SCADA Interface

Teldat-Dm 758-I

Legal Notice

Warranty

This publication is subject to change.

Teldat offers no warranty whatsoever for information contained in this manual.

Teldat is not liable for any direct, indirect, collateral, consequential or any other damage connected to the delivery, supply or use of this manual.

Table of Contents

Chapter 1 Introduction

1.1 Introduction

This document describes the operating mode and configuration for the SCADA interface implemented by the Teldat Routers.

SCADA stands for "Supervisory Control And Data Acquisition". I.e. a system that carries out centralized measurements and control over industrial processes. A SCADA system consists of a CENTER host or MASTER (normally known as Master Terminal Unit or MTU); one or more data gathering devices or remote controls (normally known as Remote Terminal Units or RTUs); and a set of software and/or hardware tools used to monitor and control remotely located devices.

A SCADA system gathers the necessary information in order to perceive the status of the entire system it is controlling at all times, drawing attention to abnormal situations, generating analysis and executing possible control functions. SCADA determines which situations may be critical, presenting a logical and organized report. A SCADA system may be simple or extremely complex.

Normally, a distributed SCADA, where the information sources (RTUs) are distanced from the control center, uses protocols over leased lines or PSTN for interconnection. Examples of SCADA protocols are MODBUS and IEC 60870-5-101 (henceforth known as IEC-101).

Generally the connection of two remote devices or RTU's with SCADA applications is done using serial connections such as RS-232 or TCP/IP. Teldat Routers allow you to configure various types of communication protocols for their SCADA interfaces as well as different operating modes for each of them. The RS-485 interface is also used within an industrial environment. Some Teldat Routers are equipped with hardware for these types of interfaces.

As an example, MODBUS uses RTUs to communicate with the center part, in devices which must be monitored. These RTUs generally use an RS-232 compatible serial interface, which can be directly connected or connected through a MODEM.
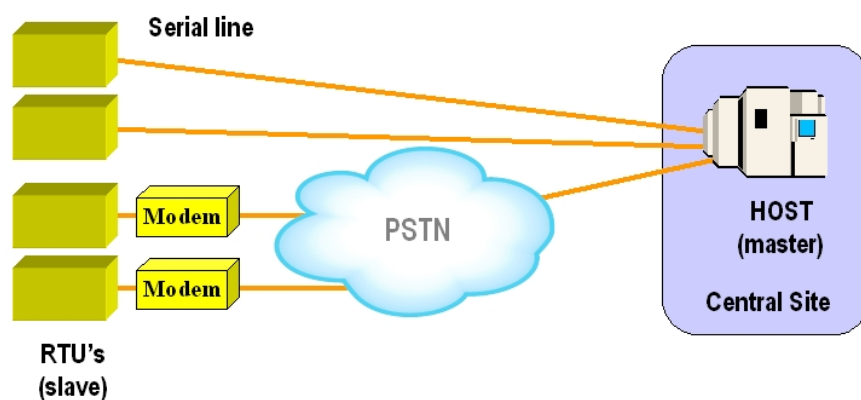
MODBUS is a MASTER-SLAVE asynchronous protocol, not connection oriented, where the "MASTER" center host queries the "SLAVE" remote unit, identifying it with an 8-bit addressing field. The remote unit responds to each query with the corresponding answer. MODBUS defines a series specific 'queries', known as functions, with the aim of acquiring data, controlling and diagnosing the remote units which shape the system.

The Teldat Routers are capable of sending information generated on SCADA interfaces over a variety of communication networks (GPRS, ADSL, ISDN, Frame Relay, etc.) encapsulating it in TCP/IP complying with the used protocol standard.

The following figure displays an example of SCADA connectivity (MODBUS) over a GPRS network.
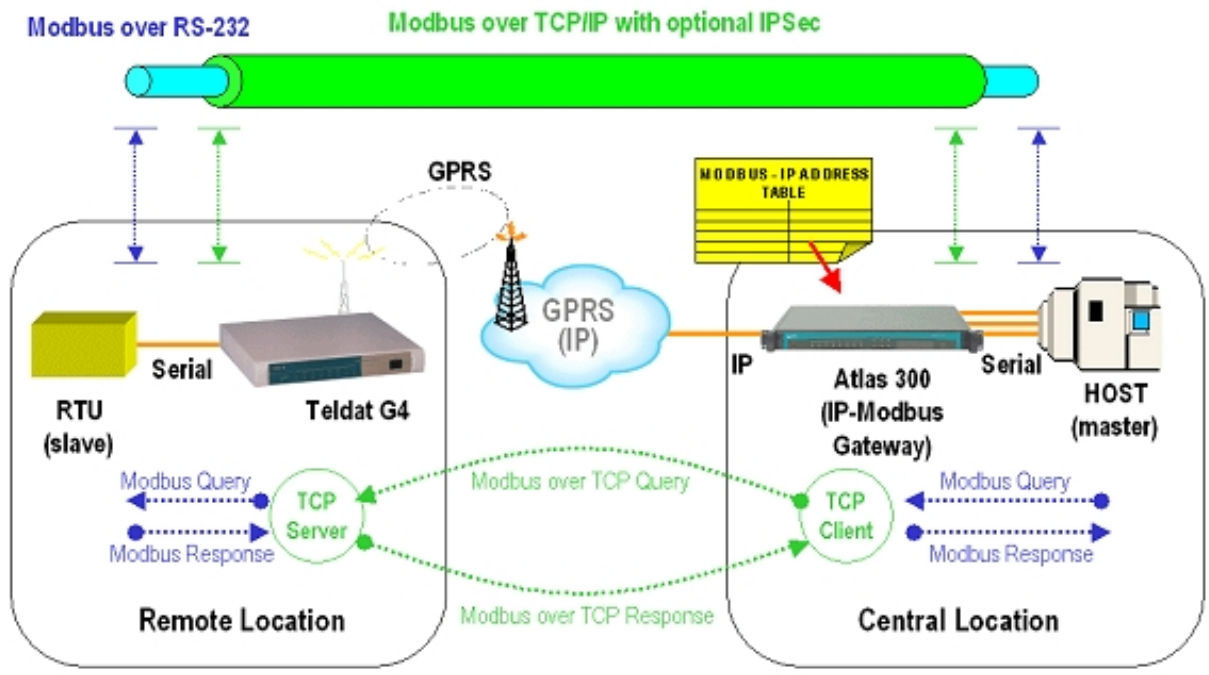
In IEC-101 mode, the Teldat Routers behave transparently, so that the traffic between both ends is monitored, however this does not implement any standard protocol over TCP/IP between the two routers.

The Teldat Router executes a SCADA interface role. Although normally the user scenario is similar to that indicated in the above figure, other types of environments exist where there is only one router with a SCADA interface, at only one end of the line, the other device being able to directly access via the SCADA TCP/IP format for the protocol being used.

In addition, the Teldat Router SCADA interface function allows connections or SCADA queries to be carried out from more than one center point. This therefore provides greater versatility for the SCADA system. The router itself avoids any collisions in the responses and routes them correctly

On the center side, it's possible to access a remote router that has more than one SCADA interface through the same TCP session. The center side router is in charge of establishing the necessary TCP sessions, i.e. if there is more than one remote device sharing the same IP address and TCP port, only one session is established with the remote router. The remote router decodes the received serial address and routes the message to the correct interface.

In certain types of communication networks, it might be necessary to limit the time in which the TCP sessions are maintained active. For example, in situations where GRPS is used as the TCP/IP transport method, depending on how the telephony carrier bills, it might be more economical to not maintain the TCP sessions active. To do this, the SCADA interface allows you to configure the time in which the router will maintain the TCP session after exchanging the last message. To make this more versatile, when the router is configured in CENTER mode, a time for each configured remote destination is set, with the aim of optimizing times for each specific case.

Some basic examples of connectivity have been given below. Scenarios can be created where some of these said scenarios are given simultaneously.
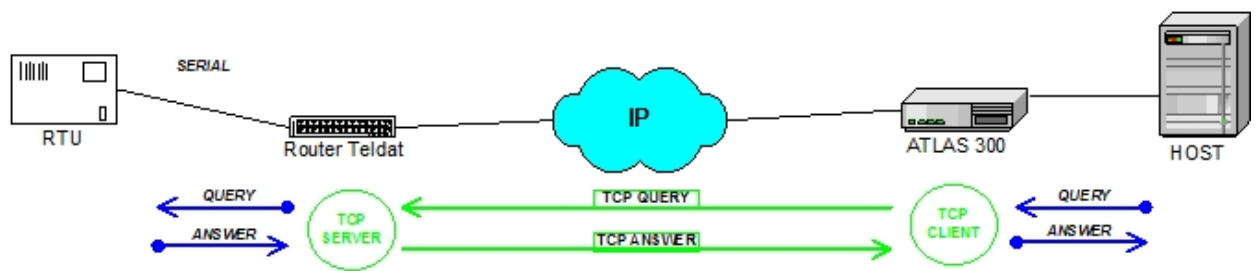
a) Basic Connection. A center host connects to a router with SCADA interface, configured in CENTER mode through a serial interface. The remote end consists of a serial RTU connected to a router with SCADA interface configured in REMOTE mode.

b) Remote router with two RTUs sharing a TCP port. A center host connects to a router with a SCADA interface, configured in CENTER mode through a serial interface. The remote end consists of two serial RTUs connected to a router with SCADA interfaces configured in REMOTE mode. In this case, the two SCADA interfaces from the remote

end have the same TCP listening port configured, therefore the queries generated by the center side are sent through a single TCP session. The remote router correctly routes both the queries and the responses. This configuration is not supported by balanced IEC-101 mode; you need to configure a different TCP port for each serial interface.
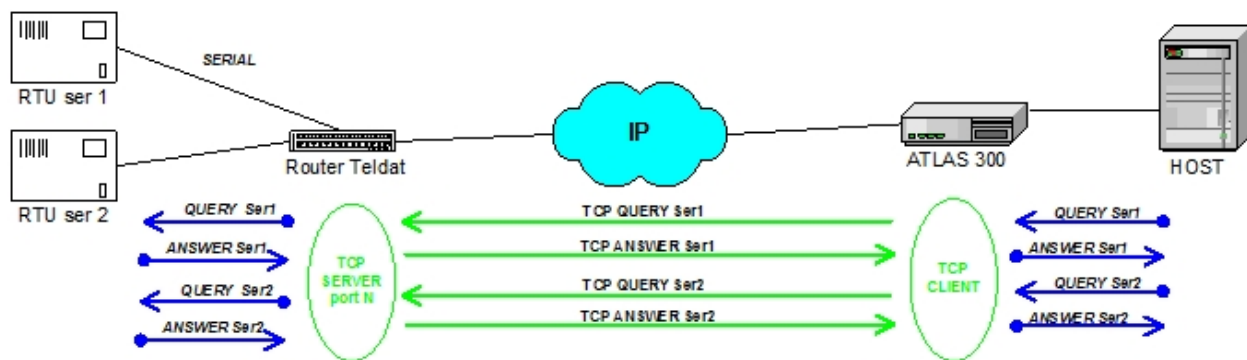
c) Remote router with two RTUs with different TCP ports. A center host connects to a router with a SCADA interface, configured in CENTER mode, through a serial interface. The remote end consists of two serial RTUs connected to a router with a SCADA interface configured in REMOTE mode. In this case, the two remote SCADA end interfaces have different TCP listening ports configured, therefore the queries generated by the center side are each sent through their corresponding TCP sessions.

d) There are two center hosts. Two center hosts each connect to a router with SCADA interface configured in CENTER mode, through a serial interface. The remote end consists of a serial RTU connected to a router with SCADA interface configured in REMOTE mode. In this case both the two center hosts can query the RTU, the remote router is responsible for avoiding query collisions and correctly routing the responses.

