Schneider Electric

Modbus Slave Protocol
XBT N/R/RT

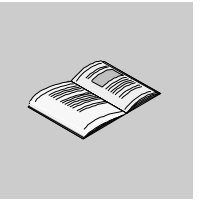
04/2009

33003980.02

www.schneider-electric.com

© 2009 Schneider Electric. All rights reserved.

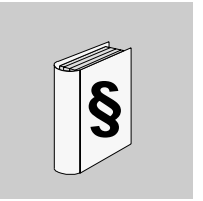
Table of Contents



Safety Information **5**
About the Book **7**
Chapter 1 Operating Principle **9**
General Information on Bus Communications 11
Master / Slave Communication Principle 13
Communication according to the OSI Model 15
Modbus RTU Transmission Mode 17
Modbus RTU Framing 18
Modbus Frame Description 20
Example of a Serial Modbus RTU Communication Bus 21
Cable Length and Grounding 22
RC Termination 23
Line Polarization 24
Addressing 26
Equipment Symbols 27
Chapter 2 Software Configuration **29**
Vijeo-Designer Lite 30
Protocol - Modbus Slave Dialog Box 32
Chapter 3 Variable Types Supported **35**
Variable Types for Modbus Slave 35
Chapter 4 Cables and Connectors **37**
Cables 38
SUB-D25 Pin Connections 40
RJ45 Pin Connections 43
Chapter 5 Diagnostics **47**
XBT Detected Error Indication 47
Chapter 6 Bandwidth Principle **51**
General Operating Principle 52
Calculating Bandwidth Usage 54
Tips 58
Appendices **59**

Appendix A Communication Requests. 61
Communication Requests 61
Glossary 65
Index 69

Safety Information



Important Information

NOTICE

Read these instructions carefully, and look at the equipment to become familiar with the device before trying to install, operate, or maintain it. The following special messages may appear throughout this documentation or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.



The addition of this symbol to a Danger or Warning safety label indicates that an electrical hazard exists, which will result in personal injury if the instructions are not followed.



This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

⚠ DANGER

DANGER indicates an imminently hazardous situation which, if not avoided, **will result in** death or serious injury.

⚠ WARNING

WARNING indicates a potentially hazardous situation which, if not avoided, **can result in** death or serious injury.

⚠ CAUTION
CAUTION indicates a potentially hazardous situation which, if not avoided, **can result in** minor or moderate injury.

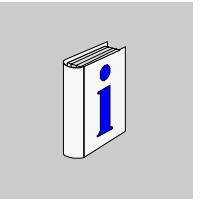
CAUTION
CAUTION, used without the safety alert symbol, indicates a potentially hazardous situation which, if not avoided, **can result in** equipment damage.

PLEASE NOTE

Electrical equipment should be installed, operated, serviced, and maintained only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.

A qualified person is one who has skills and knowledge related to the construction and operation of electrical equipment and the installation, and has received safety training to recognize and avoid the hazards involved.

About the Book



At a Glance

Document Scope

This document describes communication between automation systems and the XBT N/R/RT product range using the Modbus Slave protocol.

Validity Note

The data and illustrations found in this document are not binding. We reserve the right to modify our products in line with our policy of continuous product development. The information in this document is subject to change without notice and should not be construed as a commitment by Schneider Electric.

Related Documents

Title of Documentation	Reference Number
XBT N/R/RT Instruction sheet	W916810140111 A08
Modbus Protocol Reference Guide	PI-MBUS-300 (available at www.modbus.org
XBT N/R/RT User Manual	33003962
Vijeo-Designer Lite	Online help

You can download these technical publications and other technical information from our website at www.schneider-electric.com.

Product Related Information

Schneider Electric assumes no responsibility for any errors that may appear in this document. If you have any suggestions for improvement or amendments or have found errors in this publication, please notify us.

No part of this document may be reproduced in any form or by means, electronic or mechanical, including photocopying, without express written permission of Schneider Electric.

All pertinent state, regional and local safety regulations must be observed when installing and using this product. For reasons of safety and to ensure compliance with documented system data, only the manufacturer should perform repairs to components.

Since the XBT N/R/RT terminals are not designed to pilot safety critical processes, no specific instructions apply in this context.

User Comments

We welcome your comments about this document. You can reach us by e-mail at techcomm@schneider-electric.com.

Operating Principle

1

Overview

This chapter describes the operating principle of XBT terminals in applications using the Modbus slave protocol.

⚠ WARNING

LOSS OF CONTROL

- The designer of any control scheme must consider the potential failure modes of control paths and, for certain critical functions, provide a means to achieve a safe state during and after a path failure. Examples of critical control functions are emergency stop and overtravel stop.
- Separate or redundant control paths must be provided for critical for critical control functions.
- System control paths may include communication links. Consideration must be given to the implications of unanticipated transmission delays or failures of the link.*
- Each implementation of a Magelis XBT N/R/RT must be individually and thoroughly tested for proper operation before being placed into service.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

*For additional information, refer to NEMA ICS 1.1 (latest edition), *Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control*

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
General Information on Bus Communications	11
Master / Slave Communication Principle	13

Topic	Page
Communication according to the OSI Model	15
Modbus RTU Transmission Mode	17
Modbus RTU Framing	18
Modbus Frame Description	20
Example of a Serial Modbus RTU Communication Bus	21
Cable Length and Grounding	22
RC Termination	23
Line Polarization	24
Addressing	26
Equipment Symbols	27

General Information on Bus Communications

Overview

XBT terminals can be connected to PLCs using different protocols. This document describes the communication on Modbus field buses using the Modbus RTU protocol with the XBT terminal acting as slave.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

The protocol must be installed and used by authorized and properly trained personnel.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

Roles of XBT Terminals

The terminals are usually connected to a communication equipment (PLC or other) via a field bus. The XBT and the PLCs work autonomously of each other.

XBT terminals perform the following functions:

- monitoring function: XBT terminals visualize the processes that are active in the PLCs and indicate alarm states
- command function: XBT terminals send information to the PLC upon user request

Roles of Buses

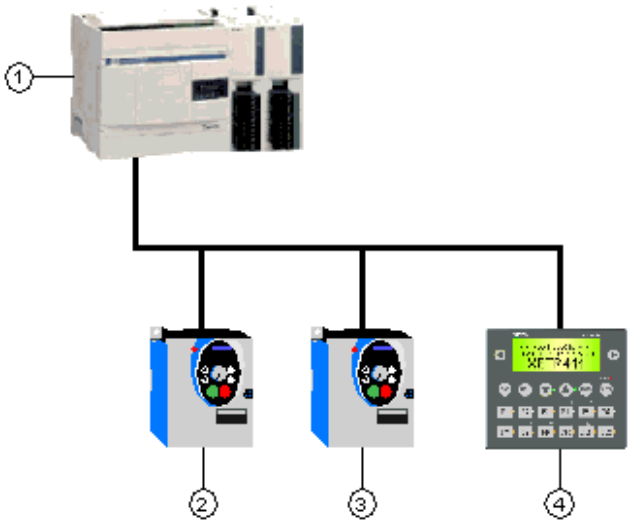
A bus system provides the possibility to connect different devices via a unique cabling.

Roles of Protocols

The protocol defines the language that is used by all the equipment connected to the bus.

Application Principle

The following figure shows a basic Modbus application with XBT acting as slave:



1 TWIDO, connected via the TER connector
2 Speed drive Altivar 31
3 Speed drive Altivar 31
4 XBT R

The XBT is totally passive with respect to communication. The PLC reads or writes the data in the XBT memory. If the PLC does not send any data to the XBT (or does not attempt to read from the XBT memory), the values of the XBT memory are not refreshed. After expiration of the communication time-out, the values displayed by the XBT are replaced by `??` characters and a system message appears to signal connection error. To avoid any time-out check by the XBT, the value 0 should be entered for this time-out parameter.

When you press a key, if the `Function key status` word has not been read by the PLC, the LED associated with the key flashes rapidly and pressing the key again has no effect. Once the word has been read by the PLC, the LED stops flashing and the key can be used again.

NOTE: In Modbus slave mode, the XBT terminal does not read/write PLC variables.

Master / Slave Communication Principle

Overview

Modbus communications are performed according to the master / slave principle that is described in the following.

