
DIN-32

User's Manual

1.2

Document history

Date	Version	Author	Remarks
23/10/2015	1.0	Sara Prior	User's Manual including new discrete inputs pulse counting feature.
12/12/2015	1.1	Sara Prior	Added Modbus TCP to Modbus RTU Routing
26/10/2016	1.2	Sara Prior	Added NTP Synchronization.

Content

DOCUMENT HISTORY	2
CONTENT	3
1 INTRODUCTION	5
2 SETTING UP CONNECTION TO THE DEVICE	6
2.1 USB connection	6
2.2 ETHERNET Connection	6
3 "START" TAB	7
3.1 DIN-32 Basic Information	7
3.2 Configuration files	7
3.3 Status indicators	7
4 "DISCRETE INPUTS" TAB	8
5 "COMMUNICATION" TAB	10
5.1 Communication > Ethernet	10
5.2 Communication > UARTs	10
5.3 Communication > Virtual interfaces	12
5.4 Communication > Routing	13
6 "ARCHIVES" TAB	16
7 "TIME PARAMETERS" TAB	18
8 ABBREVIATIONS AND EXPLANATIONS	19
9 SAFETY INSTRUCTIONS	20
10 MANUFACTURER'S WARRANTY	21
11 TECHNICAL DATA	21
11.1 Communication interfaces	21
11.2 Protection	22

11.3	Indication	22
11.4	Power supply	22
11.5	Construction	22
11.6	Working conditions	22
11.7	Safety parameters	23
11.8	Other parameters	23

1 Introduction

This Manual will show you how to configure the DIN-32 datalogger by using the DIN-32 configurator software.

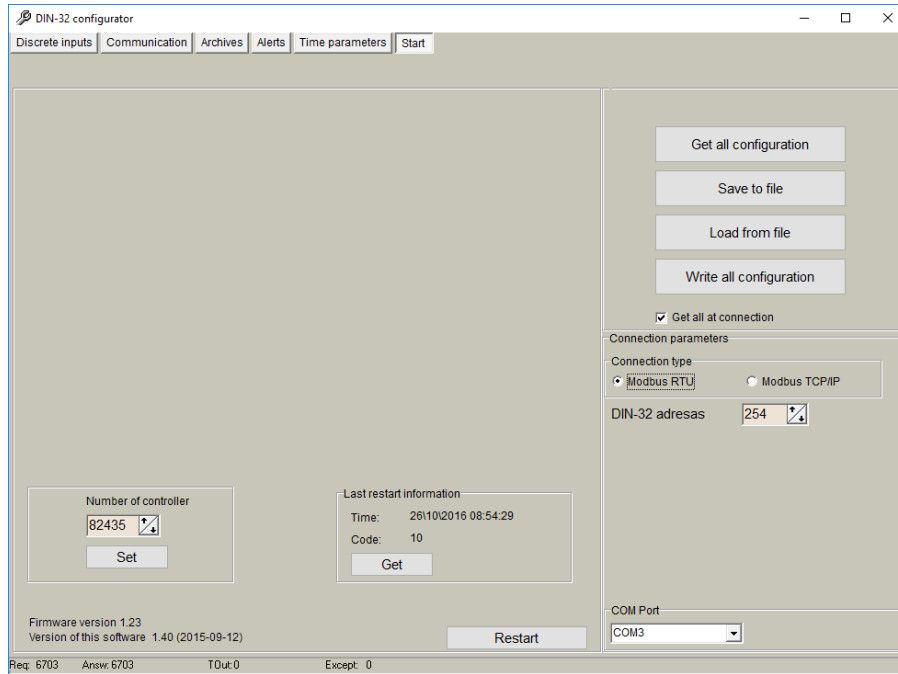


Figure 1. DIN-32 Configurator

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

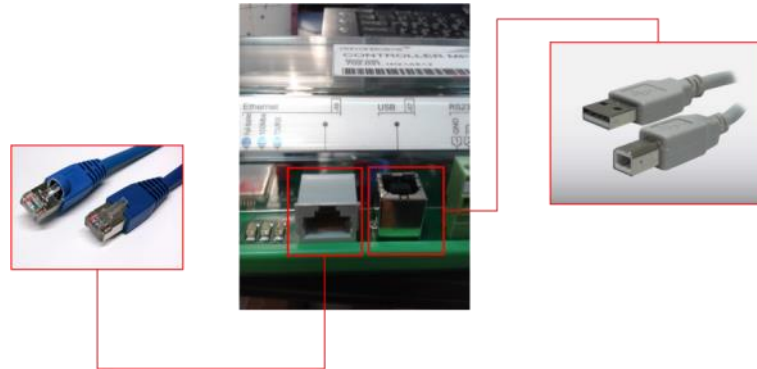


Figure 2. DIN-32 connection interfaces

2.1 USB connection

Steps to be followed:

