

# FactoryCast HMI Gateway

## TSX ETG 3000 Product Range

### User Manual

03/2012

---

The information provided in this documentation contains general descriptions and/or technical characteristics of the performance of the products contained herein. This documentation is not intended as a substitute for and is not to be used for determining suitability or reliability of these products for specific user applications. It is the duty of any such user or integrator to perform the appropriate and complete risk analysis, evaluation and testing of the products with respect to the relevant specific application or use thereof. Neither Schneider Electric nor any of its affiliates or subsidiaries shall be responsible or liable for misuse of the information contained herein. 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.

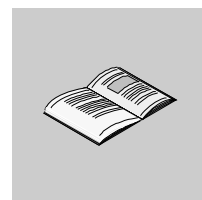
Failure to use Schneider Electric software or approved software with our hardware products may result in injury, harm, or improper operating results.

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

© 2012 Schneider Electric. All rights reserved.

---

# Table of Contents



	<b>Safety Information</b> .....	<b>9</b>
	<b>About the Book</b> .....	<b>11</b>
<b>Chapter 1</b>	<b>TSX ETG 30•• Modules</b> .....	<b>15</b>
	Overview .....	16
	Package Contents .....	18
	Physical Description of the TSX ETG 30•• Modules .....	19
	Summary of the TSX ETG 30•• Functions .....	20
	Compatibility .....	23
<b>Chapter 2</b>	<b>Security</b> .....	<b>25</b>
	Internal Security .....	26
	External Security .....	27
	Variable Access Security, Symbol, Direct Address .....	29
	Internet Communication .....	30
<b>Chapter 3</b>	<b>Hardware Characteristics</b> .....	<b>31</b>
3.1	Description .....	32
	Technical Details .....	32
3.2	Installation of the Module .....	34
	Dimensions and Mounting of the Module .....	35
	Connecting the Power Cord .....	41
	Changing the Battery .....	43
	Inserting a SIM Card .....	47
	CF Card Installation and Removal .....	51
	Wiring to the I/O Connector .....	53
3.3	Connections .....	55
	Module Connectors .....	56
	I/O Board .....	62
	Connection Cables .....	66
3.4	Diagnostics .....	68
	LED Indicators .....	68
3.5	Electrical Characteristics .....	70
	Electrical Characteristics .....	70
3.6	Standards .....	71
	Standards .....	71

3.7	Conditions of Use . . . . .	72
	Conditions of Use . . . . .	72
<b>Chapter 4</b>	<b>Module Setup . . . . .</b>	<b>73</b>
4.1	Hardware Configuration . . . . .	74
	Accessing the Module Configuration . . . . .	75
	Configuration of your Computer for Direct Connection by Serial Link . . . . .	77
	Changing Passwords . . . . .	79
4.2	Topology Examples . . . . .	81
	Remote Access Topologies . . . . .	82
	Network Topologies . . . . .	84
4.3	Modbus Communication . . . . .	92
	Modbus Communication . . . . .	93
	Modbus Server . . . . .	96
	Flat Addressing . . . . .	98
	Configuration Parameters for the Modbus Link . . . . .	99
	Modbus Performance . . . . .	100
4.4	Internal Module Registers . . . . .	101
	Local Registers . . . . .	102
	Date and Time Internal Registers . . . . .	108
	Input/Output Registers . . . . .	110
4.5	TCP/IP Messaging . . . . .	111
	IP Addressing . . . . .	112
	TCP/IP Features . . . . .	114
	Default IP Address . . . . .	115
	Opening a Connection on the Ethernet Network . . . . .	117
	Retrieving the IP Address . . . . .	120
	Configuration Parameters for TCP/IP Services . . . . .	121
	IP Filtering Configuration . . . . .	123
4.6	SOAP / XML Communications . . . . .	125
	Designing a SOAP Client Interface . . . . .	127
	ModBusXMLDA SOAP Services . . . . .	128
	SymbolicXMLDA SOAP Services . . . . .	138
	How to Optimize Massive Symbolic Requests . . . . .	143
	Visual Basic Examples . . . . .	146
	Online SOAP Documentation . . . . .	147
4.7	PSTN, GSM and External Modems . . . . .	148
	Opening a Serial Link Connection via a Modem . . . . .	149
	Modem PPP / Ethernet IP Addressing . . . . .	153
	Configuring the PSTN, GSM, or External Modem . . . . .	157
4.8	GPRS Modem Communications . . . . .	161
	GPRS Communications . . . . .	162
	Configuring the GPRS Modem . . . . .	165
	Opening an Internet Connection via a GPRS Modem . . . . .	169
	PPP Security: IP Control Configuration Page . . . . .	174

