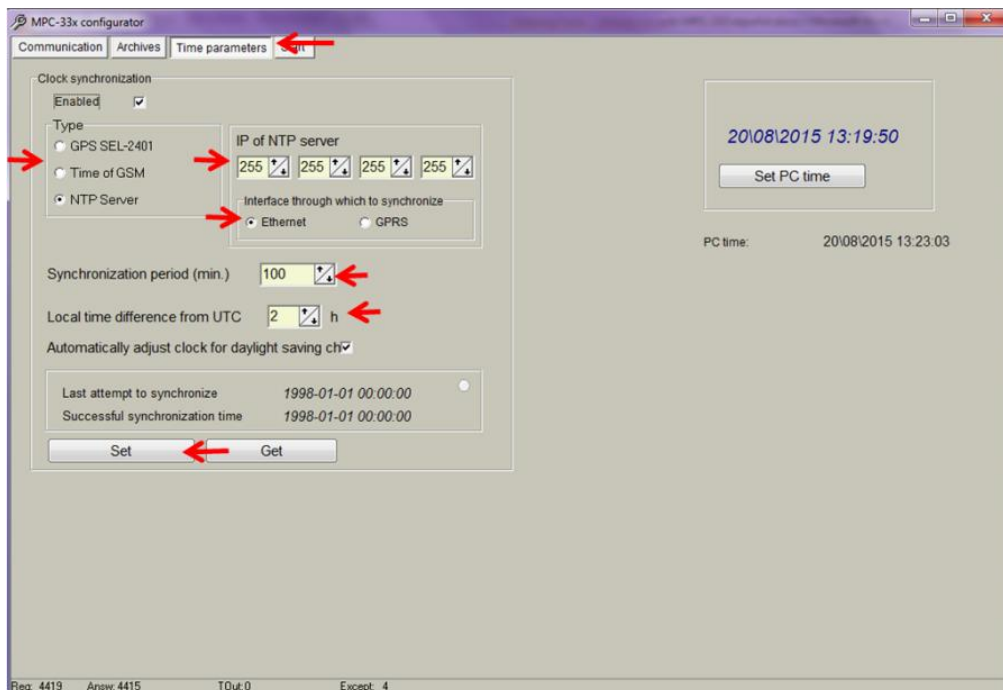


Figure 49. "Time Parameters" tab. NTP server configuration

## 10 Abbreviations and explanations

- **Xn** – is a number of socket. This information are provided for manufacturer's purpose and used in data schemas and connection diagrams.
- **GSM – Global Standart for Mobile Communications.** This interfaces is prepared for remote connections and data bidirectional data transfer over Global Standart Mobile network.
- **GPRS** - a packet oriented mobile data service on the 2G and 3G cellular communication system's global system for mobile communications (GSM).
- **Ethernet** - a family of computer networking technologies for local area networks (LANs) commercially introduced in 1980. Standardized in IEEE 802.3, Ethernet has largely replaced competing wired LAN technologies. This interfaces is prepared for connection LAN (Local Area Network).
- **IP address** - An **Internet Protocol (IP) address** is a numerical label that is assigned to devices participating in a network that uses the Internet Protocol for communication between its nodes.
- **TCP/IP** – Transmission Control Protocol is for communication between computers, used as a standard for transmitting data over networks and as the basis for standard Internet protocols.
- **MAC address –Media Access Control** address is a unique identifier assigned to most network adapters.
- **UART** – An **Universal Asynchronous Receiver/Transmitter** is a type of “asynchronous receiver/transmitter, a part of computer hardware that translates data between parallel an serial forms. UART are commonly used in conjunction with communication standards such as EIA RS-232, RS-422 or RS-485. Record (UARTx) on top of enclosure also are used as serial interface number.
- **GND** – ground wire contact
- **RS232** - the traditional name for a series of standards for serial binary single-ended data and control signals connecting between a DTE (Data Terminal Equipment) and a DCE (Data Circuit-terminating Equipment). It is commonly used in computer serial ports. The standard defines the electrical characteristics and timing of signals, the meaning of signals, and the physical size and pin out of connectors. RS232 interfaces are prepared for connection of pheripheral devices (example energy meters, controllers, machines and etc.).
- **TD** – contact for transfer data wire of RS232 socket
- **RD** – contact for read data wire of RS232 socket
- **DTR** – contact for Data Transmit Ready wire of RS232 socket
- **RS485** - standard defining the electrical characteristics of drivers and receivers for use in balanced digital multipoint systems. The standard is published by the ANSI Telecommunications Industry Association/Electronic Industries Alliance (TIA/EIA). Digital communications networks implementing the EIA-485 standard can be used effectively over long distances and in electrically noisy environments. Multiple receivers may be connected to such a network in a linear, multi-drop configuration. RS485 interfaces are prepared for connection of pheripheral devices (example energy meters, controllers, machines and etc.).
- **A+** – contact for positive wire of RS485 socket
- **B-** –contact for negative wire of RS485 socket
- **USB – Universal Serial Bus** is an industry standard, that defines the cables, connectors and protocols used for connection, communication and power supply between computer and electronic devices. USB type B socket is prepared for connection to PC(Personal Computer). USB type A socket is prepared for connection to pheripheral devices (example memory stick's and etc.).
- **M-Bus** - a European standard (EN 13757-2 physical and link layer, EN 13757-3 application layer) for the remote reading of gas or electricity meters. The M-Bus interface is made for communication on two wire, making it very cost effective.
- **MBUS+** – contact for M-Bus positive wire
- **MBUS-** – contact fot M-Bus negative wire