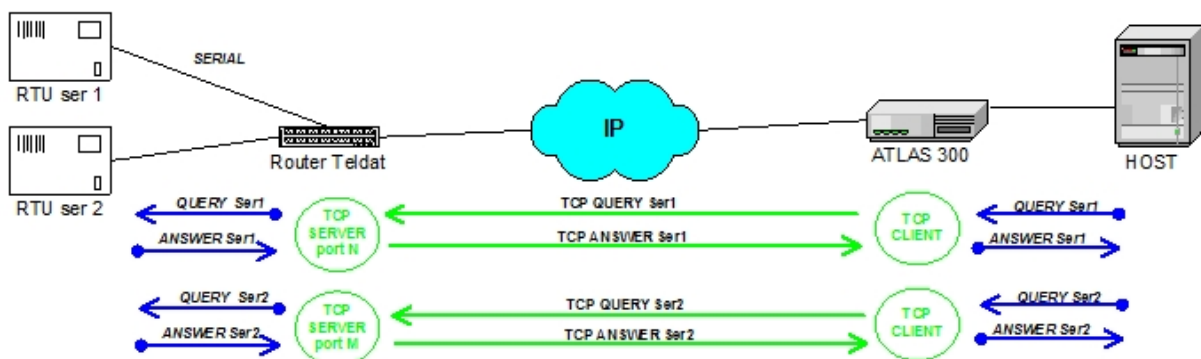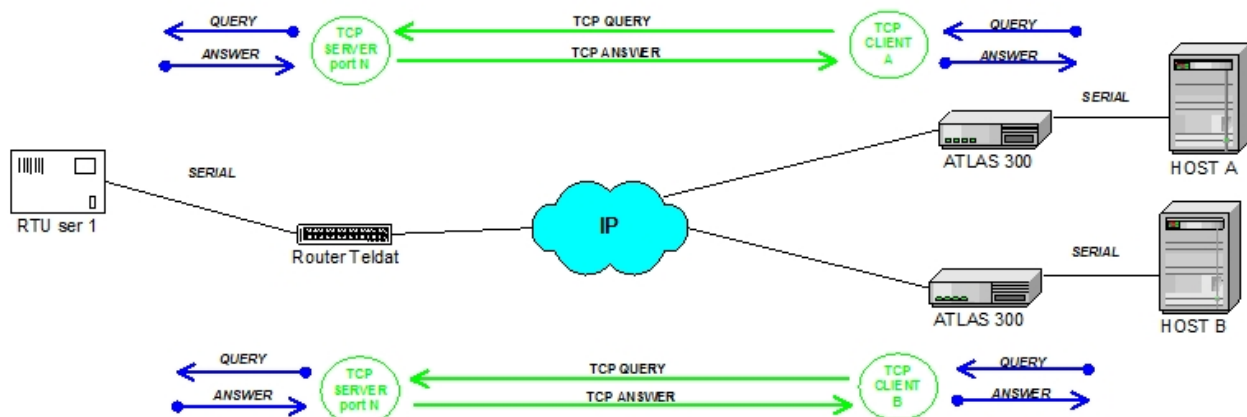e) The RTU allows TCP/IP connection. A center host connects to a router with a SCADA interface, configured in CENTER mode, through a serial interface. The remote end consists of an RTU with a SCADA TCP/IP interface. This configuration does not support IEC-101 protocol.
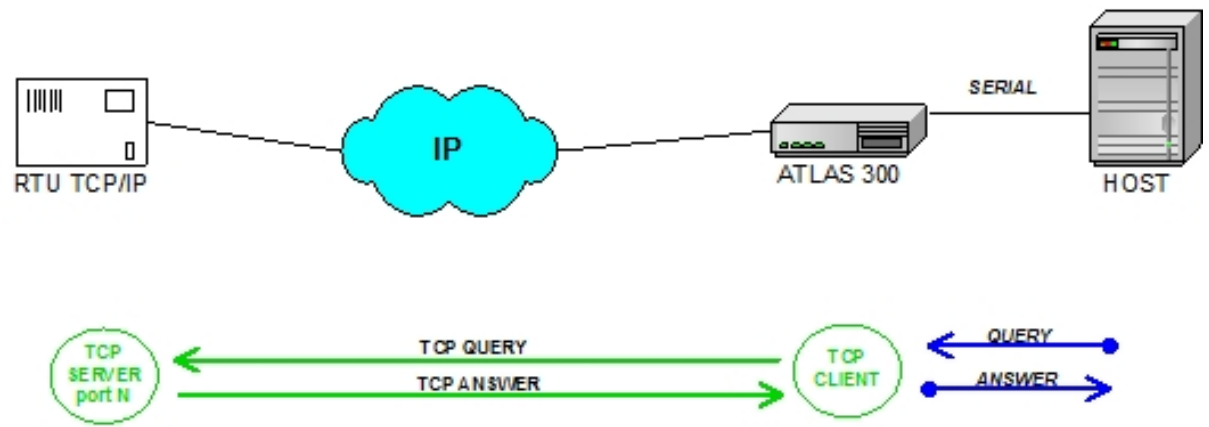
f) The host allows TCP/IP connection. A center host directly connects to an IP network, supporting the SCADA TCP/IP CLIENT protocol. The remote end consists of a serial RTU connected to a router with a SCADA interface configured in REMOTE mode. This configuration does not support IEC-101 protocol.
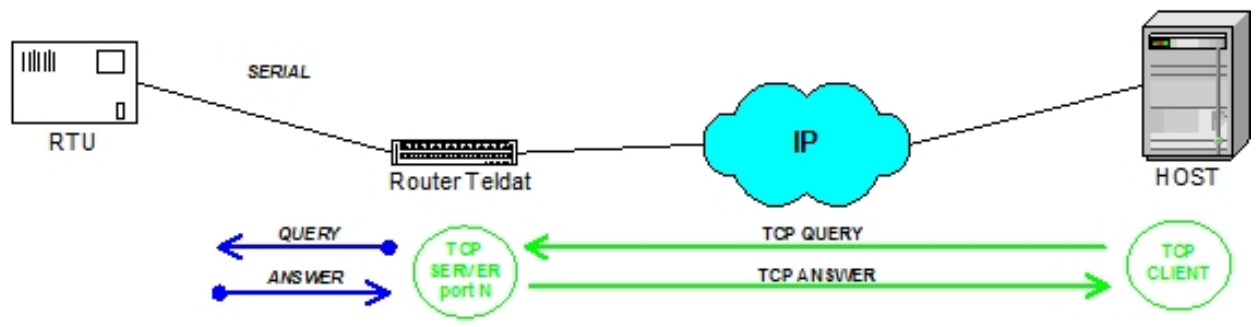
It is possible that some of the above described scenarios are not compatible with some specific operating modes. For example, the IEC 60870-5-101 balanced protocol does not allow more than one logical connection per TCP session.

1.2 SCADA Characteristics

The features supported by the SCADA Interface are as follows:

- Type of serial interface supported:

>> RS-232

>> RS-485

- Protocols over asynchronous serial line:

>> MODBUS-RTU

>> MODBUD-ASCII

>> IEC 60870-5-101 UNBALANCED

>> IEC 60870-5-101 BALANCED

- Supported character format:

>> start + 8 data + parity + 1 stop. For MODBUS-RTU and IEC 60870-5-101

>> start + 7 data + parity + 1 stop. For MODBUS-ASCII

- Types of supported parity:

>> EVEN

>> ODD

>> MARK

- SCADA serial modes:

CENTER (Behaves as a serial SLAVE and a TCP/IP client)

REMOTE (Behaves as a serial MASTER and as a TCP/IP server)

- SCADA TCP/IP Modes:

MODBUS CLIENT

MODBUS SERVER

IEC-101 PROPIETARY CLIENT

IEC-101 PROPIETARY SERVER

- Timers to control the TCP station time:

Maximum TCP session time without data traffic, inactivity.

Chapter 2 Configuring the SCADA Interface

2.1 SCADA NET

The SCADA interfaces permit connecting the Teldat devices' serial interfaces to devices that use SCADA protocols for Telecontrol.

Teldat devices supporting SCADA interfaces also mean that the Teldat serial interfaces can be configured as SCADA.

The SCADA interface feature provided by Teldat devices, allow you to establish different access modes to devices using SCADA protocols (MODBUS). To execute this function, the Teldat devices have the following functionalities incorporated:

- MODBUS SERIAL SLAVE
- MODBUS SERIAL MASTER
- MODBUS SERIAL RTU
- MODBUS SERIAL ASCII
- MODBUS TCP/IP CLIENT
- MODBUS TCP/IP SERVER

- IEC 60870-5-101 CENTER
- IEC 60870-5-101 REMOTE
- IEC 60870-5-101 UNBALANCED
- IEC 60870-5-101 BALANCED
- IEC-101 TCP/IP PROPIETARY CLIENT
- IEC-101 TCP/IP PROPIETARY SERVER

In practice, configuring the SCADA access mode (CENTER or REMOTE), the device will use the necessary functionalities.