- Connect an USB cable class B to the correspondent port
- Open DIN-32 Configuration Tool
- Set up "**Connection Parameters**" frame
 - o Select "ModBus RTU" option under "Connection type"
 - o Select 254 as "DIN-32 address"
 - o Select COM port number assigned by your PC to the USB port
 - o Click on "**Get all configuration**" to establish connection with controller.

2.2 ETHERNET Connection

Steps to be followed:

- Connect a RJ45 Ethernet cable to the correspondent port.
- Open DIN-32 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 Click on "**Connect**" to establish connection with controller.

3 “Start” tab

3.1 DIN-32 Basic Information

Once user has established communication with DIN-32, basic information such as "Serial number", "Last restart", "Firmware version", "Software version", etc. is shown in the left frame



Figure 3. “Start” tab. DIN-32 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 DIN-32 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.

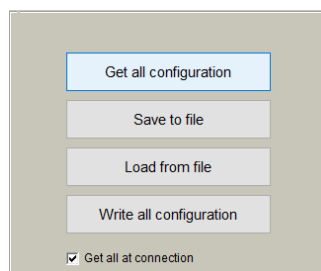


Figure 4. “Start” tab. Configuration files management

3.3 Status indicators

Several status indicators are shown in the DIN-32 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.

Req: 39 Answ: 34 TOut: 0 Except: 0

Figure 5. “Start” tab. Status indicators

4 “Discrete inputs” tab

Discrete inputs can be configured within this tab.

