Schneider Electric

Modbus Master Protocol
XBT N/R/RT

03/2009

33003986.02

www.schneider-electric.com

Schneider Electric assumes no responsibility for any errors that may appear in this document. If you have any suggestions for improvements 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 any 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 help ensure compliance with documented system data, only the manufacturer should perform repairs to components.

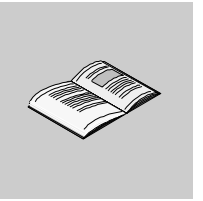
When devices are used for applications with technical safety requirements, the relevant instructions must be followed.

Failure to use Schneider Electric software or approved software with our hardware products may result in injury, harm, or improper operating results.

Failure to observe this information can result in injury or equipment damage.

© 2009 Schneider Electric. All rights reserved.

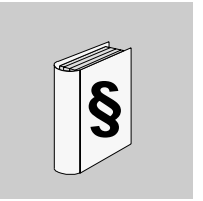
Table of Contents



Safety Information **5**
About the Book **7**
Chapter 1 Operating Principle **9**
General Information on Bus Communications 11
Master / Slave Communication Principle 12
Communication according to the OSI Model 14
Modbus RTU Transmission Mode 16
Modbus RTU Framing 17
Modbus Frame Description 19
Example of a Serial Modbus RTU Communication Bus 20
Cable Length and Grounding 21
RC Termination 22
Line Polarization 23
Chapter 2 Software Configuration **25**
Vijeo-Designer Lite 26
Protocol - Modbus Dialog Box 28
Configuring Slave Addresses 30
Modbus Equipment Address Dialog Box 32
Chapter 3 Variable Types Supported **33**
Variable Types for Modbus Master 33
Chapter 4 Cables and Connectors **35**
Cables 36
SUB-D25 Pin Connections 38
RJ45 Pin Connections 41
Chapter 5 Diagnostics **47**
XBT Detected Error Indication 47
Appendices **53**
Appendix A Modbus Master Requests **55**
Modbus Master Requests 55

Appendix B RS232/485 Recommendations 63
RS232 Recommendations 64
RS485 Recommendations 65
Glossary 67
Index 71

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.

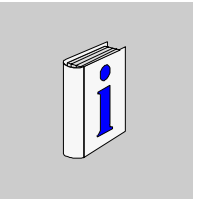
⚠ CAUTION
CAUTION indicates a potentially hazardous situation which, if not avoided, **can result in** minor or moderate injury.

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.

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 Master 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
Modbus Protocol Reference Guide	PI-MBUS-300 available at www.modbus.org
XBT N/R/RT Instruction Sheet	W916810140111 A08
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

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 master 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	12

Topic	Page
Communication according to the OSI Model	14
Modbus RTU Transmission Mode	16
Modbus RTU Framing	17
Modbus Frame Description	19
Example of a Serial Modbus RTU Communication Bus	20
Cable Length and Grounding	21
RC Termination	22
Line Polarization	23

General Information on Bus Communications

Overview

The 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 master.

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 master:



1 Modbus master: XBT RT
2 Modbus slave: Twido PLC
3 Modbus bus

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

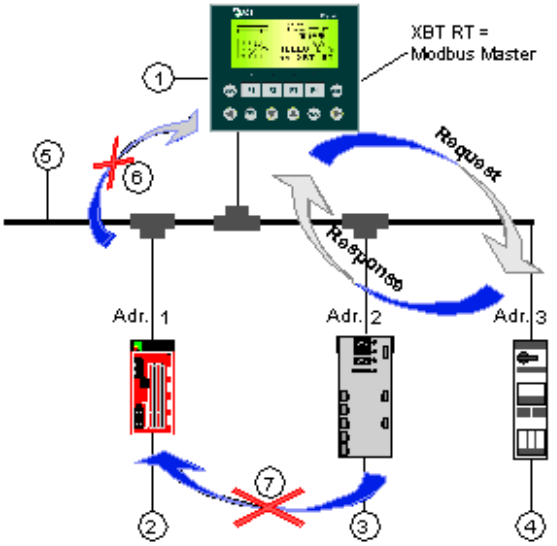
The master / slave principle is characterized as follows:

- Only one master is connected to the bus at a time.
- One or several slave nodes can be connected to the same serial bus.
- Only the master is allowed to initiate communication, i.e. to send requests to the slave nodes.
- In Modbus communications, the master can only initiate one Modbus transaction at the same time.
- In Modbus communications, the master can address each slave node individually (unicast mode) or all slaves simultaneously (broadcast mode).
- The slave nodes can only answer requests they received from the master.
- The slave nodes are not allowed to initiate communication, neither to the master nor to any other slave nodes.
- In Modbus communications, the slave nodes 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 Master in Modbus Applications

In Modbus master applications, the XBT terminal acts as master device, i.e. it provides the client role, whereas the slave devices act as servers.

Master / slave communication



1 XBT RT
2 XPSMF40 Safety PLC
3 XPSMF30 Safety PLC
4 Tesys U
5 Modbus SL bus
6 slave nodes cannot initiate the communication
7 slave nodes cannot communicate with other slave nodes

Communication according to the OSI Model

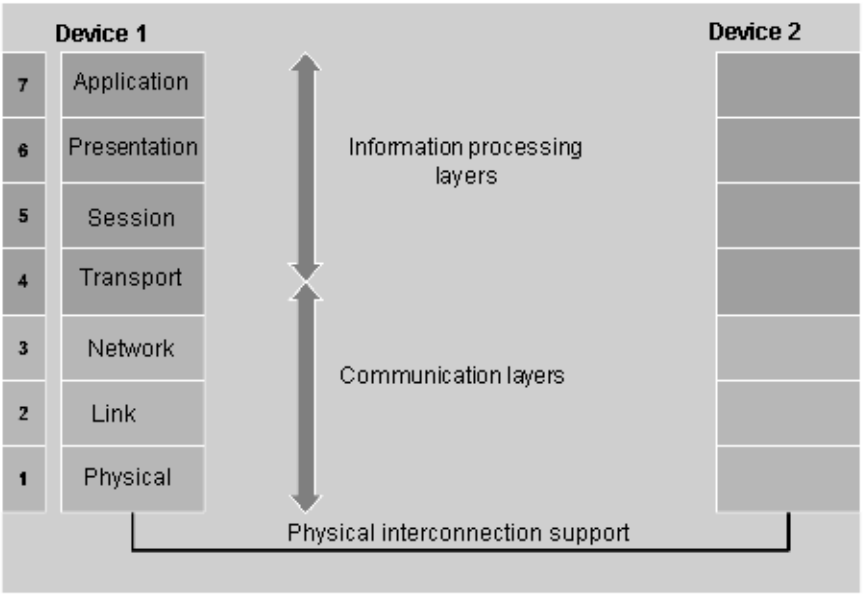
At a Glance

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

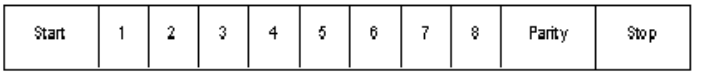
Each byte (11 bits) has the following format

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.