Characteristics of the Master / Slave Principle

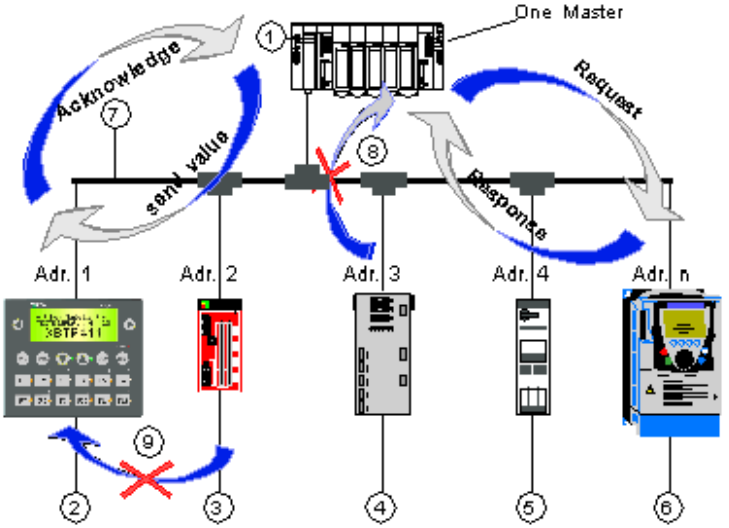
The master / slave principle is characterized as follows:

- Only one master is connected to the bus at a time.
- One or several slaves can be connected to the same serial bus.
- Only the master is allowed to initiate communication, i.e. to send requests to the slaves.
- In Modbus communications, the master can only initiate one Modbus transaction at the same time.
- In Modbus communications, the master can address each slave individually (unicast mode) or all slaves simultaneously (broadcast mode).
- The slaves can only answer requests they received from the master.
- The slaves are not allowed to initiate communication, neither to the master nor to any other slaves.
- In Modbus communications, the slaves generate an error message and send it as response to the master if an error occurred in receipt of the message or if the slave is unable to perform the requested action.

Terminals acting as Slave in Modbus Applications

In Modbus slave applications, the XBT terminal acts as slave device, i.e. as server.

Master / slave communication



1 Automation Platform Premium PLC
2 XBT R411 (in Modbus slave operation)
3 XPSMF40 Safety PLC
4 XPSMF30 Safety PLC
5 TesysU
6 Altivar 71
7 Modbus SL bus
8 slaves cannot initiate the communication
9 slaves cannot communicate with other slaves

Communication according to the OSI Model

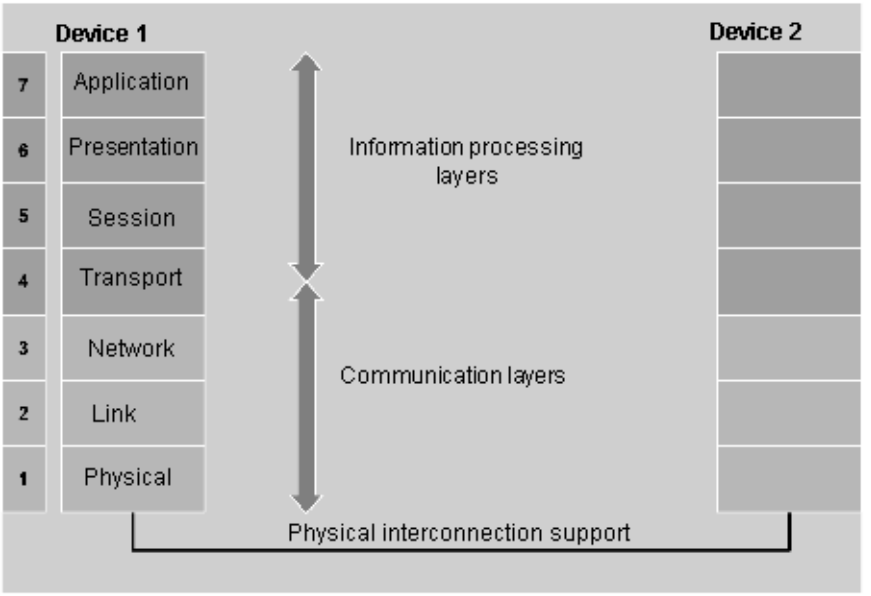
At a Glance

Communication between same-type devices can only take place by defining interconnection standards that define the behavior of each device in relation to the others. These standards were developed by ISO (the International Standard Organization), which defined a standardized Network Architecture more commonly known as the OSI (Open System Interconnection) model.

This model is made up of seven ranked layers that each perform a specific part of the functions necessary for interconnecting systems.

The layers communicate with equivalent layers from other devices, via standardized protocols. Within a single device, layers communicate with their immediate neighbors via hardware or software interfaces.

Layers of the OSI Model



NOTE: The Modbus RTU bus matches this model in terms of layers, without possessing all of them. Only the Application (Modbus), Link and Physical (Modbus RTU) layers are necessary for this field bus.

Application Layer

The application layer of the RTU Modbus serial field bus is the one visible to the programs of the interconnected devices. This is used to formulate the requests (reading/writing words and bits, etc.) that will be sent to the remote device.

The application layer used by the Modbus RTU bus is the Modbus application protocol.

Example for Modbus Master: An XBT terminal connected as master to a Modbus RTU bus will send Modbus requests to a Modbus slave device to read variables in order to update values represented by the semigraphic objects displayed on its panels.

Example for Modbus Slave: An XBT terminal connected as a slave to a Modbus RTU bus will receive the Modbus requests from the master in order to update the values represented by the semigraphic objects displayed on its panels.

NOTE: For further details on the Modbus application protocol (request codes, class details, etc.), visit *http://www.modbus.org*.

Link Layer

The link layer of the serial Modbus RTU bus uses the master/slave communication principle. The principle of a link layer is to define a low-level communication method for the communication medium (physical layer).

NOTE: One reason for master/slave management is that at any time it is possible to calculate transfer time for requests and the answers from each device. This therefore enables the terminal to size the volume of the communication on the buses precisely, in order to avoid any saturation or information loss.

NOTE: When using the Modbus (RTU) driver, the XBT terminal is the bus master. When using the Modbus Slave (RTU) driver, the XBT terminal is a slave on the bus.

NOTE: For further details (datagrams, frame sizes, etc.) visit *http://www.modbus.org*.

Physical Layer

The physical layer of the OSI model characterizes the topology of the communication bus or network, as well as the medium (cable, wire, fiber optic, etc.) that will transport the information and its electrical coding.

Within the framework of a serial Modbus RTU bus, topology may be daisy-chained, derived or a mix of both. The medium is made up of shielded twisted pairs, and the signal is a base band signal with a default speed of 19,200 bit/s

NOTE: In order for all devices to be able to communicate among themselves on the same bus, the speed must be identical.

Modbus RTU Transmission Mode

Overview

RTU is the standard Modbus transmission mode that is supported by XBT terminals. In this transmission mode each 8-bit byte of a message contains 2 x 4-bit hexadecimal characters.

The outdated ASCII transmission mode is not supported by XBT terminals.

Byte Format

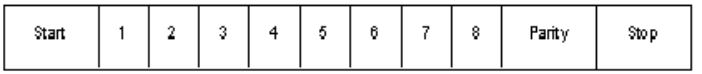
Each byte (11 bits) has the following format

Coding System	8-bit binary
Bits per Byte	1 start bit 8 data bits, least significant bit sent first 1 bit for parity completion 1 stop bit
Parity	even parity odd parity no parity

Start and stop bits are integrated in front of (start bit) and at the end (stop bit) of a byte to indicate that a byte is beginning (start bit) or ending (stop bit).

A parity bit is usually included in Modbus RTU transmission mode in order to perform an error check on the byte content. Deviating from the Modbus standard, XBT terminals also support data transmission with 1 start bit, 8 data bits, with only 1 stop bit and without parity bit. You can choose to transmit data with or without parity check, but always make sure that all equipment connected to the Modbus bus are configured to the same mode otherwise no communication will be possible.

Bit sequence in RTU mode with parity checking



NOTE: In order for all devices to be able to communicate among themselves on the same bus, the parity and data bit number characteristics must be identical for all the devices.

Modbus RTU Framing

Overview

A Modbus message is transmitted in a frame with a defined beginning and a defined end point. This indicates to the receiving devices when a new message starts and when it is completed. The receiving devices can detect incomplete messages and inform the master by issuing error codes.

RTU Frame

In addition to the user data, the RTU frame includes the following information:

- slave address (1 byte)
- function code (1 byte)
- Cyclic Redundancy Checking (CRC) field

The maximum size of an RTU frame is 256 bytes.

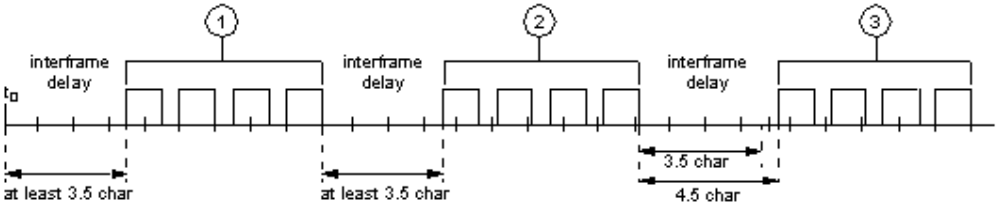
RTU message frame

Slave Address	Function Code	Data	CRC	
1 byte	1 byte	0...252 byte(s)	2 bytes	
			CRC Low Byte	CRC High Byte

Separating Message Frames by Silent Times

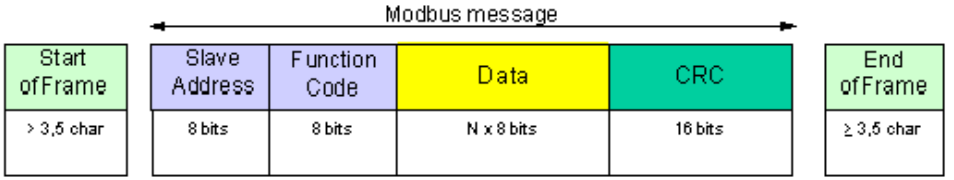
Individual frames are separated by a silent interval, also called interframe delay, of at least 3.5 character times. The following figure provides an overview of 3 frames being separated by an interframe delay of at least 3.5 character times.