4.9	GPRS Communications and NAT Routing Functions . . . . .	176
	NAT Routing Functions . . . . .	177
	Remote Access to TSX ETG 30•• and Devices Using NAT . . . . .	179
	Intersite Communication Using NAT Routing . . . . .	182
	NAT On Port Routing Function . . . . .	186
	NAT On Port Configuration . . . . .	188
	NAT On Port Limitations . . . . .	190
	Modbus UnitID Routing Function . . . . .	191
	Modbus UnitID Routing Configuration. . . . .	194
	Port 502 Forwarding . . . . .	195
	<b>Port 502 Forwarding</b> Configuration. . . . .	197
4.10	GPRS Communications and VPN Functions . . . . .	198
	Increasing Communication Security with a VPN . . . . .	199
	GPRS and VPN Features . . . . .	201
	Remote Access to TSX ETG 302• and Devices . . . . .	202
	Intersite Communications with VPN . . . . .	209
	PPP Security: VPN Control Configuration Page . . . . .	213
4.11	Common Modem Features . . . . .	215
	Phone List Configuration. . . . .	216
	Callback on Ring . . . . .	218
4.12	Ending a Connection. . . . .	219
	Closing a Connection . . . . .	220
	Managing a Broken Connection . . . . .	221
4.13	Remote Connection Availability: Modem Stand-by Function . . . . .	223
	Remote Connection Availability: Modem <b>Stand-by</b> Function . . . . .	224
	<b>Stand-by</b> Configuration Page . . . . .	226
4.14	DHCP (FDR) Service . . . . .	228
	DHCP(FDR) Service Presentation . . . . .	229
	TSX ETG 30•• as DHCP Client . . . . .	230
	TSX ETG 30•• as DHCP(FDR) Client . . . . .	231
4.15	DNS Service . . . . .	233
	DNS Service Presentation. . . . .	234
	DNS Service Configuration . . . . .	236
4.16	NTP Service . . . . .	237
	NTP Service . . . . .	238
	NTP Service Configuration . . . . .	239
4.17	SNMP server. . . . .	240
	SNMP Communication in UDP/IP . . . . .	241
	SNMP Service Configuration. . . . .	244
4.18	Device Explorer. . . . .	246
	Device Explorer Presentation . . . . .	247
	<b>Device Explorer</b> Configuration. . . . .	249
4.19	FTP Service. . . . .	253
	FTP . . . . .	253