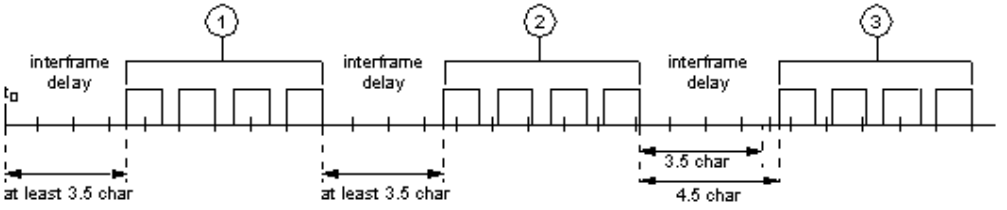
RTU message frame

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

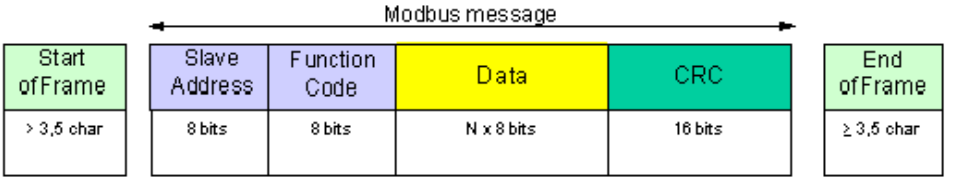
Individual frames are separated by a silent interval, also called interframe delay, of at least 3.5 character times. The following figure provides an overview of 3 frames being separated by an interframe delay of at least 3.5 character times.

Message frames separated by silent times



1 Frame 1
2 Frame 2
3 Frame 3

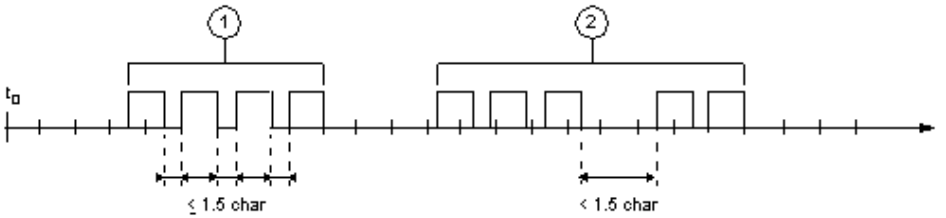
RTU message frame with start and end silent times



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



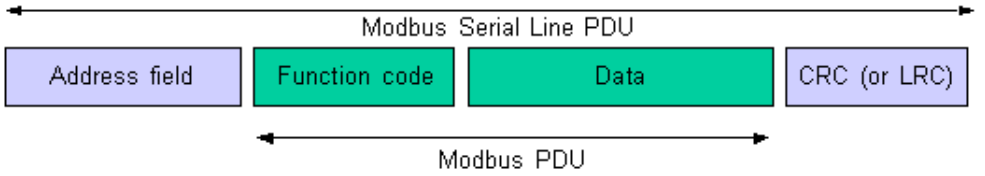
1 Frame 1 OK
2 Frame 2 Not OK

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

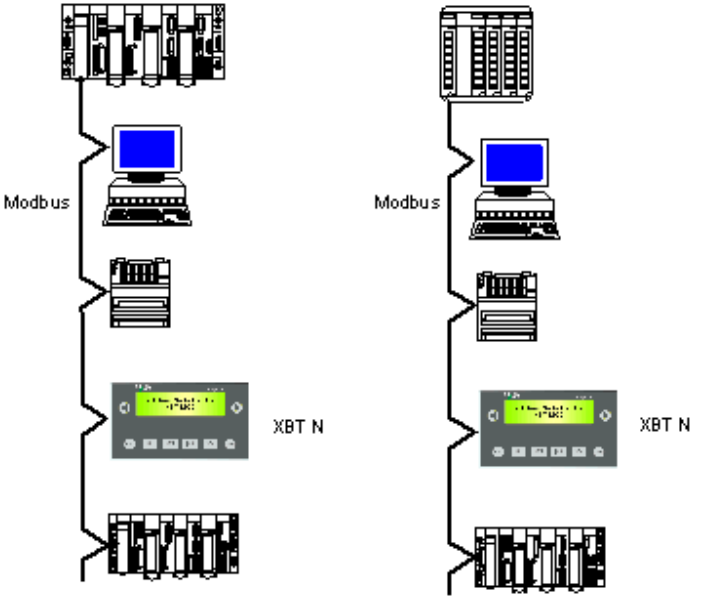
Example of a Serial Modbus RTU Communication Bus

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 23*.
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

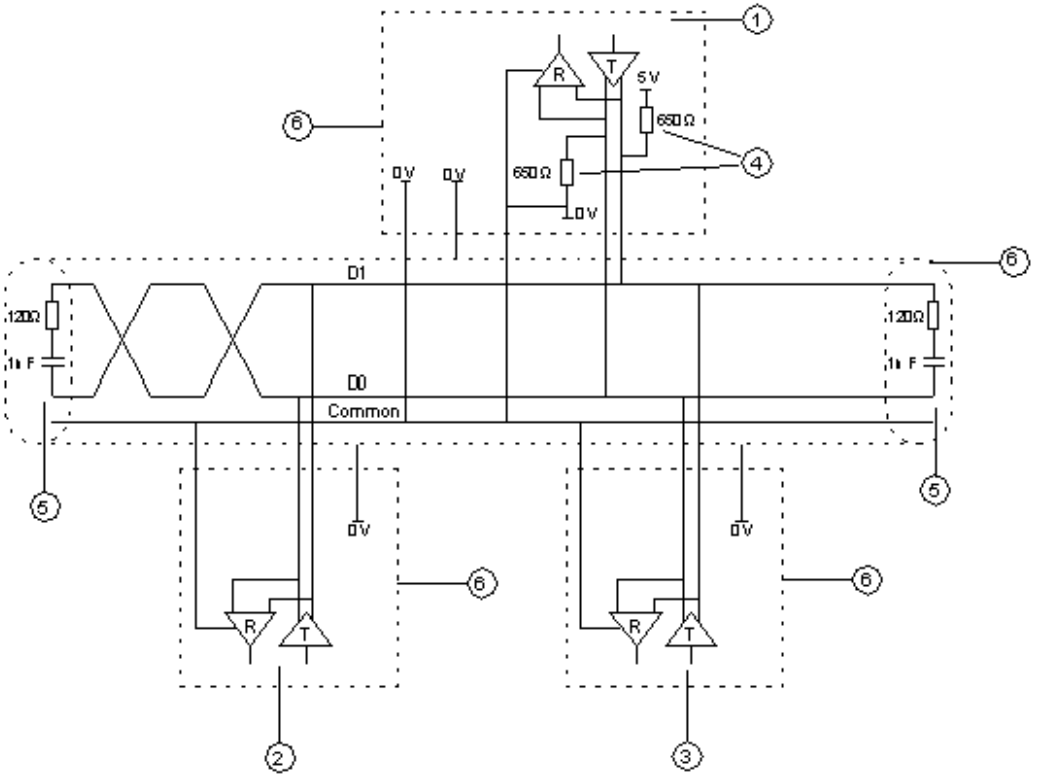
To provide proper line polarization, proceed as follows:

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

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 master.

⚠ 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.