Message frames separated by silent times



1 Frame 1
2 Frame 2
3 Frame 3

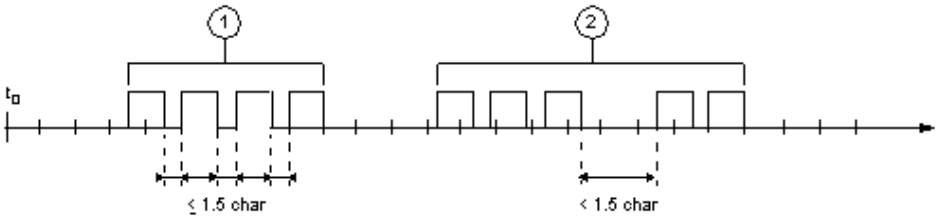
RTU message frame with start and end silent times



Detecting Incomplete Frames

In RTU mode it is required that the entire message frame is transmitted as a continuous stream of characters because silent times larger than 1.5 character times between 2 characters will be interpreted by the receiving device as incomplete frame. The receiver will discard this frame.

Detecting incomplete frames



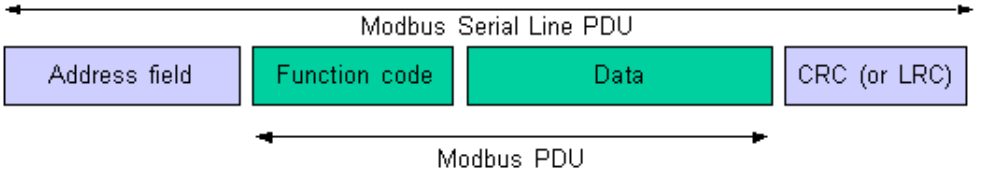
1 Frame 1 OK
2 Frame 2 Not OK

Modbus Frame Description

Overview

A Modbus frame is also referred to as data frame or telegram. The basic Modbus frame consists of the protocol data unit (PDU) that is extended in Modbus SL communications by the address field of the Modbus SL slave and the error checking field.

Modbus frame



Frame Segments

The extended Modbus Serial Line frame consists of the following segments:

Frame Segment	Size	Description
Address Field	1 byte	contains address of requested slave
Function Code	1 byte	contains the function code
Data	n bytes (high-byte, low-byte)	contains the data belonging to the request
CRC	2 bytes (low-byte, high-byte)	contains the error check sum

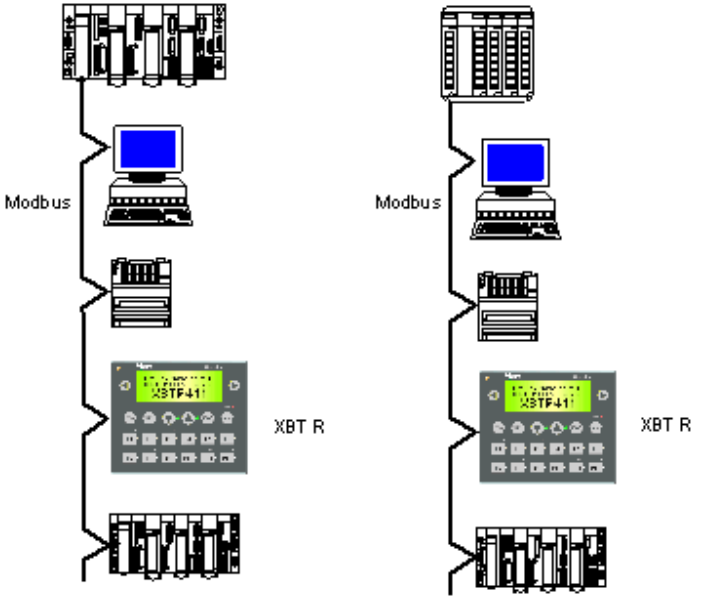
Example of a Serial Modbus RTU Communication Bus

At a Glance

Schneider devices are used to associate serial Modbus RTU communication buses with stand-alone stations, enabling them to communicate with XBT operator dialog terminals.

Examples of Buses

The following figures show two examples of serial Modbus RTU buses, that can be used with stand-alone Premium or Quantum stations:



Cable Length and Grounding

Overview

When setting up a new Modbus application, always use a shielded twisted pair cable and consider the maximum cable length allowed. Restrictions apply to the trunk cable (bus) as well as to the individual derivations.

Factors Influencing the Length of the Trunk Cable

The following factors influence the length of the trunk cable:

- transmission rate
- cable type (gauge, capacitance or characteristic impedance)
- number of loads that are directly connected (daisy chaining)
- network configuration (2-wire or 4-wire)

NOTE: If you are using a 4-wire cabling system for a 2-wire application, please note that the maximum cable length must be divided by two.

Cable Length Examples

The following table provides an example of determining the cable length according to the transmission rate and the cable type:

Transmission Rate	19,200 bit/s
Cable Type (Gauge)	0.125...0.161 mm^2 (AWG 26) (or larger)
Maximum Cable Length	1000 m (3280 ft)

Expanding the Cable Length Using Repeaters

To expand the length of your Modbus SL trunk cable you can integrate repeaters in your system. With a maximum of 3 repeaters being allowed in 1 system, you can expand the allowed cable length by factor 4, i.e. to a maximum cable length of 4,000 m (13,123 ft).

Length of Derivation Cables

The length of each derivation must not exceed 20 m (65 ft).

If you are using a multi-port tap with n derivations, make sure that the maximum length of 40 m (131.23 ft) is not exceeded for all n derivations together.

Grounding

The shield of the connector must be connected to protective ground at least at 1 point.

RC Termination

Overview

To help prevent unintended effects, like reflections, from occurring in your Modbus application, make sure to terminate the transmission lines properly.

⚠ CAUTION

LOSS OF DATA AND ELECTROMAGNETIC COMPATIBILITY ISSUES

- Terminate transmission lines at both ends. This minimizes loop current and line reflections, increases the electromagnetic compatibility, and helps protect an open input receiver.
- Program Modbus slaves such that an incomplete data transfer is sent back to the Modbus master.Failure to follow these instructions can result in injury or equipment damage.

Failure to follow these instructions can result in injury or equipment damage.

Terminating Your Network with RC Termination

To terminate your network with RC termination, proceed as follows:

Step	Action
1	Choose 2 serial capacitors of 1 nF, 10 V minimum and two 120 Ω (0.25 W) resistors as line termination.
2	Integrate these components at both ends of your Modbus communication line as shown in pos. 5 of the schematic diagram in section *Integrating Polarization Resistors into the Application, page 24*.
3	Connect these line terminations between the 2 conductors of the balanced Modbus line.

Line Polarization

Overview

In cases when there is no data activity, the bus is subjected to external noise or interference. In order to prevent the receivers from adopting improper states, some Modbus devices need to be biased, i.e. the constant state of the line must be maintained by an external pair of resistors connected to the RS485 balanced pair.

Biasing Your Network

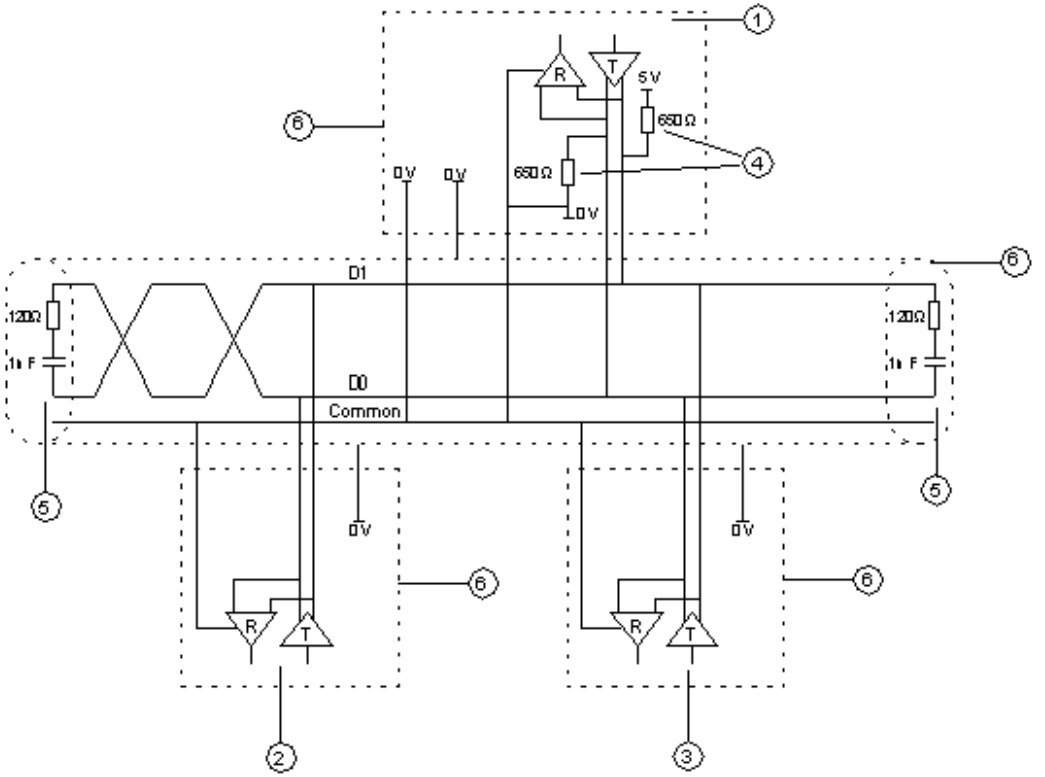
To provide proper line polarization, proceed as follows:

Step	Action
1	Check the devices you want to integrate into your Modbus application: Is there any device that needs external line polarization? If at least 1 of the devices needs external line polarization, proceed with step 2, otherwise no line polarization is required for your current application. For further details on the polarization resistors integrated in XBT terminals refer to the chapter on cables and connectors.
2	Integrate a pull-up resistor (650 Ω recommended) to a 5 V voltage into the D1 circuit.
3	Integrate a pull-down resistor (650 Ω recommended) to the common circuit into the D0 circuit.

Integrating Polarization Resistors into the Application

NOTE: The pair of polarization resistors must only be integrated at one location for the whole serial bus. You should integrate these resistors at the master device or its tap as shown in the figure below.

Schematic diagram



Elements of the application

No.	Element
1	master
2	slave 1
3	slave n
4	polarization resistors (required for XBT N, already included in XBT R)
5	line termination
6	shield

Addressing

Overview

With the Modbus slave protocol, the terminal behaves like a slave. It can therefore answer requests to addresses between 0 and 30.

Value	Meaning
0	The value 0 is reserved for broadcasting. Messages sent to address 0 will be received by all equipment connected to the bus. This can be used to send identical data to all the equipment, instead of sending a message to each item of equipment.
31	The value 31 is synonymous with disconnection for the terminal. A terminal detects an address 31 when no cable is connected to it. For this reason, any terminal configured with this address believes itself to be disconnected and displays messages requesting reconnection.

Connection to the Modbus Slave

Several types of connection are offered:

Using an...	Then ...
• XBT Z968 cable (straight) or, • XBT Z9680 cable (angled)	the address of the terminal is hard-wired and is worth 4.
XBT Z938 cable	the terminal address is configured in the software.
XBT Z908 cable and an SCA62 box	the address is "hard-wired" using the jumpers on the SCA62 box (the address will be between 1 and 30).

Equipment Symbols

Overview

Since the XBT terminal is totally passive, the Modbus slave protocol does not require equipment symbols to be declared.

Software Configuration

2

Overview

This chapter contains the protocol parameters you must configure in the Vijeo-Designer Lite software for operating XBT terminals as Modbus slave.

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Vijeo-Designer Lite	30
Protocol - Modbus Slave Dialog Box	32

Vijeo-Designer Lite

Overview

Use the Vijeo-Designer Lite software to configure your XBT terminal as Modbus slave.

⚠ WARNING

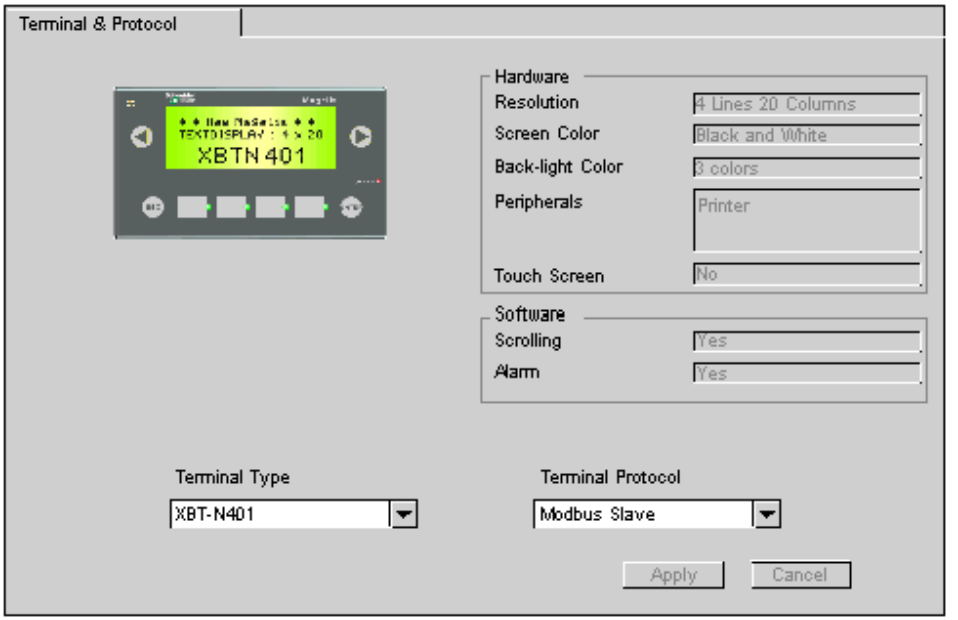
INCOMPATIBLE SOFTWARE

Use only Schneider Electric manufactured or approved software to program hardware.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

Opening the Protocol - Modbus Slave Dialog Box

To open the **Protocol - Modbus Slave** dialog box in Vijeo-Designer Lite for setting the protocol parameters, proceed as follows:

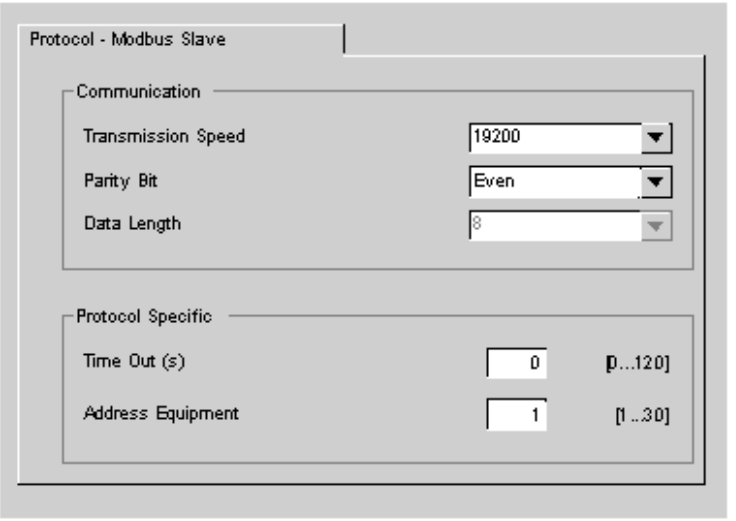
Step	Action
1	Start Vijeo-Designer Lite. To create a new application, continue with step 2, if you have already created a Modbus slave application, skip steps 2 and 3 and execute step 4.
2	From the application browser on the left-hand side of the Vijeo-Designer Lite window select the item **Configuration** → **Terminal & Protocol**. **Result**: The following dialog box will be displayed on the right-hand side of the Vijeo-Designer Lite window. 
3	From the **Terminal Protocol** list in the lower right corner select the item **Modbus Slave** and click **Apply**.
4	Select from the application browser the item **Protocol - Modbus Slave**. **Result**: The dialog box **Protocol - Modbus Slave** will be displayed on the right-hand side of the Vijeo-Designer Lite window where you can configure the protocol parameters for Modbus slave communication.

Protocol - Modbus Slave Dialog Box

Purpose

Use this dialog box to configure the protocol parameters for Modbus slave communication.

Representation



Elements of the dialog box

Element	Description
Communication	
Transmission Speed	Select the transmission speed (in bit/s) on your Modbus bus from the list. Make sure to configure the same transmission speed for all devices connected to the bus.
Parity Bit	Select either even, odd or no parity. Make sure to configure the same parity value for all devices connected to the bus.
Data Length	You cannot edit this parameter because in Modbus RTU communications the length of user data is always 8 bits.
Protocol Specific	

Element	Description
Time Out (s)	Enter a value (in seconds). In times when the PLC does not send any data to the XBT (or does not attempt to read from the XBT memory), the values of the XBT memory are not refreshed. After the time configured with this parameter has elapsed without any data interchange with the PLC, the XBT terminal replaces the values on its display unit by ??? characters and issues a system message to indicate that a connection error has occurred. To avoid any time-out check by the XBT, enter the value 0 for this parameter.
Address Equipment	Enter a unique Modbus address (between 1 and 247). This address will be ignored if the XBT terminal detects a hard-wired address on the address pins of its SUB-D25 connector.