- **Socket** – is an endpoint of a bidirectional inter-process communication flow across an Internet Protocol-based computer network, such as the Internet.
- **Data** – contact for data wire
- **Req** – contact for request wire
- **CL+** – contact for current loop positive wire
- **CL-** – contact for current loop negative wire
- **Status** – device status indicating LED
- **Uoutput** – status of power for external device indicating LED
- **TX/RX** – data transfer/receive indicating LED
- **TXD** – data transferring LED indicator
- **RXD** – data receiving LED indicator
- **100Mbps** – Ethernet High speed connection indicating LED
- **Central computer** – server or a computer, where data can be sent.

## 11 Safety instructions

To install and setup device, special technical knowledges are needed. Call to seller or certified professionals to connect and setup device !

### Before connecting to power supply, be sure that:

1. Controller is not damaged (no cracks, melted, broken or exposed areas )
2. Controller is used with right and correct thickness cables.
3. Controller and antenna are installed indoor.
4. The controller is intended for supply from a Limited Power Source (LPS) with current rating of overcurrent protective device not greater than 2A
5. The highest transients on the DC secondary circuite of LPS, derived from AC main supply, shall be less then 71V peak.
6. The associated equipments (AE): PC and PSU (LPS) shall comply with the requirements of Standard EN 60950-1.
7. Controller is dry;
8. Ambient temperature and humidity is in normal range;
9. Other types of devices (counters, etc.) are connected correctly by using manufacturer's regulations.
10. The end of stranded conductor shall not be consolidated by soft soldering and must to be terminated
11. Device, PC and other pheripheral devices are strictly connected through one double pole breaker (current break less than 5A and space between breaker contacts more than 3mm.) Pole breaker has to be in building's wiring and in reachable place with markings

### Don't use:

1. Device under open water (in rain and if water are spalshing on controller or connected devices;
2. Device if enclosure, connected cables, or other connected devices are damaged;
3. External Back-Up batterys for powering of controller.



**Use device by manufacturer's regulations otherwise** you can damage controller or other devices. In that cace **munufacturer's warranty could not be obtained.**



**If you suspect that device doesn't operate correctly** or has visible violations, please contact manufacturer or your distributor to check or run maintainance.



**Manufacturer** does not affect and is not responsible for GSM/GPRS/Internet operators' provided network service pricing and costs.

## 12 Manufacturer's warranty

All ADVANTICSYS products are warranted to be free of defects in material or workmanship under normal use and service for a period of one year from the date of shipment. This warranty does not include damage resulting from accident or misuse. The warranty is also void if the product is modified.

This warranty is in lieu of all other warranties expressed or implied including the implied warranties of merchantability of fitness for a particular purpose, whether arising by law, custom or conduct, and the rights and remedies provided under this warranty are exclusive and in lieu of any other rights or remedies. In no event shall ADVANTIC SISTEMAS Y SERVICIOS S.L. be liable for consequential damages.