The SCADA protocol can operate in two different modes, depending on the functionality required.

In CENTER mode, the Teldat device can connect to a MASTER serial device, broadcasting, via TCP its queries to the remote devices. To select the remote device IP address and destination port, the device allows you to configure registers with said information. You can configure a register through an available address in the protocol selected in the serial interface. The remote device can be either a SCADA TCP device (for MODBUS only) with the same protocol or act as a gateway in REMOTE mode.

In REMOTE mode, the Teldat device can connect to a SLAVE serial device, broadcasting, via TCP, its queries to remote devices. In this mode, the Teldat device is 'listening' through a configurable TCP port. When the queries arrive from a remote MASTER, the SCADA addressing information is checked and then this is sent to the SLAVE terminal, complying with said addressing criteria, connected to a serial port. The center device can act either as a gateway in CENTER mode or a SCADA TCP application that acts as MASTER (for MODBUS protocol only).

SCADA protocols running over a serial line are always RS-232, and do not implement any other type of interface.

2.1.1 SCADA Parameters

The SCADA interface basic configurable parameters are as follows:

- **Mode**: Mode used in the SCADA interface. By default this is configured in REMOTE mode.
- **Protocol**: Protocol used in the serial interface SCADA communication. The protocol selected in the serial interface determines the protocol used in TCP/IP. By default the protocol used in MODBUS-RTU.
- **Serial Interface SCADA Address** : This parameter identifies the SCADA address possessed by the serial device connected to the SCADA interface. This parameter is used to correctly route messages received via TCP to the destination serial interface. This parameter only makes sense when the selected mode is REMOTE.
- **Type of installed driver:** This parameter determines the behavior of data reception and transmission signals based on the type of driver installed. In cases regarding RS-485 it determines whether the communication is over 2 or 4 wires. This behavior is different for RS-232 drivers, 2-wire RS-485 drivers or 4-wire RS-485 drivers.
- **Termination Resistor** : This allows you to enable/disable the termination resistor in RS-485 interfaces. This option is available for those routers with hardware permitting a dynamic change in the resistor.
- **TCP Port:** Allows you to configure the TCP port executing the LISTEN function. The TCP queries from the remote

center TCP SCADA (CLIENT) should enter through this port. Default is standard MODBUS 502. This parameter only makes sense when the selected mode is REMOTE and in CENTER mode when the selected protocol is IEC-101 balanced.

- **Local IP Address:** Allows you to configure the SCADA interface local IP address. Through this, all the TCP messages take this IP address as source address.
- **Speed**: Serial line speed at which the SCADA interface transmits and receives. Default is 2400 bps.
- **Parity**: Allows you to configure the parity used in communications by the SCADA interface asynchronous serial line. Default is EVEN parity.
- **Serial Control Signal**: This parameter allows you to configure the mode in which the control signals are handled. These can either be ignored or attended: this allows the interface to be activated /deactivated.
- **Reception Buffer Size**: Allows you to configure the maximum size of the received messages whether it is through TCP or serial interface.
- **Transmission Buffer Size**: Allows you to configure the maximum size of the sent messages whether through TCP or serial interface.
- **IEC 60870-5 proprietary parameters:** Allows you to select a series of proprietary options for the IEC protocol. Among these options is the *link address* size, the CENTER IP address when this is configured in balanced mode, etc.
- **Remote Device Register**: This parameter allows you to configure the remote device IP address and destination port. The register is associated to the SCADA protocol address of the remote serial interface. Each remote address can only have a single register which defines the connectivity parameters. The registers are identified by the remote serial interface address.
- **Idle time in a TCP Session:** Allows you to configure the time in which a TCP session may remain established without messages being exchanged. Configuring a ZERO value means the timer will be disabled i.e. the session will permanently remain established.
- Track NSLA Advisor: Allows you to configure a track NSLA so some events occurring in the router can act over the output control signals.

2.2 Configuring the SCADA Parameters

Execute the following steps to enter the configuration process:

(1) At the (*) prompt, enter **PROCESS 4** or **P 4**. This will take you to the configuration prompt *Config>.*

```
*P 4
Config>
```

(2) Enter the **LIST DEVICES** command. This will obtain the interface numbers the device has assigned as SCADA.

```
Config>list devices

Interface       Con    Type of interface             CSR      CSR2   int
ethernet0/0     LAN1   Quicc Ethernet             FA200A00 FA203C00   5E


             serial0/0 WAN1 SCADA Interface FA200A20 FA203D00 5D


atm0/0          DSL1   ATM                        FA200A60 FA203F00   55
bri0/0          ISDN1  ISDN Basic Rate Int        FA200A40 FA203E00   5C
x25-node        ---    Router->Node                     0           0
Config>
```

(3) The interfaces that can be configured in SCADA mode are UART and SERIAL. In a device these are listed as "uartX/X" and "serialX/X". On booting the device, the required interfaces may not be configured as SCADA. To configure them, execute the **SET DATA-LINK** command as shown in the following example:

```
Config> list devices
Interface       Con    Type of interface             CSR      CSR2   int
ethernet0/0     LAN1   Quicc Ethernet             FA200A00 FA203C00   5E


             serial0/0 WAN1 X25                       FA200A20 FA203D00   5D
atm0/0          DSL1   ATM                        FA200A60 FA203F00   55
bri0/0          ISDN1  ISDN Basic Rate Int        FA200A40 FA203E00   5C
x25-node        ---    Router->Node                     0           0
Config>set data-link scada serial0/0

Config>list devices

Interface       Con    Type of interface             CSR      CSR2   int
ethernet0/0     LAN1   Quicc Ethernet             FA200A00 FA203C00   5E
```

```
             serial0/0 WAN1 SCADA Interface         FA200A20 FA203D00   5D
 atm0/0        DSL1   ATM                        FA200A60 FA203F00   55
 bri0/0        ISDN1  ISDN Basic Rate Int        FA200A40 FA203E00   5C
 x25-node      ---    Router->Node                     0           0
Config>
```

(4)   Subsequently, enter the **NETWORK** command followed by the name of the SCADA interface you wish to con-
       figure. This will take you to the indicated SCADA interface configuration menu. The character # is used in the
       generic example to indicate the name:

```
Config>NETWORK #

-- SCADA Interface Configuration --
SCADA-# Cfg>
```

   If, for example, the interface were serial0/0, the following would result:

```
Config>network serial0/0

-- SCADA Interface Configuration --
SCADA-serial0/0 Cfg>
```

There are certain commands which are common for all the device's interfaces, which do not appear in this manual.
These commands are described in the manual on configuring common interfaces (*Dm772-I Common Configurations
for Interfaces*).

The SCADA net configuration commands are numerated and described in this section. All the SCADA configuration
commands must be entered at the SCADA prompt (SCADA-# Cfg>).

Command	Function
? (HELP)	Lists the configuration commands or within a command lists any parameter associated to it.
DRIVER-TYPE	Select the type of driver installed.
IDLE-TIME	Configures the idle time to release a TCP session. (REMOTE mode only).
IEC-PARAM	Configures parameters particular to the IEC 60870-5 protocol.
LIST	Displays information on the SCADA interface configuration.
LOCAL-IP	Configures the local IP address.
LOCAL-PORT	Configures the local TCP port.
MODE	Configures the SCADA mode (REMOTE/CENTER).
NO	Negates a command or sets its default value.
ORDER-TIMEOUT	Configures the maximum wait time for a response to a query.
PARITY	Configures the parity to be used in the serial interface.
PROTOCOL	Configures the SCADA protocol to be used.
REMOTE-REGISTER	Configures a register to access a remote device (CENTER mode only).
RX-BUFFER-SIZE	Configures the buffer size in reception sense.
SERIAL-ADDRESS	Configures the serial interface SCADA address (REMOTE mode only).
SIGNAL-CONTROL	Enables the use of the SCADA serial Interface input control signal.
SIGNAL-OUTPUT-CONTROL	Enables the use of the output control signal in the SCADA serial interface.
SIGNAL-TIMEOUT	Input control signal timeout time.
SPEED	Configures the serial interface speed.
TCP-INIT	Configures the TCP behavior with outgoing sessions.
TERMINATION-RESISTOR	Enables/disables the termination resistor in RS-485 interfaces.
TX-BUFFER-SIZE	Configures the buffer size in transmission sense.
EXIT	Returns to the previous prompt.

2.2.1 ? (HELP)

The **?** (HELP) command lists all the available commands. You can enter a **?** after a specific command name to ob-
tain its options.

Syntax:

```
SCADA-# Cfg>?
```

Example:

```
SCADA-serial0/0 Cfg>?
  description            Enter interface description
  driver-type           Select driver type installed
  idle-time             Configure TCP session idle time(remote site only)
  iec-param             Configure IEC specific paramaters
  list                  List configuration
  local-ip              Configure local IP address
  local-port            Configure TCP local port(remote)
  mode                  Configure SCADA gateway mode(remote/center)
  no                    Negate a command or set its defaults
  order-timeout         Configure max order time
  parity                Configure interface parity
  protocol              Configure SCADA protocol
  remote-register       Configure remote register
  rx-buffer-size        Configure reception sense buffers size
  serial-address        Configure local serial address
  shutdown              Change state to administratively down
  signal-control        Enable serial input control signal
  signal-output-control Enable serial output control signal handle
  signal-timeout        Serial input control signal timeout
  speed                 Configure interface speed
  tcp-init              Enable TCP to init session
  termination-resistor  Set a resistor at the end of the wires RS-485
  tx-buffer-size        Configure transmission sense buffers size
  update                Update a level indicator
  exit
SCADA-serial0/0 Cfg>
```

2.2.2 DRIVER-TYPE

This parameter determines the behavior of the transmission and data reception signals based on the type of installed driver. The types of supported drivers are as follows:

RS-232

RS-485 2 wire

RS-485 4 wire

This behavior is different for RS-232, RS-485 2 wire or for RS-485 4 wire drivers. You must remember that this parameter does not configure the type of hardware driver; it only adapts the software so said driver operates correctly. There are two types of hardware drivers: RS-232 and RS-485. In cases of RS-485, this parameter determines whether the communication is over 2 or 4 wires. The driver hardware is the one the device has installed, which can be changed in some devices and not in others. If you try and modify this parameter in devices that do not admit different drivers to be installed, the device returns an error message.