Variable Types Supported

3

Variable Types for Modbus Slave

Table of Variable Types Supported by the XBT

The addressable XBT internal memory is limited to 300 words, of address 0...299.

Variable Type Supported	Syntax	Identifiers
Word Bit	%MWi:Xj	i: (0...299) j: (0...F)
Word	%MWi	i: (0...299)
Double Word	%MDi	i: (0...298)
Floating Point	%MFi	i: (0...298)

Cables and Connectors

4

Overview

This chapter specifies the cables and connectors required for XBT terminals in Modbus slave applications.

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Cables	38
SUB-D25 Pin Connections	40
RJ45 Pin Connections	43

Cables

Technical Data

The following table lists the cables required to connect the different XBT terminals as Modbus slave to different Schneider PLCs using RS485 or RS232C lines.

XBT Type	Connected Device	Physical Link	Cable Reference	Length and Type
XBT N401/N410 XBT R411	Twido	RS485	XBT Z908 + TSX SCA62	1.8 m (5.9ft.) (SUB-D25 <--> SCA62 box)
	Micro			
	Premium			
	Nano			
	LU9GC3	RS232C	XBT Z938	2.5 m (16.4 ft.) (SUB-D25 <--> RJ45)
	Quantum		XBT Z9710	2.5 m (16.4 ft.) (SUB-D25 <--> SUB-D9)
	Momentum		XBT Z9711	2.5 m (16.4 ft.) (SUB-D25 <--> RJ45)
XBT RT511	Twido	RS485	XBT Z9780 XBT Z9782	2.5 m (8.2 ft.) 2.5 m (8.2 ft.) (RJ45 <--> MiniDin)
	Modicon M340	RS485	XBT Z9980 XBT Z9982	2.5 m (8.2 ft.) 10 m (32.8 ft.) (RJ45 <--> RJ45)
	LU9GC3	RS485	VW3A8306R03 VW3A8306R10 VW3A8306R30	0.3 m (1 ft.) 1 m (3.3 ft.) 3 m (9.8 ft.) (RJ45 <--> RJ45)
	Quantum	RS232C	XBT Z9710 + XBT ZG939	2.5 m (16.4 ft.) (SUB-D25 <--> SUB-D9)
	Momentum		XBT Z9711 + XBT ZG939	2.5 m (16.4 ft.) (SUB-D25 <--> RJ45)

In Modbus slave applications, when power is first applied to the XBT N terminals, the XBT N terminals will issue noise on the bus for approximately 100 ms. This noise will disturb the communication of the equipment connected to the bus. Always apply power to the XBT N terminal first, before applying power to the master of the bus.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

When XBT N terminals are operated as Modbus slaves, always power-on these terminals before starting the master of the bus.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

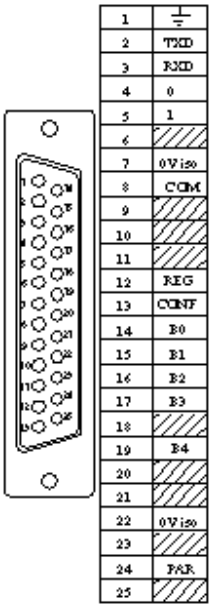
SUB-D25 Pin Connections

Overview

The following XBT terminals provide a SUB-D25 connector on their rear panels:

- XBT N401
- XBT N410
- XBT R411

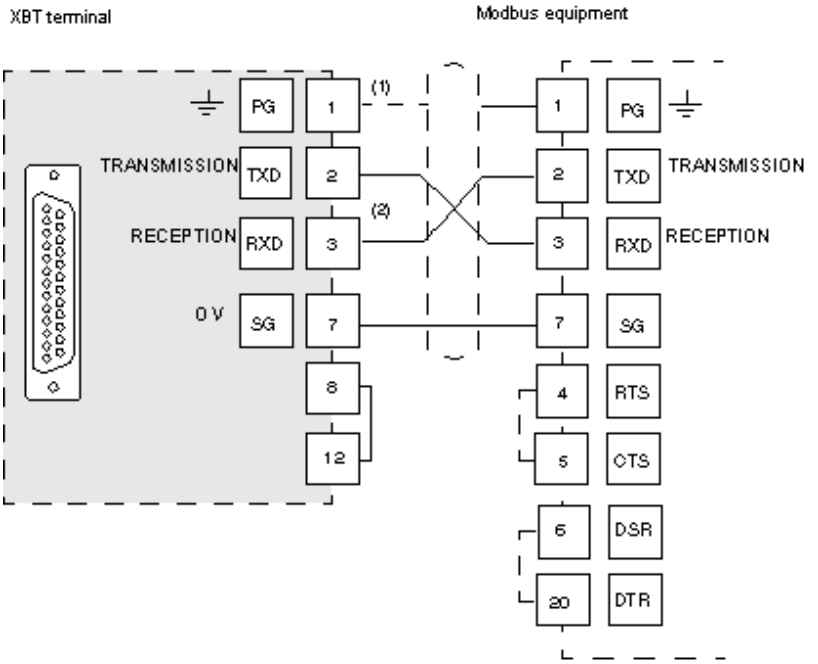
The SUB-D25 connector supports RS232 as well as RS485 lines. The pin assignment is shown in the following figure.



RS232 Cabling

The illustration below shows the cabling for RS232C equipment.

RS232C link example



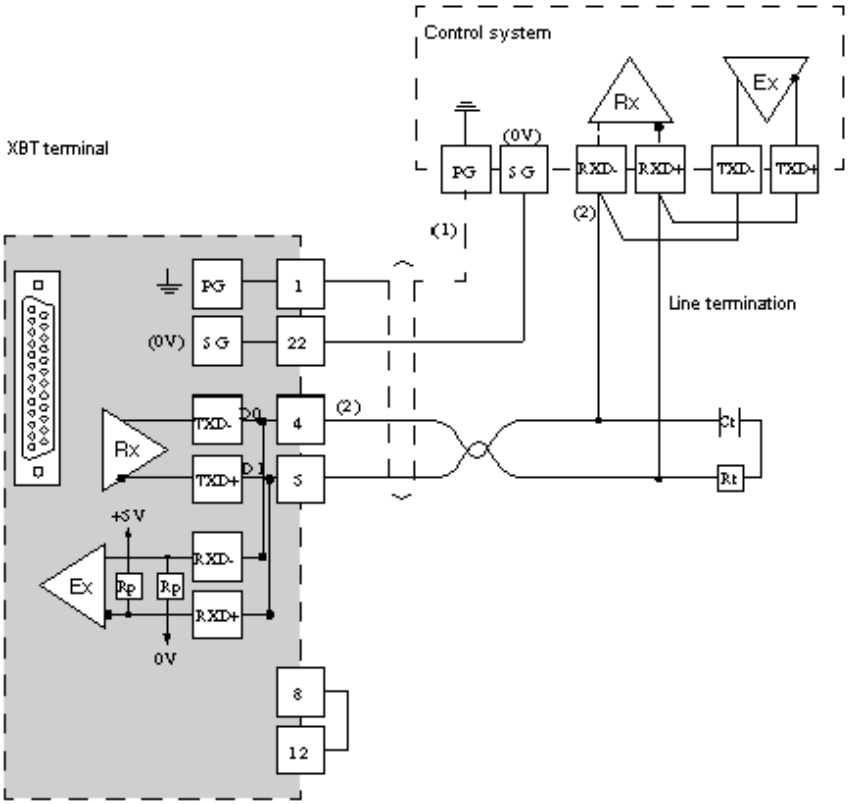
Legend

(1)	Connection of the shielding at both ends depends on any electrical restrictions affecting the installation.
(2)	In some configurations, it is not necessary to invert pins 2 and 3. Please refer to the documentation for the equipment being used.

RS485 Cabling

The illustration below shows the cabling for RS485 equipment.

RS485 link example



Legend

(1)	Connection of the shielding at both ends depends on any electrical restrictions affecting the installation.
(2)	If your automation systems provides connectors for 4-wire connections, wire the RXD and TXD pins as shown in the above figure to form a 2-wire connection.
(3)	Rp: Polarization resistors. The following polarization resistors are integrated in XBT N, XBT R and XBT RT: • XBT N: Rp = 4.7 kΩ • XBT R: Rp = 100 kΩ

RJ45 Pin Connections

Overview

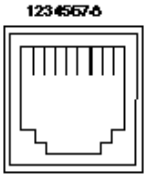
The following XBT RT terminal provides RJ45 connectors on its rear panels:

In industrial environments, it is compulsory to use a

- double shielded twisted pair cable with impedance 100 Ω 15 Ω (1...16 MHz),
- maximum attenuation 11.5 dB/100 m (11.5 dB/328 ft.),
- maximum length 100 m (328 ft.).