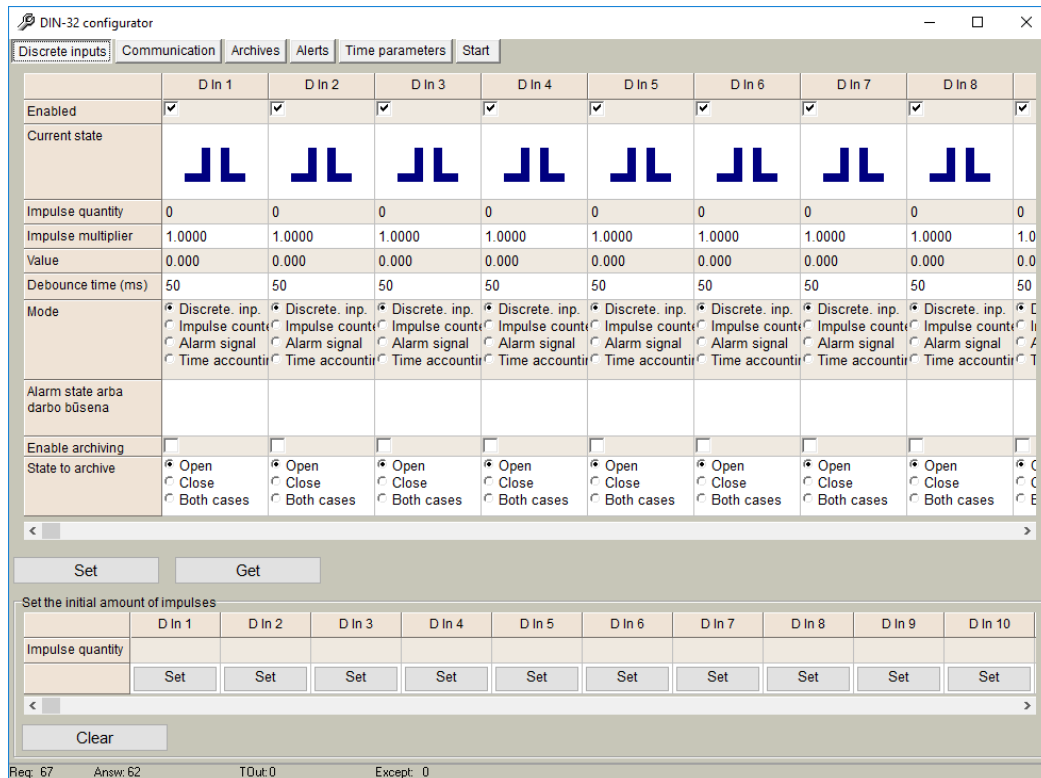


Figure 6. “Discrete inputs” tab

They are shown organized in columns and configuration parameters are shown in rows as follows:

- "Enabled": user enable/disable discrete inputs to be used. In order to save memory, it is recommend to only enable inputs in use.
- "Current State": Graphical representation of current status (open/closed).
- "Impulse quantity": Number of pulses counted since last set to "0"
- "Impulse multiplier": Gain value for each pulse counted.
- "Value": calculated by multiplying impulse quantity and impulse multiplier.
- "Debounce time": only pulses received within one or more "Debounce time" intervals are registered.
- "Mode":
 - "Discrete Input": input works as a standard discrete input detecting its current state (1 / 0)
 - "Impulse counter": input is used to count pulses such as from a water/gas meter.
 - "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.
- "State to archive": selects the discrete input state to be stored

In the lowest part of the screen, it is possible to clear the pulse counter of each discrete input or set it to an initial value if needed.

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

Discrete input parameters and values can be read and set using the following internal Modbus registers:

4200-4231	Channel status (0- enabled, 1 – disabled)	Char	R/W
4232-4263	Channel work mode (0-discrete input, 1-inpuls counter, 2-alarm signal, 3 – time accounting)	Char	R/W
4264-4295	Archiving conditions (0- when low signal, 1-when high signal, 2-both cases)	Char	R/W
4296-4327	Alarm status condition (0-on low signal, 1- on high signal, 2 – both cases)	Char	R/W
4328-4359	Archiving status (0 – disabled, 1 – enabled)	Char	R/W
4360-4391	Channel filtering time ms (0-1000ms)	Int (16 bit)	R/W
4392-4423	Channel status (0 – disconnected, 1-connected)	Char	R
4424-4487	Discrete channel impulse quantity (2*32 registers, 4 bytes per channel)	Long int (32 bit)	R/W
4552-4615	Physical value corresponding to number of impulses (8*float)	Float	R

5 “Communication” tab

5.1 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.