The default value for this parameter depends on the type of device and consequently configures automatically. E.g. in devices with UART interfaces, the default value is RS-232 and in devices which incorporate SERIAL interfaces, default is RS-485 2 wire.

Syntax:

```
SCADA-# Cfg> driver-type ?
  RS232      RS-232 driver installed
  RS485-2W   RS-485 2 wires driver installed
  RS485-4W   RS-485 4 wires driver installed
```

Example:

```
SCADA-serial0/0 Cfg> driver-type rs485-2w
SCADA-serial0/0 Cfg>
```

2.2.3 IDLE-TIME

The aim of this parameter is to control the time the TCP sessions remain established without data being exchanged. The configured value indicates the time the TCP session remains established after the exchange of the last message through this. Configuring a ZERO value means the timer will be disabled i.e. the TCP session is permanently established. This parameter is only configurable when you select the REMOTE mode. If the selected mode is CENTER, an error message will appear on entering this command. This parameter can be very useful to optimize resources regarding transport when connection time is billable. The minimum time configurable in the timer is 1 second. Default is zero.

Syntax:

```
SCADA-# Cfg> idle-time <time>
```

Example:

```
SCADA-serial0/0 Cfg>idle-time 33
SCADA-serial0/0 Cfg>
```

2.2.4 IEC-PARAM

This command allows you to configure IEC-101 proprietary parameters. Depending on the configured operating mode, REMOTE or CENTER, you can enter various commands.

In CENTER mode you can configure the following parameters:

IEC101-BALANCED	Selects IEC-101 in balanced mode.
IEC101-LINK-ADDRESS	Configures the IEC interface *link address* length.
NO	Negates a command.
LISTEN-TCP-FORCE	Enables listen TCP in unbalanced CENTER mode.

In REMOTE mode you can figure the following parameters:

ADDRESS-CHECKING	Enables the IEC-101 destination address control.
CENTER-IP-ADDRESS	Configures the IP address for the CENTER end for balanced mode.
CENTER-TCP-PORT	Configures the TCP port for the CENTER end for balanced mode.
IEC101-BALANCED	Selects IEC-101 in balanced mode.
IEC101-LINK-ADDRESS	Configures the IEC interface *link address* length.
NO	Negates a command.

If you try and enter a command in an unsupported mode, the device will return an error message.

Syntax:

```
SCADA-serial0/0 Cfg>iec-param <command> <value>
  address-checking       Enable IEC link address checking
  center-ip-address      Configure center IP address for balanced mode
  center-tcp-port        Configure center TCP port for balanced mode
  iec101-balanced        Select IEC-101 balanced mode
  iec101-link-address    Configure IEC-101 link address length
  listen-tcp-force       Enable forced TCP listen on CENTER site for no
                         balanced mode
  no                     Negates a command or sets its defaults
```

Example:

```
SCADA-serial0/0 Cfg>iec-param iec101-link-address 1
SCADA-serial0/0 Cfg>
```

The IEC protocol proprietary parameters which can be configured are described below:

2.2.4.1 IEC-PARAM ADDRESS-CHECKING

Enables the IEC-101 address verification for all the messages reaching the associated interface. Messages whose addresses do not coincide with the programmed SCADA address (serial-address) are dropped. This parameter only makes sense when the IEC-101 REMOTE protocol is configured. By default this mode is enabled. To disable it, enter the command preceded by the option **NO**.

Syntax:

```
SCADA-serial0/0 Cfg>iec-param address-checking
```

Example:

```
SCADA-serial0/0 Cfg>iec-param address-checking
SCADA-serial0/0 Cfg>
SCADA-serial0/0 Cfg>iec-param no address-checking
```

2.2.4.2 IEC-PARAM CENTER-IP-ADDRESS

Configures the IP address where the CENTER device is located. This parameter only makes sense when you configure the IEC-101 REMOTE protocol in BALANCED mode. This allows the remote device to establish a TCP session when it needs to send an event to the center side.

Syntax:

```
SCADA-serial0/0 Cfg>iec-param center-ip-address <A.B.C.D>
```

Example:

```
SCADA-serial0/0 Cfg>iec-param center-ip-address 123.33.45.66
SCADA-serial0/0 Cfg>
```

2.2.4.3 IEC-PARAM CENTER-TCP-PORT

Configures the TCP port used by the CENTER device to LISTEN. This parameter only makes sense when you configure the IEC-101 REMOTE protocol in BALANCED mode. This allows the remote device to establish a TCP session when it needs to send an event to the center side. It is essential that this parameter is correctly configured in order for the system to operate properly. This can take values between 0 and 65535.

Syntax:

```
SCADA-serial0/0 Cfg>iec-param center-tcp-port <port>
```

Example:

```
SCADA-serial0/0 Cfg>iec-param center-tcp-port 3456
SCADA-serial0/0 Cfg>
```

2.2.4.4 IEC-PARAM IEC101-BALANCED

Allows you to enable BALANCED mode in the SCADA interface configured as IEC-101. Default is not enabled. To disable this, enter the same command preceded by the option **NO**.

Syntax:

```
SCADA-serial0/0 Cfg>iec-param iec101-balanced
```

Example:

```
SCADA-serial0/0 Cfg>iec-param iec101-balanced
SCADA-serial0/0 Cfg>
```

2.2.4.5 IEC-PARAM IEC101-LINK-ADDRESS

Configures the size of the *link address* field to be used in IEC-101 by the serial interface. The size of the address field can be 1 or 2 bytes. Default is 1.

In cases where more than one SCADA interface has been configured in IEC mode, in order to share connection TCP session with the remote end, both interfaces must have the same sized *link address* configured.

Syntax:

```
SCADA-serial0/0 Cfg>iec-param iec101-link-address <value>
```

Example:

```
SCADA-serial0/0 Cfg>iec-param iec101-link-address 2
SCADA-serial0/0 Cfg>
```

2.2.4.6 IEC-PARAM LISTEN-TCP-FORCE

Allows you to enable the listening for the TCP port programmed in the associated SCADA interface for unbalanced mode. This parameter only makes sense when the IEC-101 center protocol is configured. Normally, in unbalanced mode, the device configured in CENTER mode launches TCP sessions to the REMOTEs, but in some scenarios it can be useful to not only launch TCP sessions but also accept incoming sessions. By default listening for the TCP port in unbalanced mode is not enabled.

Syntax:

```
SCADA-serial0/0 Cfg>iec-param listen-tcp-force
```

Example:

```
SCADA-serial0/0 Cfg>iec-param listen-tcp-force
SCADA-serial0/0 Cfg>
```

2.2.4.7 IEC-PARAM NO

Use this command to disable functionalities or to set the default values for some parameters.

Syntax:

```
SCADA-serial0/0 Cfg>iec-param no ?
  address-checking    Disable IEC link address checking
  iec101-balanced     Select IEC-101 unbalanced mode
  listen-tcp-force    Disable forced TCP listen on CENTER site for no balanced
```

2.2.4.7.1 IEC-PARAM NO ADDRESS-CHECKING

Disables the verification of the IEC-101 address in all the messages reaching the associated interface. This parameter only makes sense when the IEC-101 REMOTE protocol is configured. If you try and configure this in CENTER mode, the device will return an error message.

Syntax:

```
SCADA-serial0/0 Cfg>iec-param no address-checking
```

Example:

```
SCADA-serial0/0 Cfg>iec-param no address-checking
SCADA-serial0/0 Cfg>
```

2.2.4.7.2 IEC-PARAM NO IEC101-BALANCED

Allows you to select the UNBALANCED mode.

Syntax:

```
SCADA-serial0/0 Cfg>iec-param no iec101-balanced
```

Example:

```
SCADA-serial0/0 Cfg>iec-param no iec101-balanced
SCADA-serial0/0 Cfg>
```

2.2.4.7.3 IEC-PARAM NO LISTEN-TCP-FORCE

Disables the TCP port listening programmed in the associated SCADA interface for unbalanced mode.

Syntax:

```
SCADA-serial0/0 Cfg>iec-param no listen-tcp-force
```

Example:

```
SCADA-serial0/0 Cfg>iec-param no listen-tcp-force
SCADA-serial0/0 Cfg>
```

2.2.5 LIST

Use the **LIST** command in the SCADA configuration process to view the interface general parameters (serial interface speed, mode used, etc.).

Depending on how the mode is configured, CENTER or REMOTE, different parameters will be presented.

Syntax:

```
SCADA-serial0/0 Cfg>list ?
  all              List all SCADA configuration
  global           List global SCADA configuration
  remote-register    List a remote register configuration
SCADA-serial0/0 Cfg>
```

2.2.5.1 LIST ALL

Lists all the SCADA interface parameters. Depending on the protocol and the configured mode, some parameters will appear or not i.e. those parameters that make sense for the current configuration appear but not others.

Examples:

CENTER mode:

```
SCADA-serial0/0 Cfg>list all
Transfer Mode            : center
Protocol Type            : iec101-trans
Local TCP port           : 502
Driver type installed     : RS485-2W
Local IP address         : 0.0.0.0
Link speed               : 2400 (bit/sec)
Link parity              : even
Ser. Line Signal Control   : enabled
Ser. Line Signal timeout   : 20 (sec)
Max order time           : 5
Rx buffers size          : 512
Tx buffers size          : 512
TCP init session         : Enable
IEC-101 address size      : 2
IEC-101 balanced mode     : no
IEC-101 tcp listen forced  : yes
NSLA track advisor        : Enable ID: 2
------------------------------------------------------------------------------
Name..............: sdfer
Serial address....: 1
Remote IP address.: 172.24.78.99
Remote TCP port...: 502
TCP connect. Time.: 4
TCP Idle Time.....: 0
------------------------------------------------------------------------------
SCADA-serial0/0 Cfg>
```

REMOTE mode:

```
SCADA-serial0/0 Cfg>list all
Transfer Mode            : remote
Protocol Type            : iec101-trans
Local SCADA serial address : 1
Local TCP port           : 502
Driver type installed     : RS485-2W
Local IP address         : 0.0.0.0
Link speed               : 2400 (bit/sec)
Link parity              : even
Ser. Line Signal Control   : enabled
Ser. Line Signal timeout   : 20 (sec)
Max order time           : 5
Rx buffers size          : 512
Tx buffers size          : 512
TCP session Idle time     : 0
IEC-101 address size      : 2
IEC-101 balanced mode     : no
IEC central site ip addr.  : 0.0.0.0
IEC central site TCP port  : 502
IEC-101 address checking   : yes
```

```
NSLA track advisor      : Enable ID: 2
SCADA-serial0/0 Cfg>
```

2.2.5.2 LIST GLOBAL

Lists all the SCADA interface global parameters.