4.20	Power Over Ethernet . . . . .	254
	Power Over Ethernet . . . . .	254
4.21	Daisy Chain . . . . .	256
	Daisy Chain . . . . .	256
<b>Chapter 5</b>	<b>Embedded HMI Services . . . . .</b>	<b>257</b>
5.1	Overview of the Embedded HMI Services . . . . .	258
	Device and HMI Services . . . . .	259
	Service Status Overview . . . . .	261
5.2	Device Service . . . . .	266
	Device Service Presentation . . . . .	267
	Naming Variables . . . . .	269
	Device Service Configuration . . . . .	270
5.3	Email and SMS Service . . . . .	276
	Email and SMS Service Presentation . . . . .	277
	Email and SMS Services Configuration . . . . .	279
5.4	Calculation Service . . . . .	285
	Calculation Service Presentation . . . . .	286
	Calculation Service Configuration . . . . .	288
5.5	Database Service . . . . .	293
	Database Service Presentation . . . . .	294
	Database Service Configuration . . . . .	295
5.6	Datalogging Service . . . . .	301
	Datalogging Service Presentation . . . . .	302
	Datalogging Service Configuration . . . . .	306
5.7	Active Pages Service . . . . .	315
	Active Pages Service Presentation . . . . .	316
	Active Pages Service Configuration . . . . .	318
5.8	Recipe Service . . . . .	322
	Recipe Service Presentation . . . . .	323
	Recipe Service Configuration . . . . .	324
5.9	Recipe Database Service . . . . .	330
	Recipe Database Service . . . . .	330
<b>Chapter 6</b>	<b>Embedded Website . . . . .</b>	<b>339</b>
6.1	Overview of the Embedded Website . . . . .	340
	Embedded HTTP Server . . . . .	341
	<b>Home</b> Page for the HTTP Server . . . . .	344
	TSX ETG 30** <b>Version</b> Page . . . . .	345
6.2	Setup Pages . . . . .	346
	Setup Home Page . . . . .	347
	Security Configuration Page . . . . .	348
	Modbus Connection Configuration Page . . . . .	349
	Ethernet Services Configuration Page . . . . .	350
	<b>IP Filtering</b> Page . . . . .	351
	<b>UnitID Routing</b> Page . . . . .	353
	Modem Link Configuration Page . . . . .	354

	<b>Modem/PPP Security</b> Configuration Page . . . . .	355
	<b>Phone List</b> Configuration Page . . . . .	356
	<b>NAT</b> Configuration Page . . . . .	357
	<b>Stand-by</b> Configuration Page . . . . .	358
	DNS Configuration Page . . . . .	359
	NTP Configuration Page . . . . .	360
	SNMP Function Configuration Page . . . . .	361
6.3	Diagnostic Pages . . . . .	362
	<b>Diagnostics</b> Home Page . . . . .	363
	<b>Gateway Statistics</b> Page . . . . .	365
	<b>Ethernet Statistics</b> Page . . . . .	366
	Modbus Statistics Page . . . . .	368
	<b>PPP/Modem Statistics</b> Page . . . . .	370
	<b>PPP/Modem Log File</b> Page . . . . .	374
	<b>PPP/Modem VPN</b> Page . . . . .	376
	<b>PPP/Modem Stand-by</b> Page . . . . .	378
	FDR (Device Replacement) Statistics Page . . . . .	380
	NTP Statistics Page . . . . .	382
	MIB Upload Page . . . . .	384
	<b>Device Explorer</b> Page . . . . .	385
6.4	Control Pages . . . . .	386
	<b>Control</b> Home Page . . . . .	387
	<b>Reboot</b> Page . . . . .	388
	FDR Command Page . . . . .	389
	Date and Time Page . . . . .	390
	<b>Port 502 Forwarding</b> Control Page . . . . .	391
6.5	Monitoring Pages . . . . .	392
	<b>Monitoring</b> Home Page . . . . .	393
	Data Editor Page . . . . .	394
	Data Editor Lite Page . . . . .	395
	Graphic Editor page . . . . .	396
	Graphic Viewer Page . . . . .	397
	Custom Pages . . . . .	398
<b>Chapter 7</b>	<b>Data and Graphic Editors . . . . .</b>	<b>399</b>
7.1	Data Editor . . . . .	400
	Data Editor . . . . .	401
	Creating a Data Template . . . . .	403
	Data Editor Spreadsheet . . . . .	404
	Inserting Symbol Variables in a Data Table . . . . .	406
	Inserting Direct Addresses in a table . . . . .	407
	Modifying Data Values in a Data Template . . . . .	408
	Saving a Data Template . . . . .	410
	Using an Existing Data Template . . . . .	411
7.2	Data Editor Lite . . . . .	412
	Data Editor Lite . . . . .	412