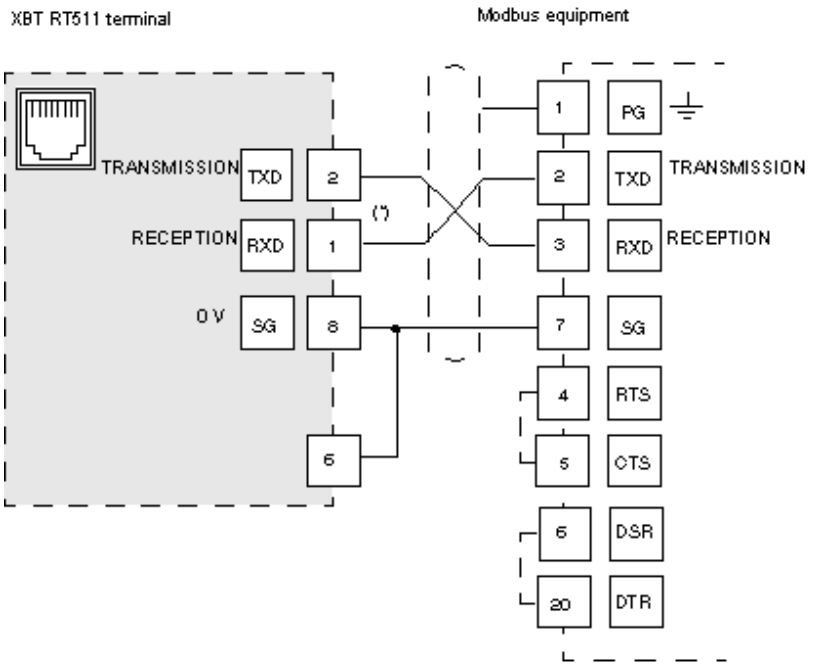
XBT RT511

Pin assignment of the RJ45 connector on XBT RT511 terminals

Representation	Pin	Signal	Comments
RJ45 	1	RXD	RXD RS232 signal
	2	TxD	TXD RS232 signal
	3	IN1	Input configuration signal
	4	D1	RS485 + signal
	5	D0	RS485 - signal
	6	IN2	Input operating signal
	7	-	-
	8	0 V ISO	0 V isolated

The illustration below shows the cabling for RS232C equipment.

RS232C link example

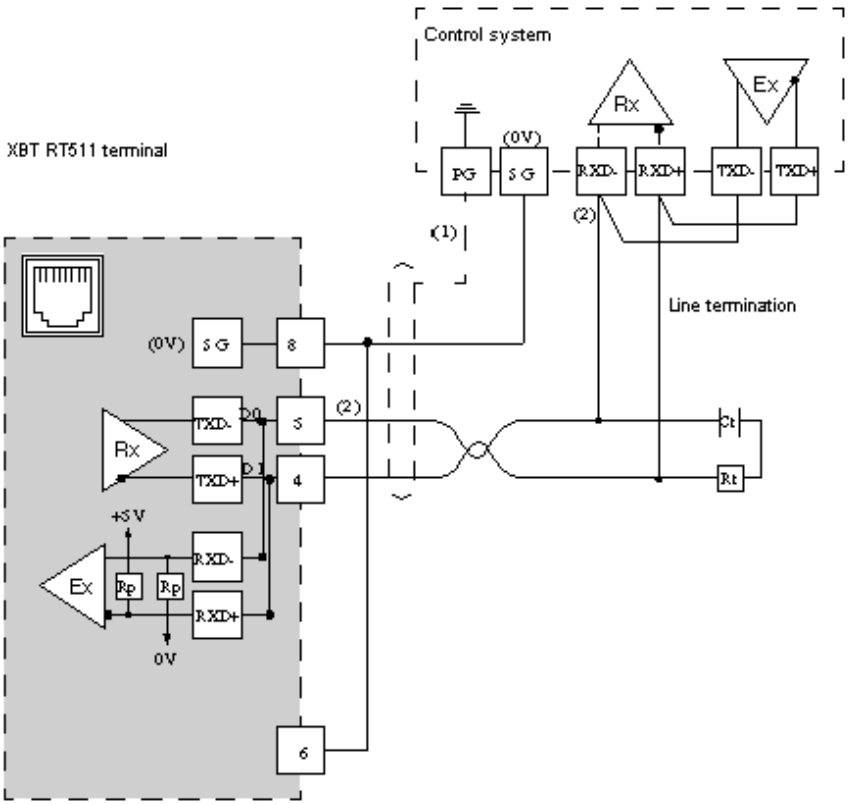


Legend

(*)	In some configurations, it is not necessary to invert pins 1 and 2. Please refer to the documentation for the equipment being used.

The illustration below shows the cabling for RS485 equipment.

RS485 link example



Legend

(1)	Connection of the shielding at both ends depends on any electrical restrictions affecting the installation.
(2)	If your control systems provides connectors for 4-wire connections, wire the RXD and TXD pins as shown in the above figure to form a 2-wire connection.
(3)	Rp: Polarization resistors: 100 kΩ

Diagnostics

5

XBT Detected Error Indication

Overview

XBT terminals indicate detected errors in 3 different ways

- by displaying question marks ?????? on alphanumerical fields
- by displaying crosses for graphic objects
- by displaying hash characters in alphanumerical fields
- by blinking alphanumerical fields
- by issuing system error messages

The following paragraphs list these three detected errors and their possible reasons.

Question Marks and Crosses

When question marks ?????? and crosses XXXXXX are displayed on the display of your XBT terminal, a transmission error has occurred. To correct this, check the following:

If...	Then ...
question marks are displayed	verify that all cables are correctly connected. If yu configured a time-out, verify that the master can access the XBT at least once during the configured time. If necessary, increase the time-out value, or set the time-out parameter to 0.
question marks are displayed	verify that the communication parameters set in the **Protocol - Modbus Slave** dialog box are identical for all equipment connected to the Modbus bus i.e. same transmission speed, same use of parity.

Hash Characters

Hash characters displayed in alphanumerical fields on your XBT terminal indicate that the value to be displayed is too long for this alphanumerical field and cannot completely be displayed. The value `100` can, for example, not be displayed in a 2-digit alphanumerical field. To correct this, enter a shorter value or adapt the size of the alpanumerical field so that it can display any of the possible values of the PLC variable.

Blinking Alphanumerical Fields

Blinking alphanumerical fields on your XBT terminal indicate that the value of this field has exceeded or fallen below a user-defined threshold.

System Error Messages

A variety of system error messages is by default configured for the terminals. All these standard system messages are assigned a panel number 200+x. A distinction is made between system error messages indicating communication interruptions and status messages provoked by inputs at the terminal.

These 2 message types differ by the numbers they are assigned and by the way they are displayed at the terminal as shown in the list below:

System Error Message Caused by:	System Error Message Numbers	Display Mode
Communication Interruptions	201 – 204	To indicate that a communicaton interruption has occurred, the message is displayed in a popup dialog box every 10 seconds.
Input at Terminal	241 – 258	The status message is displayed as a response to user input at the terminal.

Messages Caused by Communication Interruptions

Messages 201 to 204 are issued by the terminal to indicate that a communication interruption has occurred. They are displayed in a popup dialog every 10 seconds.

If...	Then ...
message `201: DIALOG TABLE AUTHORIZATION INCORRECT` is displayed	the authorization word in the dialog table does not have the expected value. (Refer to the Vijeo-Designer Lite online help for information on how this word is working.) To correct this, verify that:. • you are connected to the right PLC • the correct value has been written by the PLC in the authorization word of the dialog table located in the terminal memory.
message `203: DIALOG TABLE READING IMPOSSIBLE` is displayed	the read cycle from the dialog table of the PLC could not be ended. • too much load on the communication bus • EMC disturbances on the communication bus • The PLC has never read all the status words (XBT->PLC) of the dialog table since the XBT has been powered ON.

Messages Caused by Input at the Terminal

Messages 242 to 254 are issued by the XBT as a response to user input at the terminal. These messages are displayed directly after the operator has sent an incorrect command to the terminal and will persist until the user has corrected the entered command or value. Messages 255 to 258 are status messages displayed after the user has initiated an operation at the terminal to indicate that it has (or has not) been accepted and is in progress.

If...	Then ...
messages `243` to `249` are displayed	correct the value or command you have entered as indicated by the condition message.
message `250: LANGUAGE IMPOSED BY PLC` is displayed	the PLC forces the terminal to use a language. This language cannot be changed by the operator. For more information see the Vijeo-Designer Lite online help, functions of the dialog table.
messages `251` or `252` are displayed	correct the value or command you have entered as indicated by the condition message.
message `253: PASSWORD IMPOSED BY PLC` is displayed	you cannot change the password at the terminal because it is forced by the PLC. For more information see the Vijeo-Designer Lite online help, functions of the dialog table.

If...	Then ...
message `254: PROTECTED ACCESS PAGE` is displayed	you are trying to access a page that is password protected but you do not have the required authorization level.
messages `255` to `258` are displayed	the commands you entered at the terminal are executed or not executed, as indicated in these status messages.