Examples:

CENTER mode:

```
SCADA-serial0/0 Cfg>list global
Transfer Mode            : center
Protocol Type            : iec101-trans
Local TCP port           : 502
Driver type installed    : RS485-2W
Local IP address         : 0.0.0.0
Link speed               : 2400 (bit/sec)
Link parity              : even
Ser. Line Signal Control   : enabled
Ser. Line Signal timeout   : 20 (sec)
Max order time           : 5
Rx buffers size          : 512
Tx buffers size          : 512
TCP init session         : Enable
IEC-101 address size     : 2
IEC-101 balanced mode      : no
IEC-101 tcp listen forced  : yes
NSLA track advisor       : Enable ID: 2
SCADA-serial0/0 Cfg>
```

REMOTE mode:

```
SCADA-serial0/0 Cfg>list global
Transfer Mode            : remote
Protocol Type            : iec101-trans
Local SCADA serial address : 1
Local TCP port           : 502
Driver type installed    : RS485-2W
Local IP address         : 0.0.0.0
Link speed               : 2400 (bit/sec)
Link parity              : even
Ser. Line Signal Control   : enabled
Ser. Line Signal timeout   : 20 (sec)
Max order time           : 5
Rx buffers size          : 512
Tx buffers size          : 512
TCP session Idle time      : 0
IEC-101 address size     : 2
IEC-101 balanced mode      : no
IEC central site ip addr.  : 0.0.0.0
IEC central site TCP port  : 502
IEC-101 address checking   : yes
NSLA track advisor       : Enable ID: 2
SCADA-serial0/0 Cfg>
```

2.2.5.3 LIST REMOTE-REGISTER

Lists a configuration register to access a remote terminal. This command must be entered followed by the number of the register you wish to view. This command only makes sense when the configured mode is CENTER.

Examples:

```
SCADA-serial0/0 Cfg>list remote-register 1
-------------------------------------------------------------------------------
Name...............: sdfer
Serial address....: 1
Remote IP address.: 172.24.78.99
Remote TCP port...: 502
```

```
TCP connect. Time.: 4
TCP Idle Time.....: 0
---------------------------------------------------------------------------
SCADA-serial0/0 Cfg>
```

2.2.6 LOCAL-IP

This parameter allows you to configure the local IP address which is sent to the other end on establishing a TCP session. Value 0.0.0.0 indicates this parameter is not used automatically and the used IP address is assigned to the router.

You need to remember that in cases where more than one interface is configured in CENTER mode with IEC-101 BALANCED, you must configure distinct local IP addresses for the interfaces. Contrariwise, you can get ambiguity problems when interpreting the route for the messages received from the remote end.

Syntax:

```
SCADA-# Cfg>local-ip  <a.b.c.d>
```

Example:

```
SCADA-serial0/0 Cfg>local-ip 172.23.44.55
SCADA-serial0/0 Cfg>
```

2.2.7 LOCAL-PORT

This parameter is used to configure the TCP port executing 'LISTEN'. The TCP queries from the SCADA TCP center (CLIENT) end must enter through this port. Default is standard MODBUS 502. This parameter only makes sense when the mode selected is REMOTE and where CENTER is selected with IEC-101 protocol in BALANCED mode.

In cases where the device has more than one SCADA interface configured in REMOTE, said interfaces can share the same LISTEN TCP port provided that the protocol being used by both is the same e.g. MODBUS. The IEC-101 in BALANCED mode needs each SCADA interface configured in the device to have a different TCP port.

The range of valid values is from 0 to 65535.

Syntax:

```
SCADA -# Cfg>local-port  <port>
```

Example:

```
SCADA-serial0/0 Cfg>local-port 2323
SCADA-serial0/0 Cfg>local-port
```

2.2.8 MODE

This parameter allows you to select the mode to be used in the SCADA interface. Two modes can be configured:

• CENTER
• REMOTE

CENTER mode allows a MASTER serial SCADA device to be connected to various remote SLAVE devices through TCP sessions. For this, the master device connects to a Teldat Router through the serial interface configured as SCADA.

REMOTE mode allows you to receive remote queries via TCP from MASTER devices, either through a TCP/IP gateway or directly through a SCADA application which supports TCP/IP connections (for MODBUS mode only). The queries are addressed to the SCADA interface whose address included in the TCP message coincides with the interface address associated to the serial device.

Depending on the selected protocol, the transmission of data via TCP is carried out complying with said protocol, but complying with the TCP specification.

Default is REMOTE mode.

Syntax:

```
SCADA-# Cfg>mode  <mode>
```

Example:

```
SCADA-serial0/0 Cfg>mode remote
SCADA-serial0/0 Cfg>
```

2.2.9 NO

This command is used to disable functionalities or to set the default values in some parameters.

Syntax:

```
SCADA-serial0/0 Cfg>no ?
  remote-register     Delete remote register
  signal-control      Disable SCADA serial interface control signal
  signal-output-control    Disable serial output control signal handle
  tcp-init                 Disable TCP to init session
  termination-resistor     Set a resistor at the end of the wires RS-485
```

2.2.9.1 NO REMOTE-REGISTER

Eliminates the remote destination configuration register. To do this, introduce the address identifier for said register.

Example:

```
SCADA-serial0/0 Cfg>no remote-register 1
SCADA-serial0/0 Cfg>
```

2.2.9.2 NO SIGNAL-CONTROL

This command is used to disable the input control signal handling in the SCADA serial interface.

Example:

```
SCADA-serial0/0 Cfg>no signal-control
SCADA-serial0/0 Cfg>
```

2.2.9.3 NO SIGNAL-OUTPUT-CONTROL

This command is used to disable the output control signal handling for the SCADA serial interface. When this option is selected, the output control signals remain permanently active.

Example:

```
SCADA-serial0/0 Cfg>no signal-output-control
SCADA-serial0/0 Cfg>
```

2.2.9.4 NO TCP-INIT

This command is used to disable the launching of outgoing TCP sessions. This command is only logical when the device is configured in CENTER mode.

Example:

```
SCADA-serial0/0 Cfg>no tcp-init
SCADA-serial0/0 Cfg>
```

2.2.9.5 NO TERMINATION-RESISTOR

This command disables the termination resistor in the RS-485 interfaces that control the resistor.

Example:

```
SCADA-uartX/X Cfg>no termination-resistor
SCADA-uartX/X Cfg>
```

2.2.10 ORDER-TIMEOUT

This parameter is useful in SCADA protocols where message exchange is executed through a MASTER/SLAVE scheme. In this type of exchange, the master always takes the initiative and waits for the response from the slave for a period of time. If the response from the slave arrives outside of the stipulated time, the message is dropped. Using this scheme the slave has a certain period of time to send responses, all responses arriving outside of this must not be sent to the SCADA management station. Therefore each query made should be followed by a response from the remote device being queried and from that device alone.

Contrariwise, there are SCADA protocols where the process of sending messages and the response reception can be disordered. In this type of protocol, the center side can execute the queries it wants without waiting for the responses, subsequently the responses can arrive in a disorderly fashion.

The Teldat Router can control message exchange thus ensuring this follows the MASTER/SLAVE protocol. I.e. a period of time is waited to receive a response to a query, if the response arrives when this period has timed out, it will be dropped. Through this parameter you can configure the response wait period. You need to bear in mind that during this period new queries cannot be admitted and all responses except the one expected will be dropped.

There are certain protocols oriented to the MASTER/SLAVE processes, however the user can decide not to use them in this way. In these cases, where it is most likely that this action will not comply with the standard, this is the sole responsibility of the user who decides to carry this out. This occurs in the MODBUS protocol for example.

The aim of this parameter is to control the time, both for the device configured as CENTER as well as that configured as REMOTE, waited for the responses to the sent messages. In the case of the CENTER side, this waits for the responses from the remote stations which arrive via TCP. The REMOTE side waits for responses which arrive through the remote terminal device serial interface.

0 value disables the timer, i.e. control over the MASTER/SLAVE process is not carried out and consequently the response messages may arrive disordered. This timer only affects the protocols whose process is MASTER/SLAVE.

This parameter is particularly interesting for the MODBUS and IEC, in balanced mode.

A 0 value disables the timer control for message exchange; consequently the device behaves in transparent mode from the point of view of the SCADA protocol exchange times. Owing to this, no entry appears in the OK or WRONG transfer lists in the statistics.

You need to remember that in IEC-101 BALANCED protocol, this timer must be greater than zero.

The minimum unit configurable in the timer is 1 second. Default is 5.

Syntax:

```
SCADA-# Cfg>order-timeout <time>
```

Example:

```
SCADA-serial0/0 Cfg>order-timeout 2 2
SCADA-serial0/0 Cfg>
```

2.2.11 PARITY

This command allows you to configure the type of parity which should be applied to the asynchronous characters exchanged by the SCADA interface. The types of parity that can be configured are:

• EVEN

• ODD

• MARK

• NONE

Configured parity default is EVEN.

Syntax:

```
SCADA-serial0/0 Cfg>parity ?
  even    Parity EVEN
  odd     Parity ODD
  mark    Parity MARK
  none    Parity NONE
```

Example:

```
SCADA-serial0/0 Cfg>parity even
SCADA-serial0/0 Cfg>
```

2.2.12 PROTOCOL

This command allows you to configure the protocol to be used by the SCADA interface. The SCADA protocols that can be configured are as follows:

- MODBUS-RTU
- MODBUS-ASCII
- IEC101-TRANS

The protocol configured by default is MODBUS-RTU.

Syntax:

```
SCADA-serial0/0 Cfg>protocol ?
  modbus-ascii    MODBUS, ASCII serial
  modbus-rtu      MODBUS, RTU serial
  iec101-trans    IEC-101 transparent
```

Example:

```
SCADA-serial0/0 Cfg>protocol modbus-ascii
SCADA-serial0/0 Cfg>
```

2.2.13 REMOTE-REGISTER

When the router is configured in CENTER mode, the necessary addressing parameters must be indicated so the messages can be transmitted to the REMOTE terminals. The addressing is carried out based on the SCADA devices' serial addresses. Therefore, each serial address the center device can consult must have a register with the TCP/IP access information in the router. These registers are essential so the device in CENTER mode can execute its tasks.