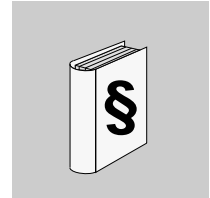
---

7.3	Pocket Data Editor . . . . .	414
	Pocket Data Editor . . . . .	415
	Using Pocket Data Editor . . . . .	417
7.4	Graphic Editor . . . . .	422
	Graphic Editor Presentation . . . . .	423
	Graphic Editor Toolbar . . . . .	425
	User Functions in the Display Window . . . . .	430
	Properties Sheet . . . . .	433
	Security . . . . .	435
	Graphic Editor Applet Parameters . . . . .	436
	Graphic Objects . . . . .	438
	Extended Graphic Objects . . . . .	459
7.5	Graphic Viewer . . . . .	469
	Graphic Viewer . . . . .	469
<b>Chapter 8</b>	<b>Creating Custom Web Pages . . . . .</b>	<b>471</b>
8.1	Working with Custom Pages . . . . .	472
	Uploading Custom Pages via an FTP Client . . . . .	473
	Uploading a Custom Home Page . . . . .	475
8.2	Creating Custom Pages with an HTML Editor . . . . .	477
	Inserting Applets into a Web Page . . . . .	478
	Inserting a LiveBeanApplet . . . . .	479
	Inserting a LiveBeanApplet Using HTML Code . . . . .	481
	Inserting a LiveLabelApplet Using HTML Code . . . . .	484
8.3	Creating Custom Pages with FrontPage . . . . .	489
	Installing a Microsoft FrontPage Extension . . . . .	490
	Inserting a LiveBeanApplet with FrontPage . . . . .	496
	Inserting LiveLabelApplet Using FrontPage . . . . .	504
	Uploading Custom Pages with FrontPage . . . . .	508
8.4	Description of Graphic Objects . . . . .	509
	Graphic Objects Library Lite . . . . .	509
<b>Glossary</b>	. . . . .	<b>531</b>
<b>Index</b>	. . . . .	<b>541</b>



---

## 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 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.

***NOTICE***

***NOTICE*** is used to address practices not related to physical injury.

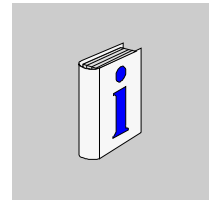
**PLEASE NOTE**

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 its installation, and has received safety training to recognize and avoid the hazards involved.

---

## About the Book



---

### At a Glance

#### Document Scope

This manual presents the TSX ETG 30\*\* gateway module and describes how to install and configure it.

#### Validity Note

This documentation applies to TSX ETG 3000, 3010, and TSX ETG 302• modules.

#### Related Documents

Title of Documentation	Reference Number
Web Designer for TSX ETG 3000 Product Range User Manual	35012721 (English) 35012722 (French)
Grounding and Electromagnetic Compatibility of PLC systems. Basic principles and measures User manual	33002439 (English) 33002440 (French)
Configuration of your computer for TSX ETZ direct connection by serial link	S1B62516

You can download these technical publications and other technical information from our website at [www.schneider-electric.com](http://www.schneider-electric.com).

---

## Product Related Information

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

When controllers are used for applications with technical safety requirements, please follow the relevant instructions.

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

Failure to observe this product related warning can result in injury or equipment damage.

### **DANGER**

#### **HAZARD OF ELECTRIC SHOCK, EXPLOSION OR ARC FLASH**

- Remove the Ethernet cable that provides PoE (only for TSX ETG 3000 modules) to disconnect all power before installing or removing any hardware and cables.
- Always use a properly rated voltage sensing device to confirm power is off.
- Always use suitable insulation equipment when inserting or removing all or part of this equipment.

**Failure to follow these instructions will result in death or serious injury.**

The TSX ETG 30\*\* module enables you to remotely write variables of your system using a Web browser or Web Designer for TSX ETG 30\*\*.

Anyone who has access to the configuration tool and to your embedded server can override your security settings and download new settings to the server.

Unauthorized or incorrect changes to data may change the behavior of your application in ways that may be undesirable or even hazardous.

### **WARNING**

#### **UNINTENDED EQUIPMENT OPERATION**

- Restrict access to the embedded server by configuring passwords.
- Carefully select the symbols and direct addresses you authorize to be modified online.
- Do not authorize online modification of variables of critical nature concerning human and material integrity.