Bandwidth Principle

6

Overview

This chapter describes the operating principle and the calculating bandwidth usage.

What's in this Chapter?

This chapter contains the following topics:

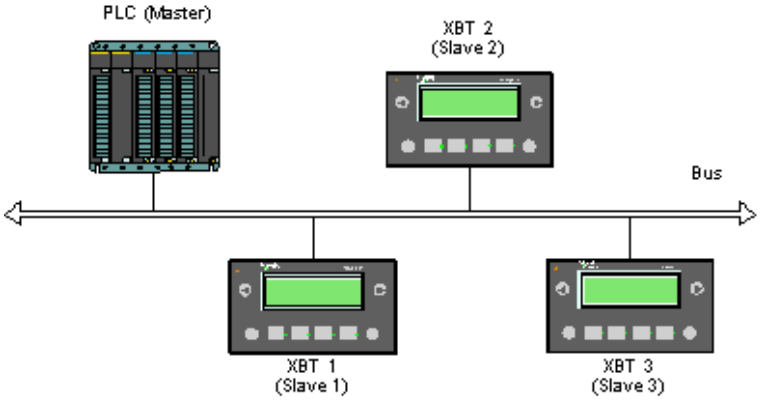
Topic	Page
General Operating Principle	52
Calculating Bandwidth Usage	54
Tips	58

General Operating Principle

Connection Diagram

The Modbus slave protocol operates in point-to-point or multidrop mode.

The PLC is connected to 1 or more terminals.



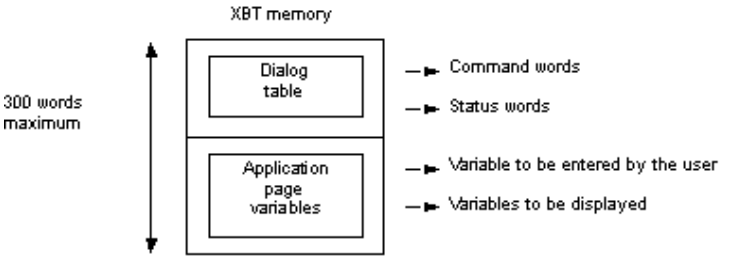
Operating Principle

Data exchanges between the terminals and the PLC are made in data-transmission cycles, during which the PLC will read and write to the XBT memory (for example, a PLC can read the values every 300 ms in the XBT memory).

The PLC will carry out the following operations:

- writing in the dialog table (command words)
- reading words from the dialog table (status words)
- writing variables (display variables)
- reading variables (variables entered by the user)

Operating principle



Each request transmission by the PLC results in a certain level of bandwidth usage. Therefore, before a communication architecture can be set up, the rate of bandwidth usage must be calculated to prevent the possibility of saturation.

General Reminders

Reminders and examples

Reminder	Example
For a transmission speed of 19,200 bit/s, the transmission time for a word is approximately 1 ms.	–
A PLC sending to a terminal a request to write n words requires • for sending: 9 bytes + 2 x n bytes • 8 bytes for acknowledgment	(see *Writing of n Words, page 61*).
A PLC sending to a terminal a request to read n words requires • 8 bytes for sending • for the answer: 5 bytes + 2 x n bytes	(see *Reading of n Output or Internal Words, page 62*).
One word = 2 bytes	Therefore, for example, sending 1 write word requires 17 + 2 = 19 bytes

Calculating Bandwidth Usage

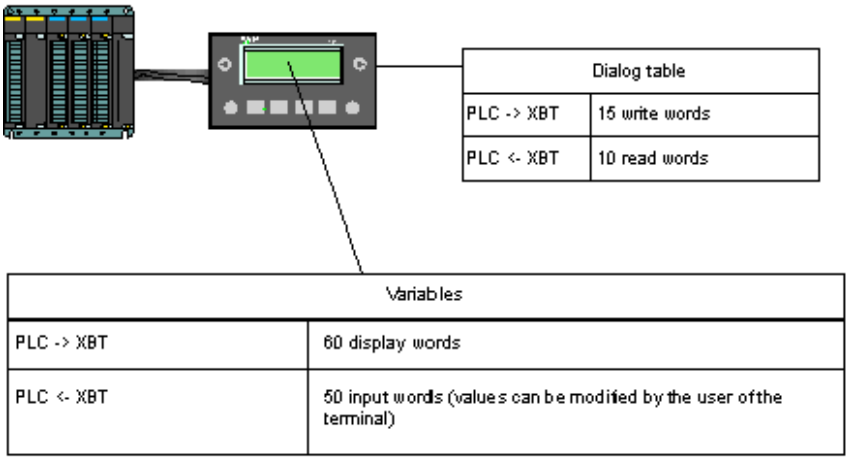
Overview

The bandwidth specifies the quantity of data, which can circulate on the network per second. This depends on several parameters, such as the transmission speed and the number of items of equipment connected to the network.

To find out how much of the bandwidth is used, calculate the time it takes to send the data during each cycle. To do this, convert the data rate (in bit/s) into the time during which the bandwidth is occupied.

Example of Calculating Bandwidth Usage in Point-To-Point Mode

Hypothesis: Say a terminal is connected to a PLC in point-to-point mode.



The dialog table contains 25 words, with a cycle of 300 ms (terminal default value).

Write Request	**15 words** PLC -> XBT
Read Request	**10 words** PLC <- XBT

Writing and displaying variables

60 words refreshed every 300 ms. Of these 60 words, 50 can be modified by the user.

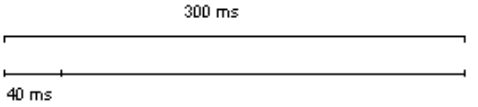
Display	**60 words** PLC -> XBT
Write (words which value can be modified by the user)	**50 words** PLC <- XBT

Calculating how much of the bandwidth is used by the dialog table

We will apply the following formula:

No. of data bytes + bytes of the request + bytes of the answer

Say in our example

30 + 9 + 8 = 47	**47 bytes** for the write request
20 + 8 + 5 = 33	**33 bytes** for the read request
A word is assumed to be sent in 1 ms (at a speed of 19,200 baud). Knowing that 1 word = 2 bytes, we get:	
(47 + 33) : 2 = 40	a transmission time of approximately 40 ms for the dialog table  The dialog table will therefore consume approximately 13% of the bandwidth.

Calculating how much of the bandwidth is used by the variables

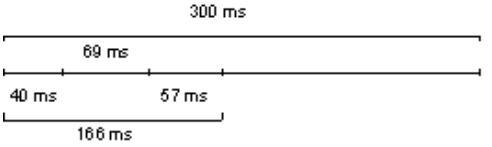
To write into the terminal the variables to be displayed, we will have bandwidth usage of:

60 words = 120 bytes + 9 bytes + 8 bytes = 137 bytes	a transmission time of approximately: 69 ms

To read from the terminal the variables that a user can modify, we will have bandwidth usage of:

50 words = 100 bytes + 8 bytes + 5 bytes = 113 bytes	a transmission time of approximately: 57 ms

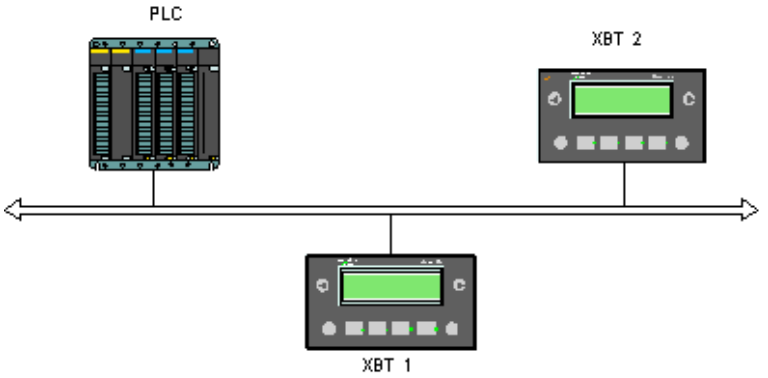
We will have a total consumption of 166 ms (40 + 69 + 57) of the bandwidth 300 ms (i.e., approximately 55% of the bandwidth).



At a speed of 9,600 baud, the bandwidth consumption would double. Consumption would therefore be 332 ms rather than 166 ms. The bandwidth would then saturated (332 ms for 300 ms maximum).

Example of Calculating Bandwidth Usage in Multidrop Mode

We want to set up an architecture comprising 1 PLC and 2 terminals.



The 2 dialog tables are made up as follows.

First dialog table (XBT 1)

Write Request	**5 words** PLC -> XBT
Read Request	**5 words** PLC <- XBT

Second dialog table (XBT 2)

Write Request	**10 words** PLC -> XBT
Read Request	**10 words** PLC <- XBT

Writing and displaying variables with the XBT 1 terminal

10 words refreshed every 300 ms. Of these 10 words, 5 can be modified by the user.

Display	**10 words** PLC -> XBT
Write (word which value can be modified by the user)	**5 words** PLC <- XBT

Writing and displaying variables with the XBT 2 terminal

30 words refreshed every 300 ms. Of these 30 words, 20 can be modified by the user.