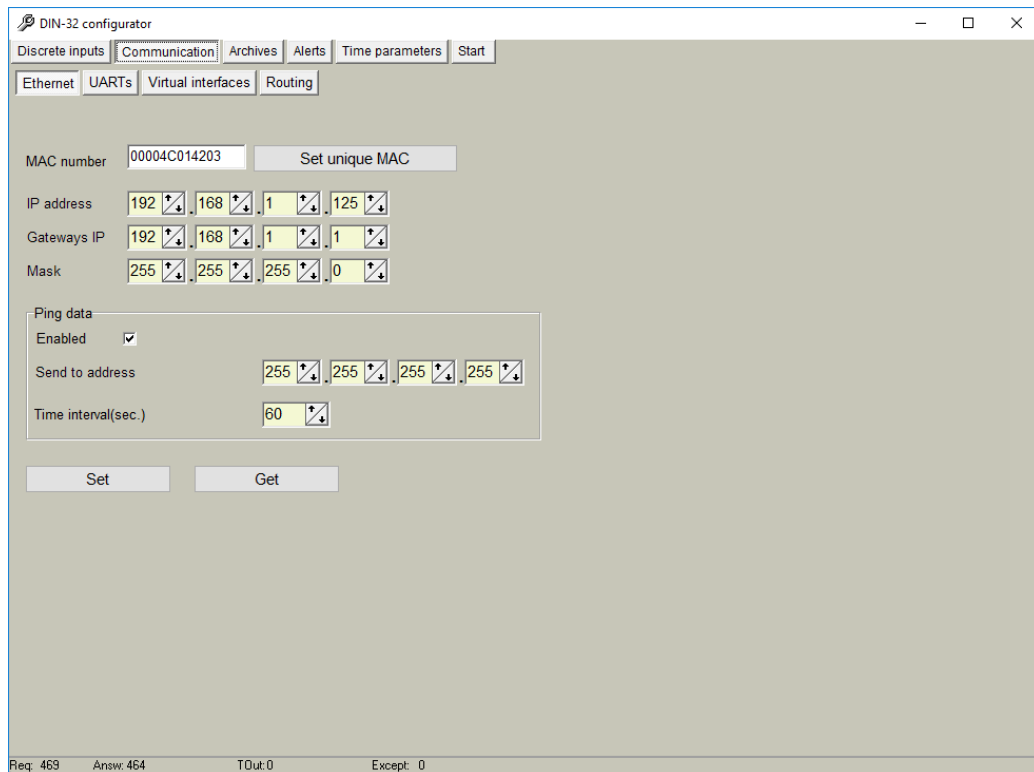


Figure 7. “Ethernet” configuration tab

5.2 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 DIN-32 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 DIN-32 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.

	UART 1	UART 2
Bode	<input type="radio"/> 300 <input type="radio"/> 600 <input type="radio"/> 1200 <input type="radio"/> 2400 <input type="radio"/> 4800 <input type="radio"/> 9600 <input checked="" type="radio"/> 19200 <input type="radio"/> 38400	<input type="radio"/> 300 <input type="radio"/> 600 <input type="radio"/> 1200 <input type="radio"/> 2400 <input type="radio"/> 4800 <input type="radio"/> 9600 <input checked="" type="radio"/> 19200 <input type="radio"/> 38400
Parity	<input checked="" type="radio"/> Even <input type="radio"/> Odd <input type="radio"/> Mark <input type="radio"/> Space <input type="radio"/> None	<input checked="" type="radio"/> Even <input type="radio"/> Odd <input type="radio"/> Mark <input type="radio"/> Space <input 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
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
Packetization	<input checked="" type="radio"/> Time <input type="radio"/> Symbol <input type="radio"/> Length	<input checked="" type="radio"/> Time <input type="radio"/> Symbol <input type="radio"/> Length
Packet time (msec.)	10	15
Packet symbol(Hex)	FF	FF
Packet byte count	65535	65535
Mode	<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

Req: 469 Answ: 464 TOut: 0 Except: 0

Figure 8. “UARTs” tab

5.3 Communication > Virtual interfaces

The DIN-32 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

The screenshot shows the 'Virtual interfaces' tab in the DIN-32 configurator. It contains three main configuration sections:

COM Client

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

Modbus RTU Client

	Modbus 1	Modbus 2
Enabled	<input type="checkbox"/>	<input type="checkbox"/>
UART	UART 1	UART 1
Stack depth	2	1
Timeout (msec.)	1000	1000
Number of repeats	2	2
Delay before next req.(msec.)	100	100

Modbus RTU Server

	Modbus 1	Modbus 2
Enabled	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
UART	UART 1	UART 2
Address	255	255

At the bottom of the window, there are status indicators: Req: 469, Answ: 464, TOut: 0, Except: 0.

Figure 9. "Virtual interfaces" tab

- **"COM Client"**. It is needed to configure a COM Client every time a "transparent" communication mode 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 DIN-32 is used as transparent gateway between TCP/IP and any other protocol used by the devices connected that particular UART.

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

- **"Modbus RTU Client"**. This working mode must be used when the DIN-32 behaves as Modbus TCP to Modbus RTU gateway (it forwards Modbus TCP requests from a Modbus master to the slaves connected to the Modbus RTU network). 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 DIN-32 is 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 DIN-32. Modbus master must send requests to the given address.
NOTE: After configuration is completed, click on **"Set"** button in order to save changes.

5.4 Communication > Routing

Since DIN-32 can work as a communication gateway, several routing modes can be implemented in it. When used as gateway, all the information received by DIN-32 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.