If you believe that your product is defective while still under warranty, contact ADVANTICSYS by e-mail at [info@advanticsys.com](mailto:info@advanticsys.com), or by phone at +34 911890521. Once the support desk confirms that the product is defective, we will issue you an RMA number and will replace your defective product.

ADVANTICSYS' warranty covers the repairs (manpower and materials) of each manufacturing defect that may obstruct the right operation of the Product. The replacement of any component or damaged equipment does not mean an extension of the guarantee period. If the Product is found upon examination by ADVANTICSYS to be defective, ADVANTICSYS shall bear shipping costs incurred in returning the Product to Customer as well as all costs involved in ADVANTICSYS' examination of the Product. If the Product is found upon examination by ADVANTICSYS to not be defective, Customer shall bear shipping costs incurred in returning the Product to Customer.

## 13 Technical Data

### 13.1 Communication interfaces

Interfaces	Technical data
RS485	up to 1.2 km, max 32 transivers, speed up to 19.2 Kbits/s
RS232	up to 15m, speed up to 19,2 Kbits/s
Data/Req	(KAMSTRUP) data transfer interface
Current loop	<30V, 14-20mA, up to 6 km, speed up to 19,2 Kbit/s
M-Bus	Up to 8 devices
GPRS	3 band 900/1800/1900 MHz
Ethernet	10/100 Mb twisted pair, up to 100m

<b>USB</b>	Type B, version 2.0
<b>Discrete INPUTS</b>	"Dry contact"
<b>Discrete OUTPUTS</b>	Open contact, up to 50V , up  to 500mA
<b>Analog INPUTS</b>	0/4-20mA, 0-5 mA ; (250Ω); error ±0,15%


## 13.2 Galvanic insulation

<b>Insulation voltage between power supply and second circuits</b>	1500 V
<b>Insulated interfaces</b>	B, C, D (see Configuration code)
<b>Analog INPUTS</b>	Yes, insulation voltage 500V
<b>Discrete INPUTS</b>	Yes. Non-insulated from discrete outputs
<b>Discrete OUTPUTS</b>	Yes. Non-insulated from discrete inputs

## 13.3 Indication

<b>Indication type</b>	LED's
<b>Indicated parameters</b>	<ul style="list-style-type: none"><li>• Each discrete input and output status</li><li>• Power for external devices</li><li>• Serial interfaces Transfer and Receive</li><li>• GSM/GPRS modem status, Transfer and Receive</li><li>• Ethernet Full duplex, 100 Mbs, Transfer/Receive</li></ul>

## 13.4 Power supply

<b>Power supply</b>	9 ÷ 36 VDC 
<b>Power consumption</b>	Up to 10VA

## 13.5 Construction

<b>Mounting</b>	DIN35 rail
<b>Dimensions</b>	277 mm x 128 mm x 50 mm

<b>Tightness</b>	IP20
------------------	------

## 13.6 Climate conditions

<b>Operating temperature</b>	From - 25 °C to + 60°C
<b>Storage temperature</b>	From - 40 °C to + 60°C
<b>Relative humidity</b>	From 5 % to 95 % non-condensing

## 13.7 Safety parameters

<b>Safety requirements</b>	Meets requirements: EN 61010-1:2002
<b>Electromagnetic compatibility</b>	Meets requirements: EN 55022:2000+A1+AC:2002+A2:2003 EN 55024:2000+A1:2003+A2:2003 EN 61000-4-2+A1+A2:2002 EN 61000-4-3+A1:2004 EN 61000-4-4:2005 EN 61000-4-5:2002+A1:2003 EN 61000-4-6:2002+A1:2003

## 13.8 Other parameters

<b>Storage memory</b>	1 ÷ 8 MB (Default 8MB)
<b>Configuration settings storage without power supply</b>	More than 5 years
<b>Remote firmware loading</b>	Yes. Through RS232 or/and Ethernet and GSM/GPRS. Through USB (under separate order).



**Advantic Sistemas y Servicios S.L**

Avda. Europa 14  
28108 Alcobendas (Madrid)



[www.advanticsys.com](http://www.advanticsys.com)  
[info@advanticsys.com](mailto:info@advanticsys.com)

+34 91 189 05 21