Display	**30 words** PLC -> XBT
Write (word which value can be modified by the user)	**20 words** PLC <- XBT

Calculating how much of the bandwidth is used by the dialog tables

XBT 1 terminal dialog table

(10 + 9 + 8) + (10 + 8 + 5) = 50 bytes	The transmission time will be approximately 25 ms for this dialog table.

XBT 2 terminal dialog table

(20 + 9 + 8) + (20 + 8 + 5) = 70 bytes	The transmission time will be approximately 35 ms for this dialog table.

Calculating how much of the bandwidth is used by the variables

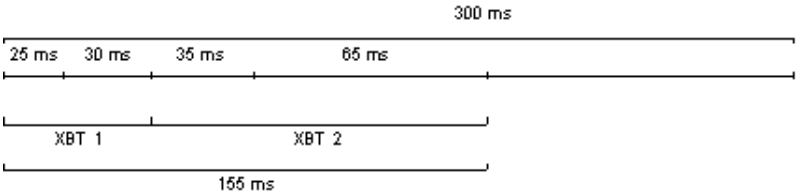
XBT 1 terminal variables (display and write)

(20 + 9 + 8) + (10 + 8 + 5) = 60 bytes	a transmission time of approximately 30 ms

XBT 2 terminal variables (display and write)

(60 + 9 + 8) + (40 + 8 + 5) = 130 bytes	a transmission time of approximately 65 ms

The bandwidth consumption can be represented as follows:



We have a total consumption of 155 ms (25 + 35 + 30 + 65) of the bandwidth 300 ms (i.e., approximately 52% of the bandwidth).

As in the example in point-to-point mode, we see that if we reduce the speed to 9,600 baud, the bandwidth would be saturated (310 ms for 300 ms maximum).

Tips

Tips for the User

The previous examples demonstrate the following:

- The more terminals are added, the less bandwidth remains.
- The more values there are to display, the higher the bandwidth consumption by the write operation.

There are therefore a number of possibilities for freeing up the bandwidth:

- increase the transmission speed (depends on the quality of the network and the connected equipment)
- reduce the number of words in the dialog table
- reduce the number of words needing to be read or written by the PLC
- reduce the refresh speed for the display
- reduce the cycle speed of the dialog table

Appendices



Communication Requests

A

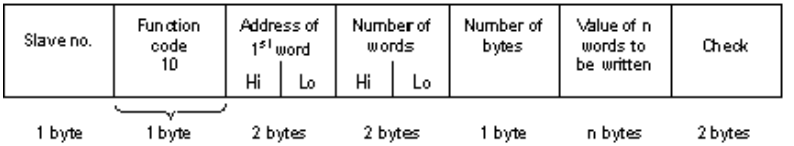
Communication Requests

Overview

The function code is in hexadecimal format.

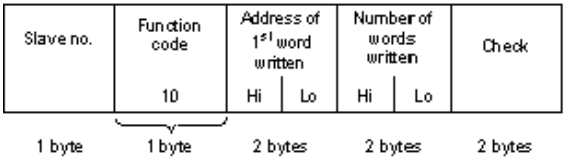
Writing of n Words

Request



Address of 1st Word	same addressing field as for the read request
Number of Words	125 words
Number of Bytes	twice the number of words
Value of Words to be Written	H'0000' to H'FFFF'

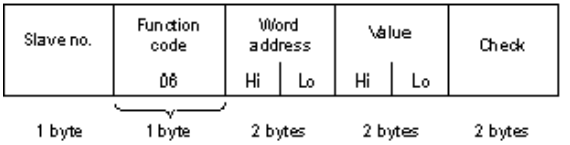
Response



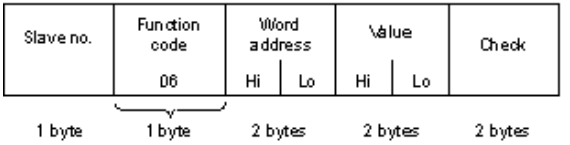
Slave number	same as request
Address of 1st Word Written	same as request
Number of Words Written	same as request

Writing of 1 Output or Internal Word

Request

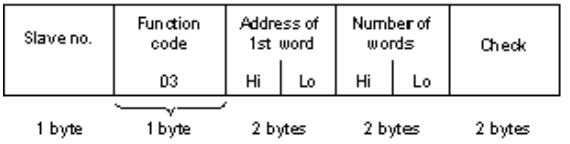


Response



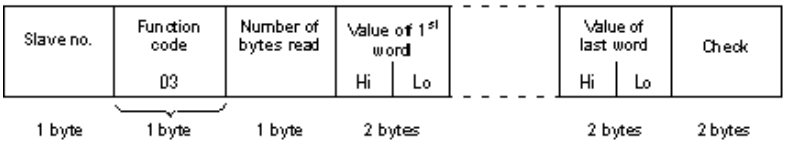
Reading of n Output or Internal Words

Request



Address of 1st Word	Corresponds to the address of the 1st word to be read in the slave.
Number of Words	125 words

Response



Slave number	same as request
Number of Bytes Read	twice the number of words read
Value of Words Read	H'0000' to H'FFFF'

Reading and Resetting Counters

Request

Slave no.	Function code	Sub-function	Data (d)	Check
	08	00xx	0000	
1 byte	1 byte	2 bytes	2 bytes	2 bytes

One sub-function code for each function

Reading Counter 1	0x000B
Reading Counter 2	0x000C
...	...
Reading Counter 8	0x0012
Counter Reset	0x000A

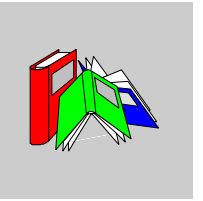
Response

Slave no.	Function code	Sub-function	Data (d)	Check
	08	00xx		
1 byte	1 byte	2 bytes	2 bytes	2 bytes

Functions Supported

		Sub-Function		Type of Functions
Hex	**Dec**	**Hex**	**Dec**	
03	03	–	–	reading n output or internal words initiated by the master
06	06	–	–	writing 1 output or internal word
08	08	00XX	00XX	reading and resetting counters initiated by the master
10	16	–	–	writing n words
2B	43	0E	14	read device identification

Glossary



A

ASCII

American standard code for information interchange = data transmission mode in Modbus communications

AWG

American wire gauge (wire diameter)

C

CRC

cyclic redundancy checking

CTS

clear to send (data transmission signal)

D

DSR

data set ready (data transmission signal)

DTR

data terminal ready (data transmission signal)

E

EMC

electromagnetic compliance

L

LRC

longitudinal redundancy checking

M

Modbus SL

Modbus serial line

O

OSI Model

open system interconnection model

P

PDU

protocol data unit

R

RJ-45

registered jack = standardized physical interface

RS232

recommended standard for connecting serial devices = EIA/TIA 232

RS485

recommended standard for connecting serial devices = EIA/TIA 485

RTS

request to send (data transmission signal)

RTU

remote terminal unit = data transmission mode in Modbus communications

RXD

receiving data (data transmission signal)

T

TXD

transmitting data (data transmission signal)

Index



A

addressing
    Modbus slave protocol, *26*

B

bandwidth
    Modbus slave protocol, *54*
biasing, *24*

C

cable length, *22*
cables
    Modbus slave protocol, *38*
capacitor, *23*
communication principle
    master / slave, *13*
communication requests
    Modbus slave protocol, *61*
configuration
    Modbus slave protocol, *32*
connection diagram
    Modbus slave protocol, *52*

D

data types
    Modbus slave protocol, *35*
diagnostics
    Modbus slave protocol, *47*

F

frame
    incomplete, *19*
frame description
    Modbus master protocol, *20*
frame segment, *20*

G

grounding, *22*, *22*

I

incomplete frame, *19*

L

length of cable, *22*

M

master / slave communication principle, *13*
Modbus master protocol
    example of a serial Modbus RTU bus, *21*
    frame description, *20*
    OSI model, *15*
    RTU framing, *18*
    RTU transmission mode, *17*

Modbus slave protocol
addressing, *26*
cables, *38*
calculating bandwidth usage, *54*
communication requests, *61*
connection diagram, *52*
data types, *35*
diagnostics, *47*
operating principles, *11*, *52*
software configuration, *30*

O

objects
Modbus slave protocol, *35*
operating principles
Modbus slave protocol, *11*, *52*
OSI model
Modbus master protocol, *15*

P

pin connections
RJ45, *43*
SUB-D25, *40*
polarization, *24*
protocol configuration, *32*

R

RC termination, *23*
repeater, *22*
resistor, *23*
RJ45
pin connections, *43*
RS232 cabling, *40*
RS485 cabling, *41*
RTU framing
Modbus master protocol, *18*
RTU transmission mode
Modbus master protocol, *17*

S

software configuration
Modbus slave protocol, *30*
SUB-D25
pin connections, *40*

T

termination, *23*

V

variable types
Modbus slave protocol, *35*