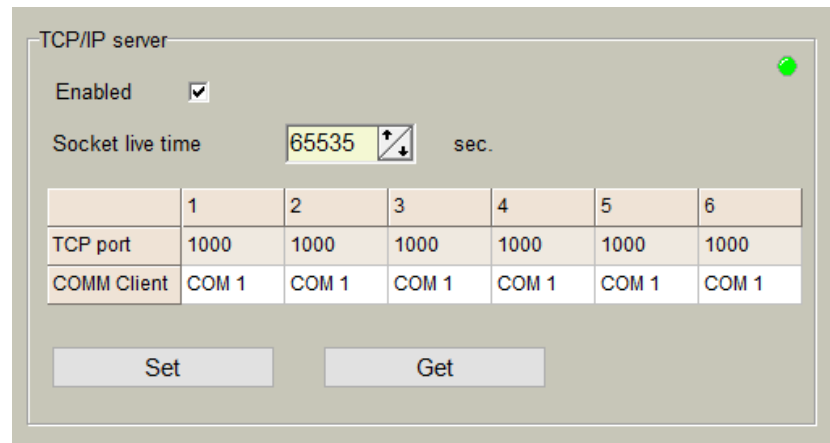


Figure 10. "Communication > Routing" tab. TCP/IP - serial (request) routing configuration

- **Modbus TCP/IP - Modbus RTU**

The DIN-32 can also forward Modbus packets from a TCP port to a Modbus RTU client. In order to do so, Modbus TCP/IP server mode must be always enabled as shown in the figure.

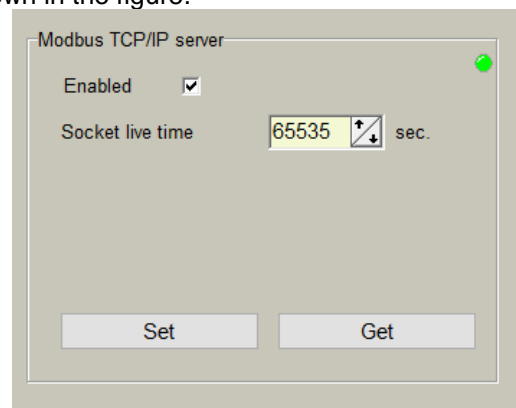


Figure 11. "Communication > Routing" tab. Modbus TCP/IP server configuration

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 "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 DIN-32 IP address and TCP port 502.

Router Modbus TCP/IP to Modbus RTU

Which parameter describe destination

Modbus address TCP port

Modbus address

	First Modbus RTU Client	Second Modbus RTU Client
From	1	1
To	1	1

Set Get

Figure 12. "Communication > Routing" tab. Router Modbus TCP/IP to Modbus RTU. Modbus address configuration

- 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.

Router Modbus TCP/IP to Modbus RTU

Which parameter describe destination

Modbus address TCP port

TCP port

	First Modbus RTU Client	Second Modbus RTU Client
Port	1500	1000

Set Get

Figure 13. "Communication > Routing" tab. Router Modbus TCP/IP to Modbus RTU. TCP port configuration

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

6 "Archives" tab

The DIN-32 has an internal 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 DIN-32.