This parameter can only be configured when CENTER mode is selected. If REMOTE has been selected as the mode, when this command is entered, an error message is displayed.

Depending on the selected protocol, you can enter a register number. E.g. for the MODBUS protocol, register address values are permitted that can vary between 1 and 247.

To create a register, simply enter any command from those permitted.

Syntax:

```
SCADA-serial0/0 Cfg>remote-register <serial address> <command> <value>
  name               Configure remote register name
  ip-address         Configure remote IP address
  tcp-port           Configure remote TCP port
  idle-time          Configure TCP session idle time
  connection-time    Configure TCP session connection time
```

Example:

```
SCADA-serial0/0 Cfg>remote-register 33 ip-address 23.233.33.22
SCADA-serial0/0 Cfg>
```

The **LIST REMOTE-REGISTER** command allows you to view the contents of a register.

Example:

```
SCADA-serial0/0 Cfg>list remote-register 33
---------------------------------------------------------------------------
Name..............:
Serial address....: 33
Remote IP address.: 23.233.33.22
Remote TCP port...: 502
TCP connect. Time.: 4
TCP Idle Time.....: 0
---------------------------------------------------------------------------
SCADA-serial0/0 Cfg>
```

To eliminate a register, enter the command **NO** followed by **REMOTE-REGISTER** and the number of the register you wish to remove.

Example:

```
SCADA-serial0/0 Cfg>no remote-register 33
```

The parameters that can be configured in each register are explained below:

2.2.13.1 NAME

Allows you to configure a name for the register. This parameter is useful to identify the location of the remote device.

Syntax:

```
SCADA-serial0/0 Cfg>remote-register <serial_address> name <name>
```

Example:

```
SCADA-serial0/0 Cfg>remote-register 33 name ALS22
SCADA-serial0/0 Cfg>
```

2.2.13.2 IP-ADDRESS

Configures the IP address for the remote device. This parameter is essential in order to correctly access the remote device.

Syntax:

```
SCADA-serial0/0 Cfg>remote-register <serial_address> ip-address <A.B.C.D>
```

Example:

```
SCADA-serial0/0 Cfg>remote-register 33 ip-address 123.33.45.66
SCADA-serial0/0 Cfg>
```

2.2.13.3 TCP-PORT

Configures the TCP port where the remote device is executing LISTEN. This parameter must be correctly configured in order for the system to function properly. The SCADA protocols normally have a predefined TCP port for this. This can take values between 0 and 65535.

Syntax:

```
SCADA-serial0/0 Cfg>remote-register <serial_address> tcp-port <port>
```

Example:

```
SCADA-serial0/0 Cfg>remote-register 33 tcp-port 3456
SCADA-serial0/0 Cfg>
```

2.2.13.4 IDLE-TIME

The aim of this parameter is to control the time the TCP sessions remain established without data being exchanged. The configured value indicates the time the TCP session will remain established after the last message exchange during said session. Configuring a ZERO value means the timer is disabled i.e. the TCP session remains permanently established. This parameter can be very useful to optimize resources regarding transmission where connection time is billable. The minimum time configurable in the timer is 1 second. Default is zero. This can take values between 1 second and 2 days.

Syntax:

```
SCADA-serial0/0 Cfg>remote-register <serial_address> idle-time <time>
```

Example:

```
SCADA-serial0/0 Cfg>remote-register 33 idle-time 60
SCADA-serial0/0 Cfg>
```

2.2.13.5 CONNECTION-TIME

The aim of this parameter is to limit the time waited to establish the TCP session. The minimum time configurable in the timer is 1 second. Default is 4 seconds. This can take values between 1 and 300 seconds.

Syntax:

```
SCADA-serial0/0 Cfg>remote-register <serial_address> connection-time <time>
```

Example:

```
SCADA-serial0/0 Cfg>remote-register 33 connection-time 10
SCADA-serial0/0 Cfg>
```

2.2.14 RX-BUFFER_SIZE

Configures the maximum size of the data field for the messages that can be received through TCP. The permitted values are from 0 to 2048 octets. Default is 512. This parameter defines the maximum size of the buffers used in the reception path, i.e. TCP reception buffer and transmission buffer towards the serial interface.

Syntax:

```
SCADA-serial0/0 Cfg>rx-buffer-size <buff-size>
```

Example:

```
SCADA-serial0/0 Cfg>rx-buffer-size 1024
SCADA-serial0/0 Cfg>
```

2.2.15 SERIAL-ADDRESS

Allows you to configure the address that the protocol being used (MODBUS, IEC-101, etc.,) assigns to the RTU connected to the serial interface. This parameter only makes sense when this is configured in REMOTE mode. If you try and configure this in CENTER mode, the device returns an error message.

Syntax:

```
SCADA-serial0/0 Cfg>serial-address <address>
```

Example:

```
SCADA-serial0/0 Cfg>serial-address 145
SCADA-serial0/0 Cfg>
```

2.2.16 SIGNAL-CONTROL

This command is used to enable the handling of input control signal in the SCADA serial interface. This can be disabled by placing NO before the command, this means the signals are ignored. When this feature is activated, the input control signals must be active so the interface can operate correctly. This feature can be useful to deactivate the SCADA interface when the connected terminal is not operative. This command interacts with the output control signals, therefore, when the input control signals are configured to be ignored, the output ones are always active. On the other hand, when the input control signals are configured to not being ignored, those for the output only activate when the input signals are active. The **SIGNAL-OUTPUT-CONTROL** command interferes in the output signals control, please see this command for further information.

The **SIGNAL-TIMEOUT** command allows you to configure the time that must lapse from when the input control signal is deactivated until the router deactivates the interface.

Syntax:

```
SCADA-# Cfg>signal-control
```

Example:

```
SCADA-serial0/0 Cfg>signal-control
SCADA-serial0/0 Cfg>
```

2.2.17 SIGNAL-OUTPUT-CONTROL

This command is used to enable handling of the output control signal in the SCADA serial interface, based on the state changes produced by a track NSLA advisor. To disable this, enter the command with the word **NO** preceding it. So that this feature is operative, you need to enable the signal control through the **SIGNAL CONTROL** command.

The aim of this feature is to offer the possibility of notifying the terminal connected to the SCADA interface on the status of IP connectivity to the remote terminals. This feature, together with the others offered by the router, allows you to create rules so you can receive information on the link status through the track NSLA. The status signaling is

executed with the RTS signal when the serial interface behaves as DTE, and the DCD and CTS signals when the serial interface behaves as DCE.

The track NSLA always returns a FALSE or TRUE value depending on the configured conditions. Behavior is such that with track NSLA FALSE indications, the control signal is deactivated and with TRUE indications the control signal is activated.

The output signal control, which indicates the track NSLA status, is controlled based on the input signals, therefore if the input signals are not active, the output ones aren't either even if the conditions configured in the track NSLA indicate the contrary. The moment the input signals activate, the output signals change to the status corresponding to the current tract NSLA situation.

This feature is only operative when the output hardware interface is SERIAL and is not operative over UART interfaces. In SERIAL interfaces, this is only operative when the installed driver is an RS-232. It is not operative over RS-485 drivers as this type of interface does not use control signals.

As a parameter for this command, you need to enter the track NSLA ID. This field can takes values between 0 and 65535.

Manual *Dm754-I NSLA* explains how to operate NSLA. For further information on this, please see said manual.

Syntax:

```
SCADA-# Cfg>signal-output-control track nsla-advisor <id>
```

Example:

```
SCADA-serial0/0 Cfg>signal-output-control track nsla-advisor 2
SCADA-serial0/0 Cfg>
```

2.2.18 SIGNAL-TIMEOUT

This command allows you to configure the time in seconds that must lapse from the moment the input control signal deactivates until the router deactivates the interface. This parameter only makes sense when the serial interface control signal handling is enabled. Default is 1 second. You can enter values between 0 and 255 seconds.

Syntax:

```
SCADA-# Cfg>signal-timeout <time>
```

Example:

```
SCADA-serial0/0 Cfg>signal-timeout 20
SCADA-serial0/0 Cfg>
```

2.2.19 SPEED

The command specifies the interface reception and transmission speed. The range of values is between 300 and 115,200 bps. Default is 2400.

Syntax:

```
SCADA -# Cfg>speed <speed>
```

Example:

```
SCADA-serial0/0 Cfg>speed 9600
SCADA-serial0/0 Cfg>
```

2.2.20 TCP-INIT

This command allows you to disable the launching of outgoing TCP sessions on the center side. In some scenarios, on the center side, you do not want the device to launch TCP session, this initiative being taken by the remote devices (RTU). This command only makes sense when the device is configured in CENTER mode. By default the launching of TCP sessions from the center side is enabled. To disable this, enter this command preceded by the word No.

Syntax:

```
SCADA -# Cfg>tcp-init
```

Example:

```
SCADA-serial0/0 Cfg>tcp-init
SCADA-serial0/0 Cfg>
```

2.2.21 TERMINATION-RESISTOR

RS-485 buses spanning long distances need 120 ohms terminations at each end of the line (characteristic imped-
ance of a 2-wire line) to avoid signal degradation from reflections. RS-485 serial interfaces enable a 120 ohms ter-
mination resistor of through configuration.

The **TERMINATION-RESISTOR** command enables/disables the termination resistor. Default is disabled

Syntax:

```
SCADA-uartX/X Cfg>termination-resistor{ enable| disable }
```

Example:

```
SCADA-uartX/X Cfg>termination-resistor enable
SCADA-uartX/X Cfg>
```

This command is only available for hardware supporting RS-485 interfaces that control the termination resistor.

The termination resistor must be enabled if the device is located at the end of the bus.

2.2.22 TX-BUFFER_SIZE

Configures the maximum size of the message data field that can be sent over TCP. The permitted values are from 0
to 2048 octets. Default is 512. This parameter defines the maximum size of the buffers used in the transmission path,
i.e. TCP transmission buffer and reception buffer from the serial interface.