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Vijeo-Designer Lite	26
Protocol - Modbus Dialog Box	28
Configuring Slave Addresses	30
Modbus Equipment Address Dialog Box	32

Vijeo-Designer Lite

Overview

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

⚠ WARNING

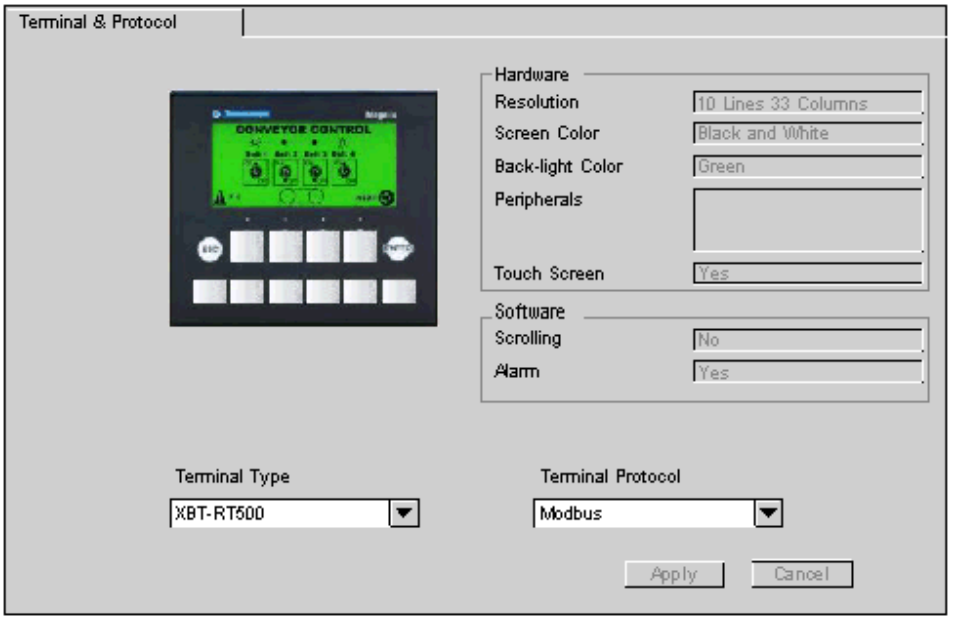
INCOMPATIBLE SOFTWARE

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 Dialog Box

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

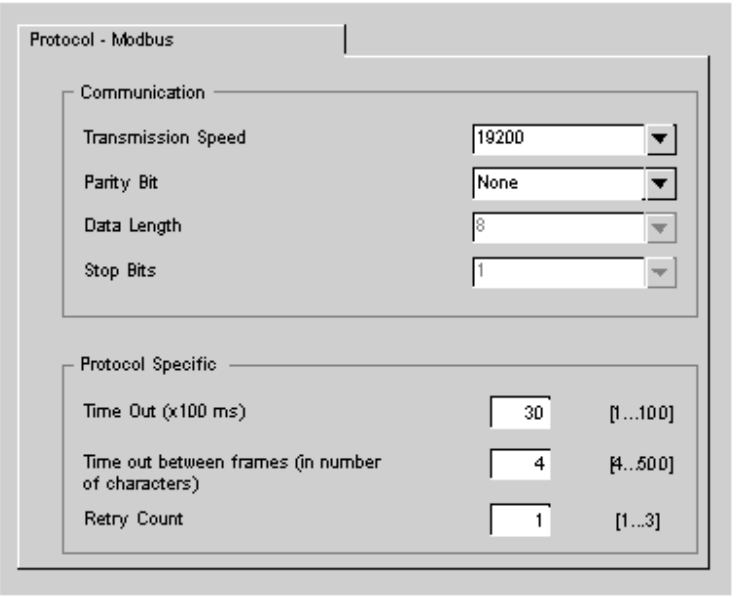
Step	Action
1	Start Vijeo-Designer Lite. To create a new application, continue with step 2, if you have already created a Modbus application, skip steps 2 and 3 and execute step 4.
2	From the application browser on the left-hand side of the Vijeo-Designer Lite window select the item **Configuration**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

Protocol - Modbus Dialog Box

Purpose

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

Element	Description
Stop Bits	You cannot edit this parameter because the XBT terminals only support 1 stop bit. The combination of no parity with only 1 stop bit is therefore possible, even though this is not conform to the Modbus standard.
Protocol Specific	
Time out (x100 ms)	Enter a time (in multiple of 100 ms) that has to elapse after the master has sent a request to a slave. If the master does not receive a response within this time, it will resend the request. The number of times the master will resend the request is defined in the **Retry Count** parameter.
Time out between frames (in number of characters)	Enter a time (in number of characters) that defines the time that has to elapse without any data on the Modbus bus. This means that, after the master has received a response from a slave, the master must wait until this time has elapsed until it sends another request to a slave.
Retry Count	Enter the number of times the master should try to send a request to a slave that is not responding.

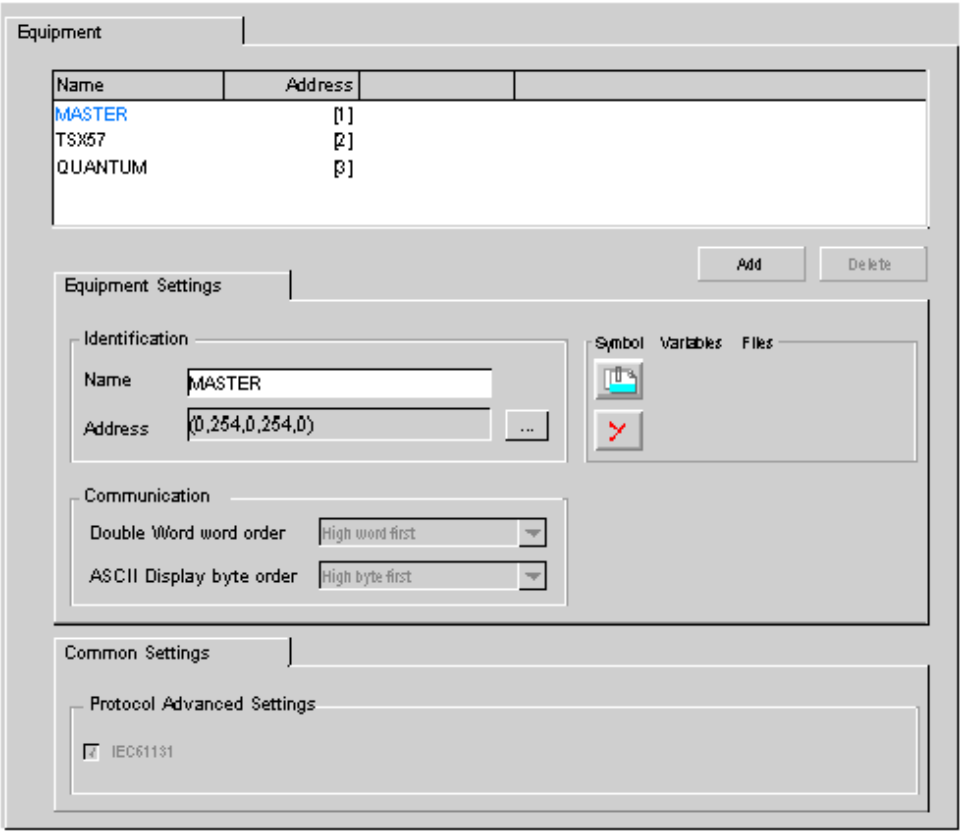
Configuring Slave Addresses

Overview

Use the Vijeo-Designer Lite software to configure addresses for the slave devices your XBT terminal should communicate with.

Opening the Modbus Equipment Address Dialog Box

To open the **Modbus Equipment Address** dialog box in Vijeo-Designer Lite for configuring slave addresses, proceed as follows:

Step	Action
1	Start Vijeo-Designer Lite.
2	From the application browser on the left-hand side of the Vijeo-Designer Lite window select the item **Equipments**. **Result**: The following dialog box will be displayed on the right-hand side of the Vijeo-Designer Lite window. 
3	In the **Equipments** dialog box select a slave device from the list.
4	In the **Equipment Settings** → **Identification** box below click the **...** button right to the **Address** text box. **Result**: The dialog box **Equipment Address** will be displayed where you can configure a Modbus address for the selected slave device.