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

---

TSX ETG 302• includes a low power radio transmitter and receiver (GSM modem). When it is ON, it sends and receives radio frequency (RF) signals. GSM communication produces magnetic fields.

## **WARNING**

### **DISTURBANCE OF ELECTRONIC DEVICES**

- Switch off the TSX ETG 302• modules during air travel by disconnecting the power cable. Using GSM devices during air travel is illegal.
- Do not use the TSX ETG 302• modules in hospitals or health care facilities that may be using equipment that could be sensitive to external RF energy (especially near pacemakers and hearing aids).

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

## **CAUTION**

### **DISTURBANCE OF ELECTRONIC DEVICES AND LOSS OF DATA**

- Do not place the module next to magnetic media such as floppy disks, tapes, etc.
- Do not operate the modem close to other electrical equipment such as televisions, telephones, radios and personal computers.
- Do not use the TSX ETG 302• modules while driving. In case of use on cars, check that electronic equipment is shielded against RF signals.

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

The TSX ETG 302• modules, like all wireless devices, are subject to interference that may reduce its performance.

### **User Comments**

We welcome your comments about this document. You can reach us by e-mail at [techcomm@schneider-electric.com](mailto:techcomm@schneider-electric.com).



---

# TSX ETG 30•• Modules

# 1

---

## Scope of this chapter

This chapter contains general information about the TSX ETG 30•• module.

## What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Overview	16
Package Contents	18
Physical Description of the TSX ETG 30•• Modules	19
Summary of the TSX ETG 30•• Functions	20
Compatibility	23

## Overview

### Introduction

The TSX ETG 30•• is a standalone 'all-in-one' TCP-IP/Modbus gateway module that can be used to provide:

- remote control to distributed installations or RTU applications (remote programming, monitoring, diagnostic and maintenance)
- integration of Modbus serial devices into a TCP/IP network
- Web portal function with built-in HMI/SCADA processing

TSX ETG 30•• remote access capabilities allow you to easily transfer application programs to your remote PLCs and devices and perform online modifications using your programming software: Unity Pro, PL7, Concept, Twido Soft or Power Suite. You can also transfer HMI applications to remote Magelis terminals and IPCs using Vijeo Designer software. Remote supervision can be performed using Vijeo Citect SCADA or any other SCADA software.

The main features of the TSX ETG 30•• are:

- gateway messaging between Ethernet network and serial network
- gateway messaging between PPP link and Ethernet/serial network
- remote access with integrated RAS server and Modem (GSM/GPRS or PSTN) or external modem
- support of a second modem (external) depending on the model
- integrated I/O board (6 digital inputs, 2 digital outputs) depending on the model
- flash memory: 32 Mb of internal memory with memory extension up to 1 GB (USB memory stick, CF card)
- FTP server for firmware updates and application up/download
- embedded Web server with predefined setup, diagnostic and monitoring pages and hosting of user-defined Web pages and documents (PDF, doc, etc.)
- built-in HMI/SCADA services with internal processing such as graphical monitoring, datalogging, alarm or report notification via email/SMS, etc.

You can configure TSX ETG 30•• hardware and gateway using either the embedded Web server or Web Designer software for TSX ETG 30••.

The design of Web and HMI applications requires Web Designer software for TSX ETG 30••.



**TSX ETG 30•• Models**

Depending on the model, the TSX ETG 30•• module is delivered either without an internal modem, with an integrated PSTN modem, or with an integrated GSM/GPRS modem. All models include an RS232 serial link for connecting an external modem.