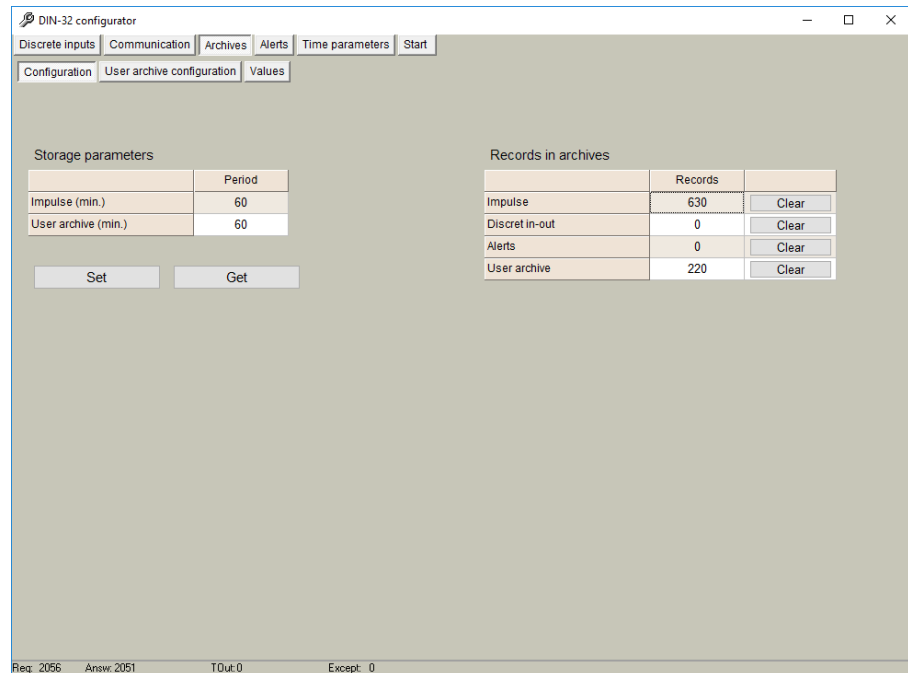


Figure 14. "Archives" tab. Storage frequency configuration

- 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 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.

As an example, if user needs to log impulse quantities, he would need to set the registers listed in "Discrete inputs" chapter in this user manual as shown in the following figure:

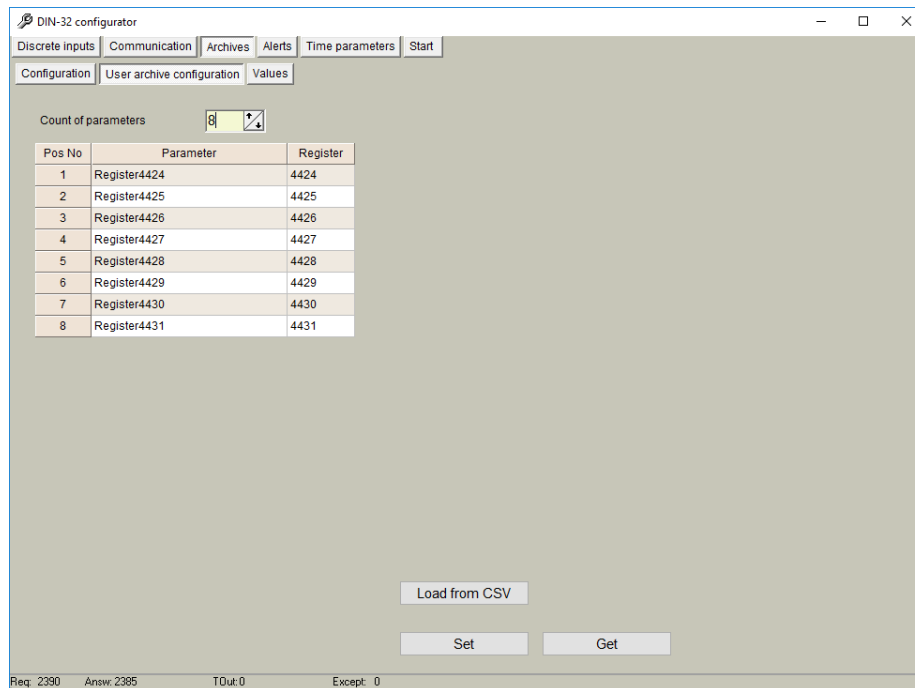


Figure 15. “Archives” tab. User archive configuration

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.