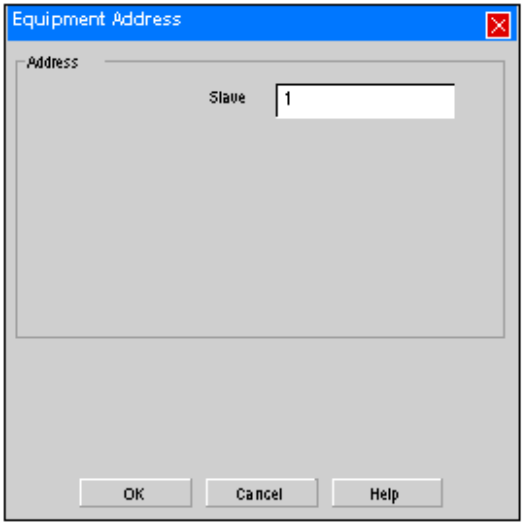
Modbus Equipment Address Dialog Box

Purpose

Use this dialog box to configure the address of equipments connected to the network.

Representation

Modbus equipment address



Elements of the dialog

Element	Description
Slave	Enter a unique Modbus address between 1 and 247 for the selected slave device.
OK button	Click the **OK** button to assign the entered address to the selected slave device.
Cancel button	Click the **Cancel** button to discard the changes and to close the dialog box.
Help button	Click the **Help** button to open the Vijeo-Designer Lite online help.

Variable Types Supported

3

Variable Types for Modbus Master

Table of Variable Types Supported by the XBT

The following table lists all Modbus variables XBT terminals can access.

Variable Type Supported	Regular Modbus Syntax	IEC 61131-2 Syntax
Bit of Word	40001+i,j	%MWi:Xj
Internal Bit and Output Bit	00001+i	%Mi
Input Bit	10001+i	not supported by the XBT
Input Register	30001+i	not supported by the XBT
Word, String	40001+i	%MWi
Double Word	40001+i	%MDi
Floating Point*	40001+i	%MFi
Mnemonic Identifier	i: (0...65535) j: (0...F)	

***** compliant with standard IEEE754

Note About the Protocols

Double words and floating point words are managed as follows:

Most Significant Byte	%MWi+1 or 40001+(i+1)
Least Significant Byte	%MWi or 40001+i

NOTE: Check that the connected device uses the same format.

Cables and Connectors

4

Overview

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

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Cables	36
SUB-D25 Pin Connections	38
RJ45 Pin Connections	41

Cables

Overview

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

Technical Data

Connected Device	XBT Type	Cable Reference	Length and Type
Twido	XBT N200/N400** XBT R400 XBT RT500/RT511	XBT Z9780	2.5 m (8.2 ft.) (RJ45 <--> MiniDin)
	XBT N401/N410 XBT R410/R411	XBT Z968 (straight) XBT Z9680 (angled)	2.5 m (8.2 ft) 2.5 m (8.2 ft) (SUB-D25 <--> MiniDin)
Modicon M340	XBT N200/N400** XBT R400 XBT RT500/RT511	XBT Z9980	2.5 m (8.2 ft.) RJ45 <--> RJ45
Modicon Quantum 984	XBT N401/N410 XBT R410/R411 XBT RT500/511	XBT Z9710*	2.5 m (8.2 ft) (SUB-D25 <--> SUB-D9)
Modicon Momentum M1	XBT N401/N410 XBT R410/R411 XBT RT500/511	XBT Z9711*	2.5 m (8.2 ft) (SUB-D25 <--> RJ45)
Modbus Ethernet Gateway (174CEV30010)	XBT N401/N410 XBT R410/R411	XBT Z9713	2.5 m (8.2 ft) (SUB-D25 <--> RJ45)
Advantys STB	XBT N401/N410 XBT R410/R411	XBT Z988	2.5 m (8.2 ft) (SUB-D25 <--> HE13)
	XBT RT500/511	XBT Z9715 (+ XBT ZRTPW for XBT RT500)	2.5 m (8.2 ft) (RJ45 <--> HE13)
Tesys Model U	XBT NU400	XBT Z938	1.8 m (5.9 ft.) (SUB-D25 <--> RJ45)
Altivar	XBT N401/N410 XBT R410/R411	XBT Z938	1.8 m (5.9 ft.) (SUB-D25 <--> RJ45)
Zelio with communication module			

Connected Device	XBT Type	Cable Reference	Length and Type
TSX SCA62/SCA64 (multipoint cabling tap)	XBT N401 XBT R410/R411 XBT RT 511	XBT Z908*	1.8 m (5.9 ft.) (SUB-D25 <--> SUB-D15)
Modbus devices multipoint	XBT RT511	VW3A8306R03 VW3A8306R10 VW3A8306R30	0.3 m (1 ft.) 1 m (3.3 ft.) 3 m (9.8 ft.) (RJ45 <--> RJ45)

*
- **XBT RT 511**: you must add a XBT ZG939 cable adapter (SUB-D25 <-> RJ45) (+ XBT ZRTPW for XBT RT500.)
- **XBT RT 500**: you must add a XBT ZG939 cable adapter (SUB-D25 <-> RJ45) and a XBT ZRTPW for power supply.

** Depending on your hardware (for XBT N200/N400 display units), you may need to use a different cable if you have connection problems (see *XBT Detected Error Indication, page 47*).

NOTE: When connecting XBT terminals to Twido or M340 PLCs using Schneider cables with MiniDin connectors, i.e. XBT Z968 (straight), XBT Z9680 (angled) and XBT Z9780, which do not provide any jumper on the MiniDin side (PLC side), the communication parameters are automatically set to the following settings:

- 19,200 bit/s
- no parity
- RTU (8 bits)
- 1stop bit on the base port

The communication parameters can only deviate from these settings if the following two conditions apply:

- a jumper is present AND
- another speed has been configured with the configuration tool of the PLC

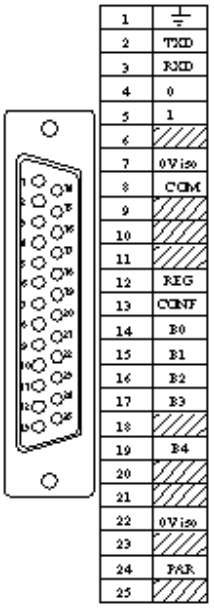
SUB-D25 Pin Connections

Overview

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

- XBT N401
- XBT NU400
- XBT N410
- XBT R410
- XBT R411

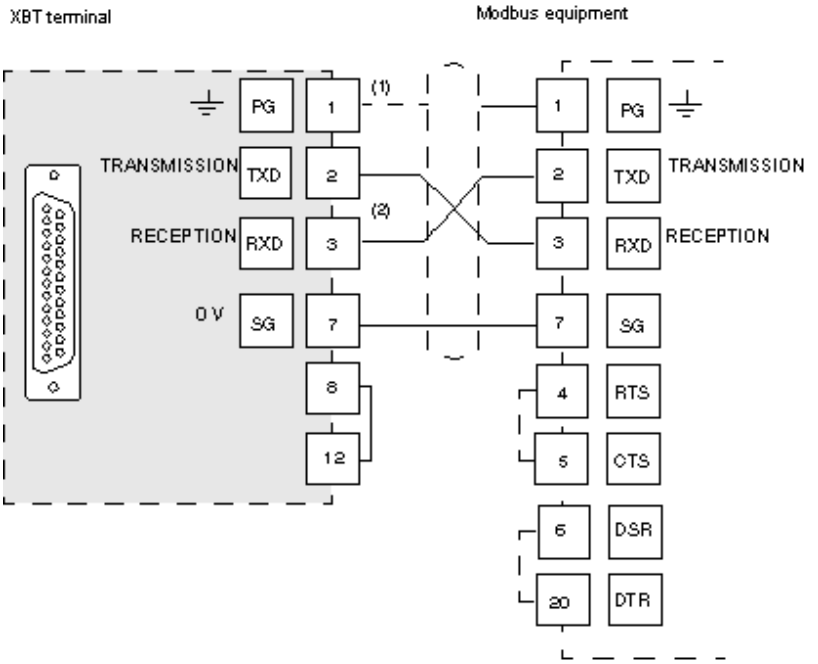
The SUB-D25 connector supports RS232 as well as RS485 lines. The pin assignment is shown in the following figure.