Syntax:

```
SCADA-serial0/0 Cfg>tx-buffer-size <buff-size>
```

Example:

```
SCADA-serial0/0 Cfg>tx-buffer-size 1024
SCADA-serial0/0 Cfg>
```

2.2.23 EXIT

Use the **EXIT** command to return to the previous prompt level.

Syntax:

```
SCADA -# Cfg>exit
```

Example:

```
SCADA-serial0/0 Cfg>exit
Config>
```

2.3 Show Config

Show Config is a configuration console tool (PROCESS 4) which allows you to list the commands required to con-
figure the router starting from an empty configuration (no conf).

This command can be used to both copy configurations as well as to list them or to simply view them.

Show Config starts from the internally defined default configuration, generating the commands from the part of the
configuration that defers to this.

Show Config can incorporate comments, which are placed after a semi-colon ('**;**').

This command can be executed from any menu, displaying the configuration introduced in all the submenus attached
to said menu.

Example:

The following example shows the configuration of 4 SCADA interfaces in a router with an IP address and an access
route to Internet. A TRMTP profile is included, which defines the access address to the host:

```
Config>show conf
; Showing System Configuration for access-level 15 ...

log-command-errors
no configuration
set data-link x25 serial0/1
set data-link x25 serial0/2
set data-link scada serial0/0
;
network ethernet0/0
; -- Ethernet Interface User Configuration --
   ip address 172.24.78.125 255.255.0.0
;
exit
;
network serial0/0
; -- SCADA Interface Configuration --
   protocol modbus-ascii
   speed 115200
   parity odd
   remote-register 1 name PZD_1
   remote-register 1 ip-address 172.24.78.99
   remote-register 2 name SRT_2
   remote-register 2 ip-address 123.233.4.22
   remote-register 2 tcp-port 20033
   remote-register 12 name BCN_12
   remote-register 12 ip-address 123.233.4.25
   remote-register 12 idle-time 1m
exit
;
;
;
;
network x25-node
; -- X25-node interface configuration --
   no ip address
;
exit
;
protocol ip
; -- Internet protocol user configuration --
   classless
;
exit
;
;
dump-command-errors
end
; --- end ---
Config>
```

You can copy, modify or edit the list of commands you obtain in **SHOW CONFIG** and use it as a templete for future configurations.

Chapter 3 Monitoring the SCADA Interface

3.1 Monitoring

Execute the following steps to enter the SCADA monitoring process:

(1)   At the (*) prompt, enter **PROCESS 3** or **P 3**. This will take you to the monitoring prompt +.

```
*p 3
+
```

(2)   At the (+) prompt enter the **NETWORK** command followed by the name that identifies the SCADA interface you wish to monitor, generically this is called <ifc>.

```
+network <ifc>
-- SCADA Console --
SCADA-ifc+
```

If, for example, the interface were **serial0/0**, the following would result:

```
+network serial0/0
-- SCADA Console --
SCADA-serial0/0+
```

The SCADA monitoring commands are numerated and described in this section. All the SCADA monitoring commands must be introduced at the SCADA prompt (SCADA- ifc+).

Command	Function
? (HELP)	Lists the available commands or their options.
CLEAR	Clears the terminal statistics counters.
LIST	Displays the terminal statistics information.
EXIT	Exits the SCADA monitoring environment.

3.1.1 ? (HELP)

The **?** (HELP) command lists all the commands available at the normal prompt level. You can also enter a **?** after a specific command name to obtain the options.

Syntax:

```
SCADA-ifc+?
```

Example:

```
SCADA-serial0/0+?
  clear    Initialize terminal statistics counters
  list     Display monitoring information
  exit
SCADA-serial0/0+
```

3.1.2 CLEAR

Command to delete statistics.

```
SCADA-serial0/0+clear ?
  all           All the monitoring information
  statistics    Interface traffic statistics
  transactions  Latest transactions
SCADA-serial0/0+
```

3.1.2.1 CLEAR ALL

Clears all the interface statistics: transactions and traffic.

Syntax:

```
SCADA-ifc+clear all
```

Example:

```
SCADA-serial0/0+clear all
SCADA-serial0/0+
```

3.1.2.2 CLEAR STATISTICS

Clears the interface traffic statistics.

Syntax:

```
SCADA-ifc+clear statistics
```

Example:

```
SCADA-serial0/0+clear statistics
SCADA-serial0/0+
```

3.1.2.3 CLEAR TRANSACTIONS

Clears the transactions statistics.

```
SCADA-serial0/0+clear transactions ?
  ok      Successful transactions
  wrong   Incomplete transactions
SCADA-serial0/0+
```

3.1.2.3.1 CLEAR TRANSACTIONS OK

Clears the successful transactions statistics.

Syntax:

```
SCADA-ifc+clear transactions ok
```

Example:

```
SCADA-serial0/0+clear transactions ok
SCADA-serial0/0+
```

3.1.2.3.2 CLEAR TRANSACTIONS WRONG

Clears the transaction attempts statistics where these have been dropped due to errors.

Syntax:

```
SCADA-ifc+clear transactions wrong
```

Example:

```
SCADA-serial0/0+clear transactions wrong
SCADA-serial0/0+
```

3.1.3 LIST

Allows you to view the different traffic, transactions, etc. parameters from the terminal connected to this interface.

```
SCADA-serial0/0+list ?
  all             All the monitoring information
  signal-changes  Registered changes in serial interface control signals
  statistics      Interface traffic statistics
  tcp             Statistics for sessions that affect the interface
  transactions    Latest transactions
SCADA-serial0/0+
```

3.1.3.1 LIST ALL

Lists all the interface traffic statistics and the status of the signals.

Syntax:

```
SCADA-ifc+list all
```

Example:

```
SCADA-serial0/0+list all

Mode         : center
Protocol     : iec101-trans
Driver type  : RS485-2w
Protocol State : 1


------- TRANSACCTIONS STATISTICS -------
Total transactions attempts      : 378477
Transactions OK                  : 378376
Transactions WRONG               : 101


------ SERIAL INTERFACE STATISTICS -----
Bytes sent to interface          : 2348351
Bytes received from interface    : 1894863
Packets sent to interface        : 378376
Packets received from interface  : 378583


-------- TCP INTERFACE STATISTICS ------
Bytes sent to TCP packets        : 1894047
Bytes received over TCP packets  : 2348351
TCP packets sent                 : 378475
TCP packets received             : 378376


---------- TCP GLOBAL STATISTICS -------
Attempt TCP sessions             : 4
Opened TCP sessions              : 2
Current TCP sessions             : 1
Max simultaneous TCP sessions    : 1
Bytes sent to TCP packets        : 1894047
Bytes received over TCP packets  : 2348351
TCP packets sent                 : 378475
TCP packets received             : 378376
Disconnects received             : 0
Disconnects transmitted          : 3

Errors or congestion detected    : 0

RTS Changes 1
CTS Changes 0
DSR Changes 24
DTR Changes 1
CD  Changes 8

SCADA-serial0/0+
```

3.1.3.2 LIST STATISTICS

Lists all the interface traffic statistics as well as the current status of the SCADA net. This is divided into various functional subgroups. Please bear in mind that some Teldat devices may have more than one SCADA interface and consequently the statistic information given here refers to the specified interface. To obtain an overview of what happens in the SCADA TCP session, the TCP global statistics are also given

Syntax:

```
SCADA-ifc+list statistics
```

Example:

```
SCADA-serial0/0+list statistics

Mode         : center
Protocol     : modbus-ascii
Driver type  : 1
Protocol State : 1
NSLA track   : TRUE
```

```
------- TRANSACCTIONS STATISTICS -------
Total transactions attempts     : 607
Transactions OK                 : 601
Transactions WRONG              : 6

------ SERIAL INTERFACE STATISTICS -----
Bytes sent to interface         : 99661
Bytes received from interface   : 20808
Packets sent to interface       : 601
Packets received from interface : 612

-------- TCP INTERFACE STATISTICS ------
Bytes sent to TCP packets       : 7224
Bytes received over TCP packets : 51934
TCP packets sent                : 602
TCP packets received            : 601

---------- TCP GLOBAL STATISTICS -------
Attempt TCP sessions            : 6
Opened TCP sessions             : 1
Current TCP sessions            : 1
Max simultaneous TCP sessions   : 1
Bytes sent to TCP packets       : 7224
Bytes received over TCP packets : 51934
TCP packets sent                : 602
TCP packets received            : 601
Disconnects received            : 0
Disconnects transmitted         : 5

Errors or congestion detected   : 0

SCADA-serial0/0+
```

The SCADA net status can take distinct values depending on the protocol used. The following values can be taken for the MODBUS protocol:

0. - Interface DOWN, not active.

1. - Waiting for a message from the MASTER device.

2. - Waiting for response from the SLAVE device.

The meaning of each of the fields is as follows:

Mode	Configured operation mode.
Protocol	Configured protocol.
Driver type	Type of configured serial driver.
Protocol State	Protocol state.
NSLA track	NSLA track status, if this is activated.

To calculate the transactions:

Total transactions attempts	Total number of transaction attempts.
Transactions OK	Number of successful transactions.
Transactions WRONG	Number of erroneous transactions.

Traffic in the serial interface:

Bytes sent to interface	Bytes sent to the interface.
Bytes received from interface	Bytes received from the interface.
Packets sent to interface	Messages sent to the interface.
Packets received from interface	Messages received from the interface.
CRC/LRC error packets received	Messages received with CRC or LRC errors.

TCP traffic generated by the interface:

Bytes sent over TCP packets	Bytes sent in TCP packets.
Bytes received over TCP pack-	Bytes received in TCP packets.

ets

TCP packets sent	TCP packets sent.
TCP packets received	TCP packets received.

TCP global traffic in all SCADA interfaces:

Attempt TCP sessions	TCP session establishment attempts.
Opened TCP sessions	Total number of TCP sessions which have been established.
Current TCP sessions	Number of active TCP sessions.
Max simultaneous TCP sessions	Maximum number of simultaneous TCP sessions.
Bytes sent over TCP packets	Bytes sent in TCP packets.
Bytes received over TCP packets	Bytes received in TCP packets.
TCP packets sent	TCP packets sent.
TCP packets received	TCP packets received.
Disconnects received	Disconnect queries received.
Disconnects transmitted	Disconnect queries sent.
Errors or congestion detected	Error situations due to overflow.

3.1.3.3 LIST SIGNAL-CHANGES

Lists the changes registered in the serial interface control signals.