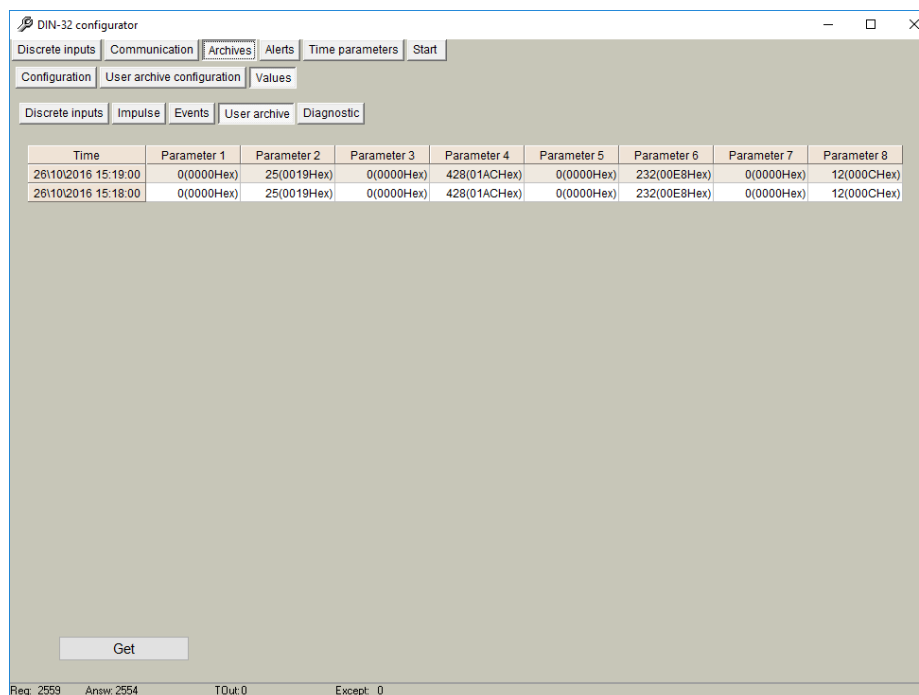


Figure 16. “Archives” tab. User archive current values

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

7 "Time parameters" tab

This tab enables users to set up time synchronization between DIN-32 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.
- "Clock synchronization": It must be enabled if synchronization between DIN-32 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.

The screenshot shows the 'DIN-32 configurator' window with the 'Time parameters' tab selected. The 'Clock synchronization' section is expanded, showing the following configuration:

- Enabled:**
- Type:** GPS SEL-2401, NTP Server
- IP of NTP server:** 217, 160, 20, 251
- Synchronization period (min.):** 2000
- Local time difference from UTC:** 2 h
- Automatically adjust clock for daylight saving changes:**
- Last attempt to synchronize:** 1998-01-01 00:00:00
- Successful synchronization time:** 1998-01-01 00:00:00

On the right side, the current times are displayed:

- PC time:** 26/10/2016 15:24:54
- Controller time:** 26/10/2016 15:24:53

A 'Set PC time' button is visible below the time display. At the bottom of the configuration area, there are 'Set' and 'Get' buttons. The status bar at the bottom shows: Req: 2805, Answ: 2800, TOut: 0, Except: 0.

Figure 17. "Time Parameters" tab. NTP server configuration

8 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.
- **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 and 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 ppheripheral 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 ppheripheral 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 ppheripheral devices (example memory stick's and etc.).
- **Socket** – is an endpoint of a bidirectional inter-process communication flow across an Internet Protocol-based computer network, such as the Internet.
- **Status** – device status indicating LED
- **Uoutput** – status of power for external device indicating LED
- **TX/RX** – data transfer/receive indicating LED
- **Central computer** – server or a computer, where data can be sent.

9 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.

10 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, 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.

11 Technical Data

11.1 Communication interfaces

Interfaces	Technical data
RS485	up to 1.2 km, max 32 transceivers, speed up to 38,4 Kbits/s
RS232	up to 15m, speed up to 38,4 Kbits/s
Ethernet	RJ45. Twisted pair Ethernet 10/100 Mb, up to 100m.
USB	Type B, version 2.0
Discrete INPUTS	"Dry contact"


11.2 Protection

Insulation voltage between power supply and second circuits	1000 V
Discrete INPUTS	Yes

11.3 Indication

Indication type	LED's
Indicated parameters	<ul style="list-style-type: none">• Each discrete input status• UART Transfer/Receive• Ethernet status, Transfer and Receive

11.4 Power supply

Power supply	9 ÷ 36 VDC 
Power consumption	Up to 10 VA

11.5 Construction

Mounting	DIN rail
Dimensions	197x128x50 mm
Enclosure protection	IP20

11.6 Working conditions

Working temperature	From - 25 °C to + 60°C
Storage temperature	From - 40 °C to + 60°C
Relative humidity	From 5 % to 95 % non-condensing

11.7 Safety parameters

Safety requirements	EN 61010-1:2002
Electromagnetic compatibility	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

11.8 Other parameters

Archive storage memory	2 MB
Configuration settings storage without power supply	More than 5 years
Real time clock	Yes
Firmware loading	Yes. Through RS232/USB and Ethernet.

Advantic Sistemas y Servicios S.L

Avda. Europa 14
28108 Alcobendas (Madrid)



www.advanticsys.com
info@advanticsys.com

+34 91 189 05 21