RS232 Cabling

The illustration below shows the cabling for RS232C equipment.

RS232C link example



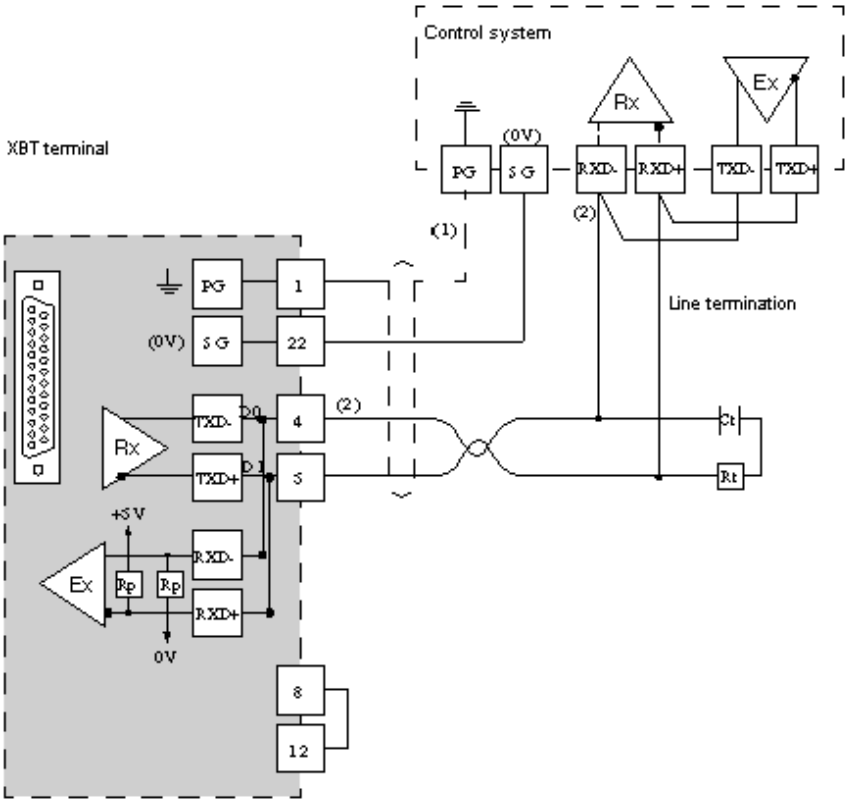
Legend

(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 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. The following polarization resistors are integrated in XBT N and XBT R / RT: • XBT N: Rp = 4.7 kΩ • XBT R: Rp = 470 Ω

RJ45 Pin Connections

Overview

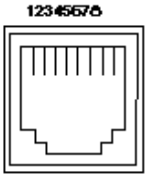
The following XBT terminals provide RJ45 connectors on their rear panels:

- XBT N200
- XBT N400
- XBT R400
- XBT RT500
- XBT RT511

These XBT terminals are equipped with different RJ45 connectors. Their individual pin assignments are listed in the following tables.

XBT N200/XBT N400/XBT R400

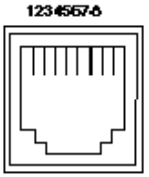
Pin assignment of the RJ45 connector on XBT N/R terminals

Representation	Pin	Signal	Comments
RJ45 	1	CONF	Reserved
	2	TxD	TXD RS232 signal
	3	RXD	RXD RS232 signal
	4	D1	RS485 + signal
	5	D0	RS485 - signal
	6	REG	Reserved
	7	+5 V	Power supply: +5 V, I = 200 mA
	8	GND	Common non isolated

NOTE: Some XBT N200 or XBT N400 hardware types require different cables. If you have connection problems with these terminals, refer to *Diagnostics, page 47*.

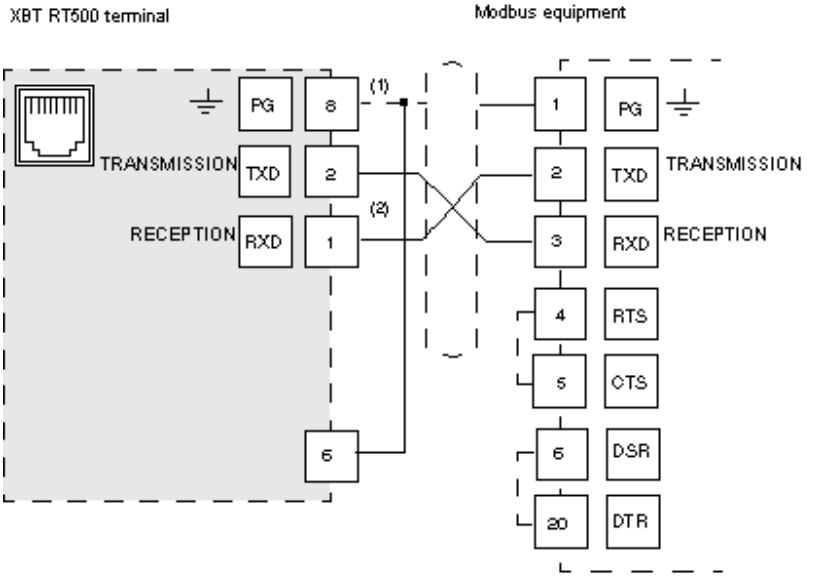
XBT RT500

Pin assignment of the RJ45 connector on XBT RT500 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	+5 V	Power supply: +5 V, I = 200 mA
	8	GND	Common non isolated

The illustration below shows the cabling for RS232C equipment.

RS232C link example

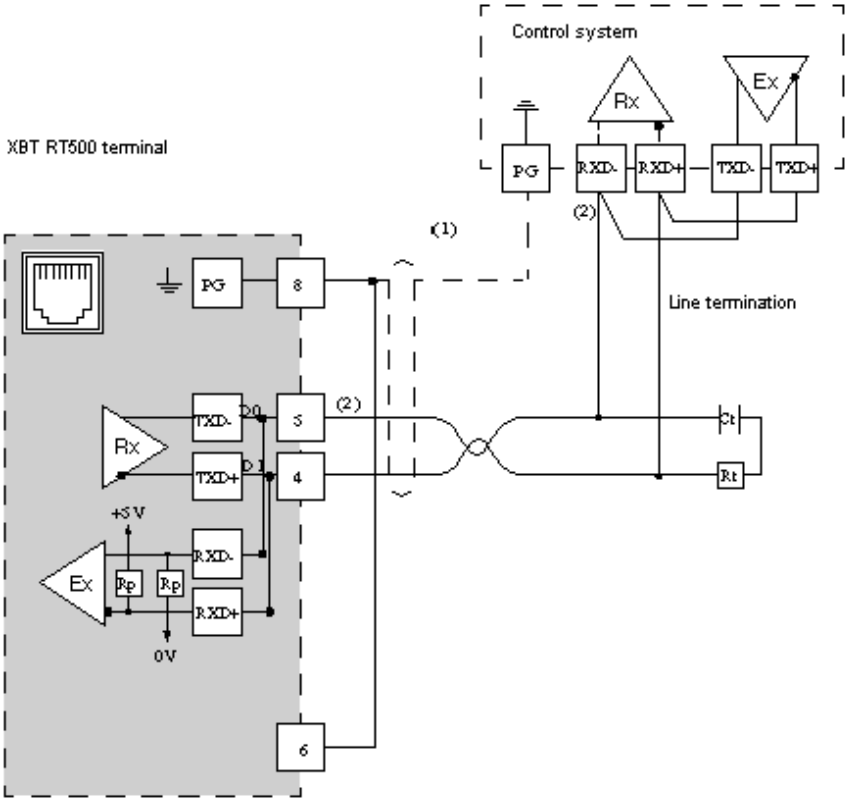


Legend

(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 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