Syntax:

```
SCADA-ifc+list signal-changes
```

Example:

```
SCADA-serial0/0+list signal-changes
RTS Changes 1
CTS Changes 0
DSR Changes 6
DTR Changes 1
CD  Changes 568


SCADA-serial0/0+
```

3.1.3.4 LIST TCP

Lists the statistics for the sessions which affect the specified interface configuration. Please bear in mind that some Teldat devices may have more than one SCADA interface and consequently the statistic information given here refers to the specified interface. It is possible that more than one interface will use the same TCP session to exchange information and therefore this can be present in both interfaces.

Syntax:

```
SCADA-ifc+list tcp
```

Example:

```
SCADA-serial0/0+list tcp

---- SCADA INTERFACE ASSOCIATED TCP INFO ------

---- GLOBAL INFO ------
SCADA protocol            : iec101-trans
IEC-101 mode              : unbalanced
TCP listen port           : 502
Opened sessions           : 1
Current sessions          : 1
Max simultaneous sessions : 1
Disconnects received      : 0
Disconnects transmitted   : 0

---- TCP SESSIONS INFO ------
  STATE    REMOTE IP ADDR. REM.PORT  ESTABLISH TIME    ORIG.
```

```
--------- --------------- -------- ----------------- ------
 ESTAB     172.24.78.125   1027     12:15:32 29/12/04 remote


SCADA-serial0/0+
```

The meaning of each field is as follows:

Global information on TCP sessions that affect the interface:

SCADA protocol	Configured protocol.
Attempt sessions	TCP session establishment attempts (only in CENTER or REMOTE mode in IEC-101 BALANCED).
IEC-101 mode	Indicates if the IEC-101 uses BALANCED mode (this only appears if the configured protocol is IEC).
TCP listen port	TCP listen port (should it exist).
Opened sessions	Total number of TCP sessions which have been established.
Current sessions	Number of active TCP sessions.
Max simultaneous sessions	Maximum number of simultaneous TCP sessions.
Disconnects received	Disconnect queries received.
Disconnects transmitted	Disconnect queries sent.

Information on established TCP sessions which affect the interface:

STATE	State of the session.
REMOTE IP ADDR.	Remote end IP address.
REM. PORT	Remote end TCP port.
ESTABLISH TIME	Time when the connection was established.
ORIG	TCP session origin.

3.1.3.5 LIST TRANSACTIONS

Use this command to view the latest transactions executed by this interface, both successful transactions as well as those that could not be completed.

```
SCADA-serial0/0+list transactions ?
  ok      Successful transactions
  wrong   Incomplete transactions
SCADA-serial0/0+
```

3.1.3.5.1 LIST TRANSACTIONS OK

Lists the latest successful transactions. After the command you can enter the number of transactions you wish to view. If you omit this number, all the correct transactions stored are displayed.

Syntax:

```
SCADA-ifc+list transactions ok [<num_transac>]
```

Example:

```
SCADA-serial0/0+list transactions ok


SER.ADD IP ADDRESS      PORT   START     END       DATE
   1     172.24.78.99    502    12:16:58 12:16:58 29/12/04
   1     172.24.78.99    502    12:16:58 12:16:58 29/12/04
   1     172.24.78.99    502    12:16:58 12:16:58 29/12/04
   1     172.24.78.99    502    12:16:58 12:16:58 29/12/04
   1     172.24.78.99    502    12:16:58 12:16:58 29/12/04
   1     172.24.78.99    502    12:16:58 12:16:58 29/12/04
   1     172.24.78.99    502    12:16:58 12:16:58 29/12/04
   1     172.24.78.99    502    12:16:57 12:16:58 29/12/04
   1     172.24.78.99    502    12:16:57 12:16:57 29/12/04
   1     172.24.78.99    502    12:16:57 12:16:57 29/12/04
   1     172.24.78.99    502    12:16:57 12:16:57 29/12/04
   1     172.24.78.99    502    12:16:57 12:16:57 29/12/04
   1     172.24.78.99    502    12:16:57 12:16:57 29/12/04
   1     172.24.78.99    502    12:16:57 12:16:57 29/12/04
   1     172.24.78.99    502    12:16:57 12:16:57 29/12/04
```

```
   1     172.24.78.99     502   12:16:56 12:16:57 29/12/04
   1     172.24.78.99     502   12:16:56 12:16:56 29/12/04

SCADA-serial0/0+
```

The meaning of each field is as follows:

SER.ADD	Serial interface address.
IP ADDRESS	Remote end IP address.
PORT	Remote end TCP port.
START	Indicates the operation start time.
END	Indicates the operation end time.
DATE	Indicates the data the operation was executed.

3.1.3.5.2 LIST TRANSACTIONS WRONG

Lists the latest transactions that were not successful. After the command you can enter the number of transactions you wish to view. If you omit this number, all the incomplete transactions stored are displayed.

Syntax:

```
SCADA-ifc+list transactions wrong [<num_transac>]
```

Example:

```
SCADA-serial0/0+list transactions wrong 3

SER.ADD IP ADDRESS       PORT  CAUSE  START     END       DATE
   1    172.24.78.99     502   3      12:15:40 12:15:50 29/12/04
   1    172.24.78.99     502   3      12:06:59 12:07:09 29/12/04
   1    172.24.78.99     502   3      12:06:47 12:06:57 29/12/04

SCADA-serial0/0+
```

The meaning of each field is as follows:

SER.ADD	Serial interface address.
IP ADDRESS	Remote end IP address.
PORT	Remote end TCP port.
CAUSE	Reason for failure.
START	Indicates the operation start time.
END	Indicates the operation end time.
DATE	Indicates the data the operation was executed.

The possible values for the cause field are:

(1) No register is configured for the RTU number received by the serial interface.

(2) The serial terminal control signal has deactivated.

(3) Timeout has occurred waiting for a response from the remote terminal via TCP.

(4) An erroneous transaction identifier has been received via TCP.(MODBUS).

(5) An erroneous protocol identifier has been received via TCP. (MODBUS).

(6) An erroneous length field has been received via TCP. (MODBUS).

(7) A serial RTU identifier received via TCP has not been found. (MODBUS).

(8) The RTU number received by the serial interface is erroneous.

(9) Timeout has occurred waiting for a response from the RTU for the serial interface.

(10) A response message cannot be sent via TCP. (MODBUS).

(11) RTU query received, via TCP, on the center side with PRM bit = 0, in balanced mode (IEC-101).

(12) RTU query received, via TCP, in the center side which does not have a configured register, in balanced mode (IEC-101).

(13) Invalid response received, via serial, where the address in the message received from the RTU does not coincide with the address received from the center side (IEC-101).

(14) Response to RTU received, via serial, in the center side which does not have a configured register, in balanced mode (IEC-101).

(15) Response to RTU received, via serial, from the center side with PRM bit = 1, in balanced mode (IEC-101).

(16) Timeout waiting for response from the center side via the serial interface (IEC-101).

(17) Query from the center side received, via TCP, with PRM bit = 0 (IEC-101).

(18) Received, via TCP, message address which does not coincide with the RTU address (IEC-101).

(19) RTU query received, via serial, with PRM bit = 0, in balanced mode (IEC-101).

(20) Messages cannot be sent via TCP (IEC-101).

(21) RTU response received, via serial, with PRM bit = 1, in balanced mode (IEC-101).

(22) Timeout in remote device waiting for a response from the center side, via TCP, in balanced mode (IEC-101).

Chapter 4 SCADA Interface ELS

4.1 SCADA Interface Events Logging System

Allows real time monitoring of the events that occur in one or various SCADA interfaces, when the events logging system is enabled for this protocol.

This system is enabled from the configuration menu in the following way:

```
*process 4
Config>event

-- ELS Config --
ELS Config>enable trace subsystem scada all
ELS Config>exit
Config>save
Save configuration [n]? y

Saving configuration...OK
Config>
```

This system can also be enabled from the monitoring menu at any point, without this being stored in the configuration, in the following way:

```
*process 3
Console Operator
+event

-- ELS Monitor --
ELS+enable trace subsystem scada all
ELS+exit
+
```

To view the events once these are enabled, enter the following:

```
*process 2
12/29/04 12:16:57  SCADA.001 Rx Message from Ser.Term. 17 bytes, nt serial0/0
12/29/04 12:16:57  SCADA.002 Rx Message from Ser.Term. add 1, 7 bytes, nt serial
0/0 st 1
12/29/04 12:16:57  SCADA.010 Sending message to Remote 12 bytes, nt serial0/0
12/29/04 12:16:57  SCADA.012 Rx TCP message header: 6 bytes. rem. IP: 172.24.78.
99, Port: 1029
12/29/04 12:16:57  SCADA.012 Rx TCP Complete: 28 bytes. rem. IP: 172.24.78.99, P
ort: 1029
12/29/04 12:16:57  SCADA.016 Tx Message to Ser.Term.,addr:1 61 bytes nt serial0/
0 st 2
12/29/04 12:16:57  SCADA.009 Transfer OK, nt serial0/0
```

The list of available events for the SCADA interface depends on the software release. Each software release distribution is accompanied by its own group of available events.

4.2 Example of events for a successful transaction

An example of the events you will see after a successful transaction is as follows:

```
*process 2
12/29/04 13:40:33  SCADA.001 Rx Message from Ser.Term. 17 bytes, nt serial0/0
12/29/04 13:40:33  SCADA.002 Rx Message from Ser.Term. add 1, 7 bytes, nt serial
0/0 st 1
12/29/04 13:40:33  SCADA.008 Address 1 connecting: IP 172.24.78.99, P 502, nt
serial0/0
12/29/04 13:40:33  SCADA.015 TCP Session Opened. rem. IP: 172.24.78.99, rem. Port: 502
12/29/04 13:40:33  SCADA.010 Sending message to Remote 12 bytes, nt serial0/0
12/29/04 13:40:33  SCADA.012 Rx TCP message header: 6 bytes. rem. IP: 172.24.78.
99, Port: 1024
12/29/04 13:40:33  SCADA.012 Rx TCP Complete: 28 bytes. rem. IP: 172.24.78.99, Port: 1024
```

```
12/29/04 13:40:33  SCADA.016 Tx Message to Ser.Term.,addr:1 61 bytes nt serial0/
0 st 2
12/29/04 13:40:33  SCADA.009 Transfer OK, nt serial0/0
```