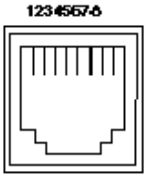


Legend

(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: 600 Ω

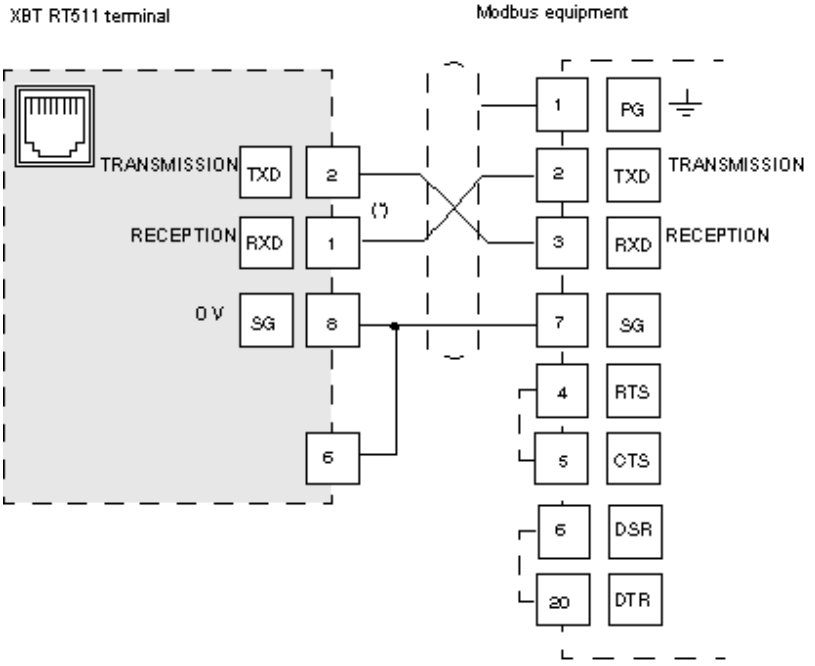
XBT RT511

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

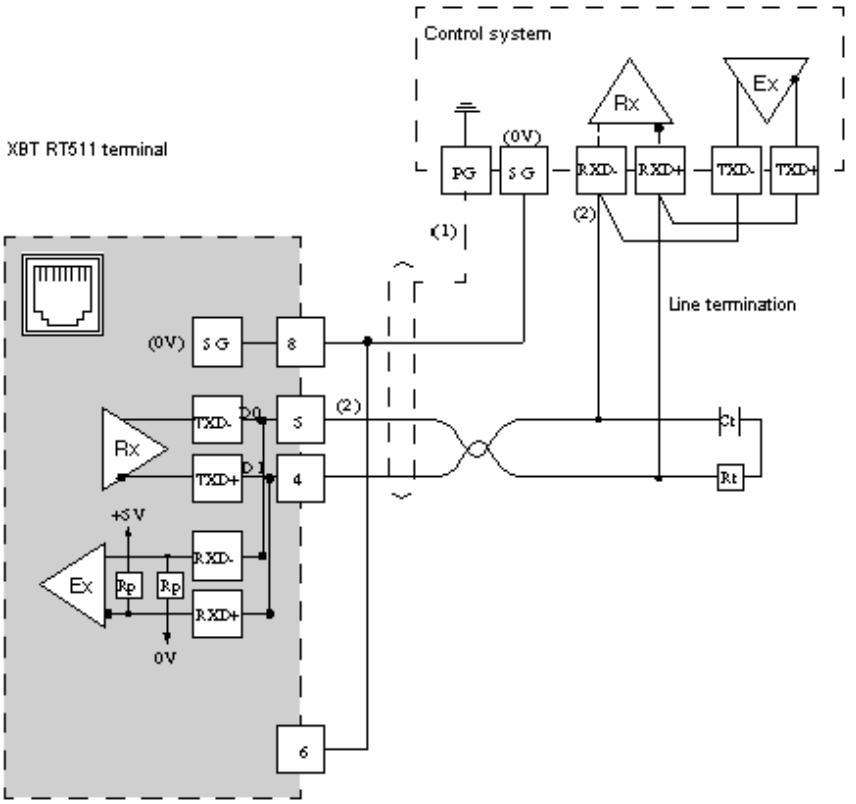


Legend

(*)	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: 600 kΩ

Diagnostics

5

XBT Detected Error Indication

Overview

XBT terminals indicate detected errors in 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.
question marks are displayed	an inadequate cable may be in use between the XBT N200/N400 display units and the PLC: If you use an XBT Z978 cable, then replace it by an XBT Z9780 cable. If you use an XBT Z9780 cable, then add an XBT ZN999 dongle (see *RJ45 Pin Connections, page 41* for further information).
question marks are displayed	verify that the communication parameters set in the **Protocol - Modbus** dialog box are identical for all equipment connected to the Modbus bus i.e. same transmission speed, same use of parity.

If...	Then ...
question marks are displayed	the XBT terminal may have received no response from the slave device.
question marks are displayed	the XBT terminal may have received an exception response from the slave device.

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 alphanumerical 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 communication 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 memory of your PLC is not corrupted ● the correct value is saved on the PLC
message `202: DIALOG TABLE WRITING IMPOSSIBLE` is displayed	the write cycle to the dialog table of the PLC could not be ended. This condition may have the following causes: ● too much load on the communication bus ● EMC disturbances on the communication bus ● If the dialog table is located in a Twido PLC, check that all the words of the dialog table are located between %MW0 and the last word declared in the PLC application.
message `203: DIALOG TABLE READING IMPOSSIBLE` is displayed	the read cycle from the dialog table of the PLC could not be ended. This condition may have the following causes: ● too much load on the communication bus ● EMC disturbances on the communication bus ● If the dialog table is located in a Twido PLC, check that all the words of the dialog table are located between %MW0 and the last word declared in the PLC application.

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 ...
message `241: IMPOSSIBLE TO READ VARIABLE` is displayed	the terminal has attempted to read a variable and could not retrieve its value. This condition may have the following causes: • too much load on the communication bus • EMC disturbances on the communication bus • for connections to Twido PLCs: The variable does not exist in the Twido application.
message `242: IMPOSSIBLE TO WRITE VARIABLE` is displayed	the terminal has attempted to write in a memory area of the equipment and has received a negative acknowledgement or no acknowledgement at all. This condition may have the following causes: • too much load on the communication bus • EMC disturbances on the communication bus • for connections to Twido PLCs: The variable does not exist in the Twido application.
messages `243` to `249` are displayed	correct the value or command you have entered as indicated by the 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 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.
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.

Modbus Master

The sub function codes of the Modbus function code "08 Diagnostics" provide statistical information on requests and responses exchanged between the Modbus master and the slaves. These values are compatibilized in the "counters" of the protocols, displayed by the system panel `#100` on the terminal.

Counter	Meaning
1	Number of responses received by the master without a CRC error.
2	Number of messages received by the master with a CRC error.
3	Number of exception responses received by the master. The exception responses are ● unknown function code ● incorrect address ● incorrect data ● PLC not ready ● acknowledgment ● non-acknowledgment ● write error ● protection zone overlap
4	Number of polling requests sent by the master.
5	Number of requests still unanswered.
6	Number of `PLC not ready` responses received by the master.
7	Number of erroneous characters received (format, parity, overrun, etc).
8	Number of requests correctly executed (the slave event counter read request does not increment this counter).

Appendices



Overview

This chapter contains Modbus master requests and RS232/485 recommendations.

What's in this Appendix?

The appendix contains the following chapters:

Chapter	Chapter Name	Page
A	Modbus Master Requests	55
B	RS232/485 Recommendations	63

Modbus Master Requests

A

Modbus Master Requests

Modbus Functions Supported

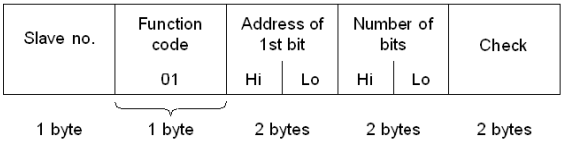
The following sections list the supported Modbus function codes in hexadecimal format.

Supported Modbus functions

Function Code		Sub Code		Name of Function Code	Description
Hex	Dec	Hex	Dec		
01	01	–	–	Read Coils	reading n output or internal bits
02	02	–	–	Read Discrete Inputs	reading n input bits
03	03	–	–	Read Holding Registers	reading n output or internal words
04	04	–	–	Read Input Register	reading n input words
05	05	–	–	Write Single Coil	writing 1 output or internal bit
06	06	–	–	Write Single Register	writing 1 output or internal word
08	08	000B – 000F	00XX	Diagnostic	reading counters no. 1 to no. 8 and counter reset
10	16	–	–	Write Multiple Registers	writing n words
OB	11	–	–	Get Com Event Counter	reading the event counter

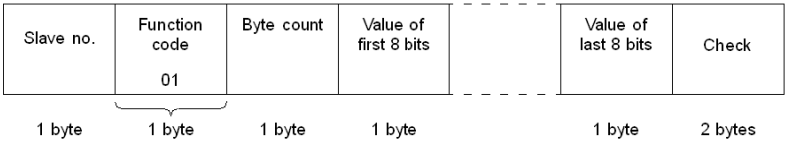
FC 01: Reading n Output or Internal Bits

Read request



Address of the 1st Output or Internal Bit	Corresponds to the address of the 1st output or the 1st internal bit to be read in the slave (output or internal bit 1 = 0, output or internal bit 2 = 1, etc.).
Number of Bits	Number of bits to be read

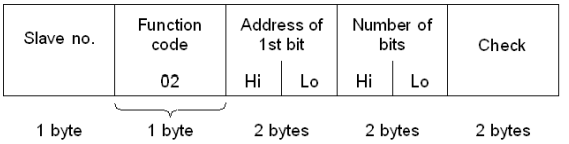
Slave response



Slave number	same as request
Status of Output or Internal Bit According to the Bit Position in the Byte	0000 hex to FFFF hex

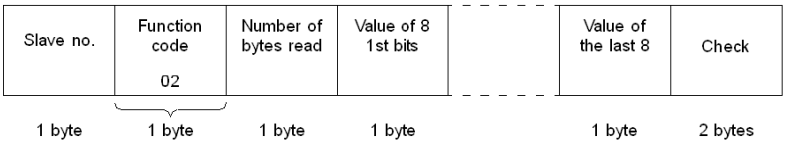
FC 02: Reading n Input Bits

Read request



Address of 1st Input Bit	Corresponds to the address of the 1st input bit to be read in the slave (input 1 = 0, input 2 = 1, etc.).

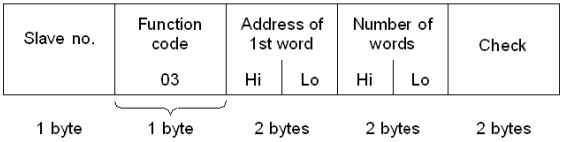
Slave response



Slave number	same as request
Status of Input Bit According to the Bit Position in the Byte	0000 hex to FFFF hex

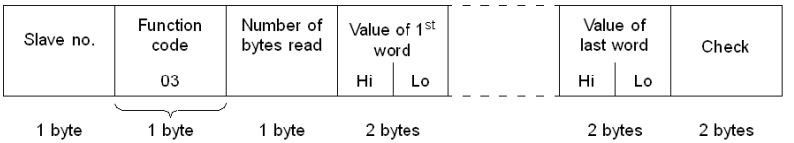
FC 03: Reading n Output or Internal Words

Read request



Address of 1st Word	Corresponds to the address of the 1st word to be read in the slave.
Number of Words	123 words

Slave response

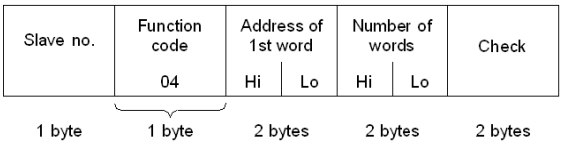


Slave number	same as request
Number of Bytes Read	twice the number of words read, then 1 word is on 2 bytes.
Value of Words Read	0000 hex to FFFF hex

FC 04: Reading n Input Words

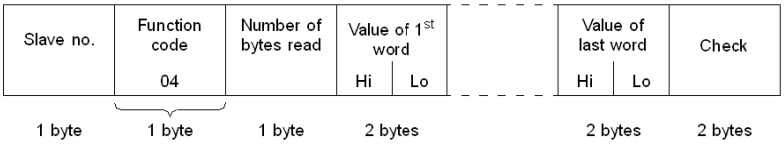
The number of words must be 125.

Read request



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

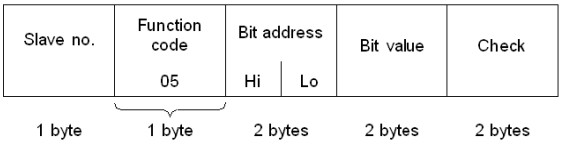
Slave response



Slave number	same as request
Number of Bytes Read	twice the number of words read
Value of Word Read	0000 hex to FFFF hex

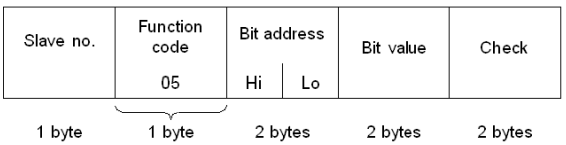
FC 05: Writing 1 Output or Internal Bit

Write request



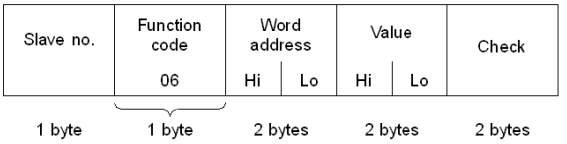
Address of the Output or Internal Bit	Corresponds to the address of the output bit or to the internal bit to be read in the slave (output 1 = 0, output 2 = 1 etc.).
Value of Bit to be Written	• FF00 hex: bit ON • 0000 hex: bit OFF

Slave response

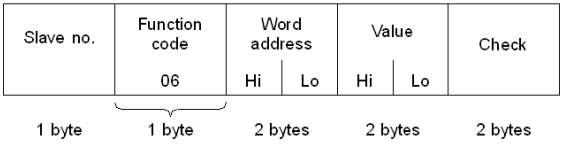


FC 06: Writing 1 Output or Internal Word

Write request

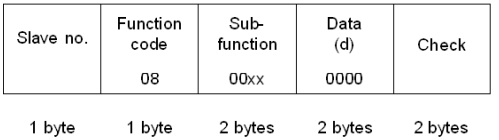


Slave response



FC 08: Reading Counters No. 1 to No. 8 and Counter Resets

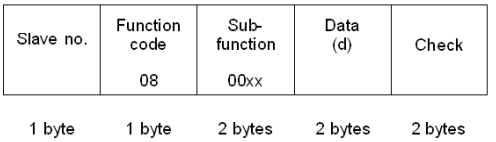
Question



One sub-function code for each function

Reading Counter 1	0x0B
Reading Counter 2	0x0C
...	...
Reading Counter 8	0x0012
Counter Reset	0x000A

Response

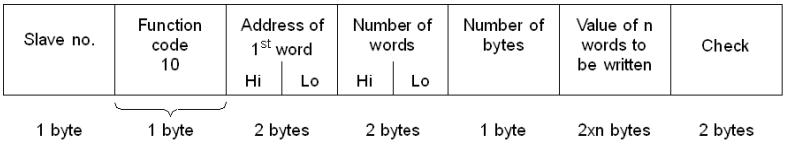


Slave number	same as request

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

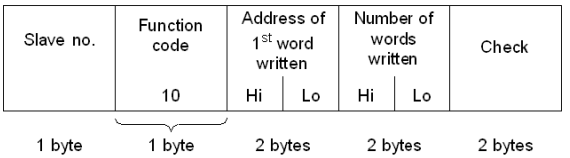
FC 10: Writing n Words

Write 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	0000 hex to FFFF hex

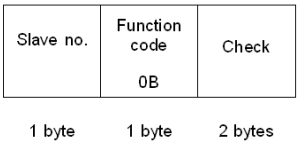
Slave response



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

FC OB: Reading the Event Counter (No. 9)

Read request



Response

Slave no.	Function code 0B	0000	Data (d)	Check
1 byte	1 byte	2 bytes	2 bytes	2 bytes

RS232/485 Recommendations

B

At a Glance

This chapter decribes the RS232 and RS485 recommendations.

What's in this Chapter?

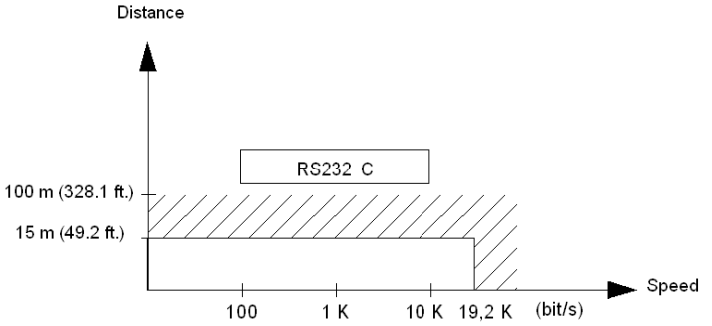
This chapter contains the following topics:

Topic	Page
RS232 Recommendations	64
RS485 Recommendations	65

RS232 Recommendations

Diagrams for RS232C Link

RS232C link



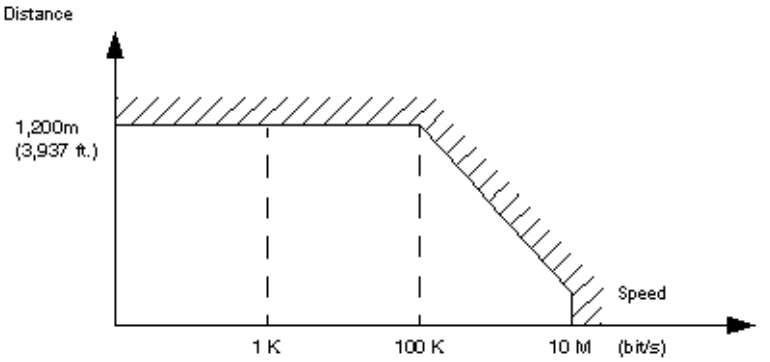
- Maximum length for the link is 15 m (49.2 ft.).
- Wiring = 3 shielded wires with a minimum cross-section of 0.6 mm^2 (AWG22)

NOTE: The maximum length including the RS232 link is 15 m (49.21 ft.), provided that the equipment connected to the XBT terminal is not subject to more stringent restrictions (refer to connected devices instruction sheet) and for XBT RT500 provided that the length of the cable is below 10 m (32.8 ft.) (because power is also supplied by this cable).

RS485 Recommendations

Diagrams for RS485 Link

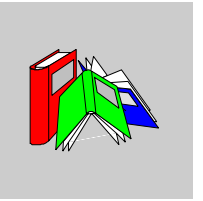
RS485 link



- Maximum length for the link is 1,200 m (3,937 ft.).
- Wiring = 2 shielded twisted wires with a minimum cross-section of 0.6 mm^2 (AWG22) and one 0 V wire

NOTE: The maximum length including the RS485 link is 1,200 m (3,937 ft.), provided that the equipment connected to the XBT terminal is not subject to more stringent restrictions (refer to connected devices instruction sheet) and for XBT RT500 provided that the length of the cable is below 10 m (32.8 ft.) (because power is also supplied by this cable).

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

address
    configuring slave address, *30*
address configuration, *32*

B

biasing, *23*

C

cable length, *21*
cables
    Modbus master protocol, *36*
capacitor, *22*
communication principle
    master / slave, *12*
communication requests
    Modbus master protocol, *55*
configuration
    Modbus protocol, *28*

D

data types
    Modbus master protocol, *33*
diagnostics
    Modbus master protocol, *47*

E

error messages, *47*

F

frame
    incomplete, *18*
frame description
    Modbus master protocol, *19*
frame segment, *19*

G

grounding, *21*, *21*

I

incomplete frame, *18*

L

length of cable, *21*

M

master / slave communication principle, *12*

Modbus master protocol
cables, *36*
communication requests, *55*
data types, *33*
diagnostics, *47*
example of a serial Modbus RTU bus, *20*
frame description, *19*
operating principles, *11*
OSI model, *14*
RTU framing, *17*
RTU transmission mode, *16*
software configuration, *26*

O

objects
Modbus master protocol, *33*
operating principles
Modbus master protocol, *11*
OSI model
Modbus master protocol, *14*

P

pin connections
RJ45, *41*
SUB-D25, *38*
polarization, *23*
protocol configuration, *28*

R

RC termination, *22*
repeater, *21*
resistor, *22*
RJ45
pin connections, *41*
RS232
recommendations, *64*
RS232 cabling, *38*
RS485
recommendations, *65*
RS485 cabling, *39*
RTU framing
Modbus master protocol, *17*
RTU transmission mode
Modbus master protocol, *16*

S

slave address, *30*
slave address configuration, *32*
software configuration
Modbus master protocol, *26*
SUB-D25
pin connections, *38*

T

termination, *22*

V

variable types
Modbus master protocol, *33*