The following table describes the various TSX ETG 30•• models:

<b>Product Name</b>	<b>Product Characteristics</b>
TSX ETG 3000	without an embedded modem
TSX ETG 3010	with a PSTN/RTC modem and embedded Digital Input/Output (DIO)
TSX ETG 3021	with a GSM/GPRS modem (900/1800 MHz band) and embedded DIO
TSX ETG 3022	with a GSM/GPRS modem (850/1900 MHz band) and embedded DIO

## Package Contents

### Content of the Module Package

Each product contains:

- a TSX ETG 30\*\* module
- a quick reference guide (A5 format)
- a DC power supply connector and a TSX ETGACC01 DIO connector
- battery with cable and TSX ETGACC02 connector
- a CD-ROM containing:
  - Web Designer for TSX ETG 30\*\*
  - documentation in French, English
  - dedicated Macro Frontpage for configuring widgets
  - sample application to demonstrate product use and configuration
  - get started guide
  - readme.txt which provides primary information
- safety information for batteries
- an antenna for the GSM/GPRS connection for TSX ETG 302\* modules only

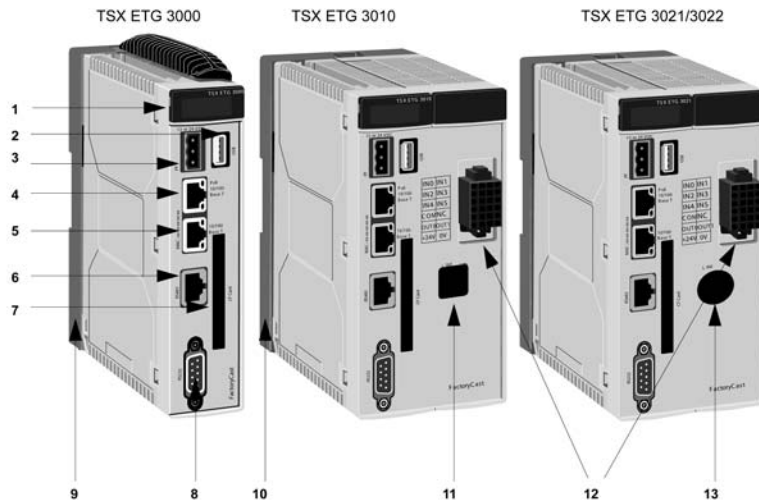
## Physical Description of the TSX ETG 30•• Modules

### Introduction

This topic describes the communication ports, LEDs, and the connectors of the TSX ETG 30••.

### Illustration

The physical elements common to all TSX ETG 30•• models are shown in the illustration of the TSX ETG 3000 module. Physical elements provided for the double-width modules are shown in the illustration of the TSX ETG 3010. The antenna connection for the TSX ETG 302• modules is shown in the third illustration.



- 1 LED indicators
- 2 USB port
- 3 Terminal for 12 or 24 Vdc power supply connection
- 4 RJ45 PoE (Power over Ethernet) connector for Ethernet link. PoE is only available on the TSX ETG 3000
- 5 RJ45 connector for Ethernet link
- 6 RJ45 connector for Modbus RS485 link
- 7 CF card slot
- 8 9-pin SUB D connector for modem link
- 9 Support plate for fixing the module directly to an AM1-DE200/DP200 DIN rail or to a Telemecanique AM1-PA pre-slotted plate
- 10 Support plate for fixing the module directly to an AM1-DE200/DP200 DIN rail or to a Telemecanique AM1-PA pre-slotted plate
- 11 RJ11 connector for internal modem link
- 12 Digital I/O interface
- 13 Connector for the GSM antenna

## Summary of the TSX ETG 30•• Functions

### Presentation

The TSX ETG 30•• module includes the following main functions:

- **gateway and communication functions** such as TCP/IP - Modbus gateway, PPP - Modbus gateway, RAS server, Modbus master, etc
- **embedded Web server functions** such as module setup and diagnostic, application monitoring and control, etc
- **embedded HMI functions** such as graphic monitoring, datalogging, alarm or report notifications via email/SMS, etc

### Gateway and Communication Functions

The module includes the following gateway and communication functions:

- Gateway functions:
  - between Ethernet network and serial link using Modbus/TCP and Modbus/RTU protocols
  - between PPP link (internal or external PSTN/GSM/GPRS modem) and Ethernet or serial devices using Modbus/TCP and Modbus/RTU protocols
- RAS between devices over a modem network (TCP/IP - PPP) and devices over an Ethernet network
- RAS/router function between modem connection (PPP - PSTN/GSM/GPRS) and devices connected to the gateway via the Ethernet network
- NAT routing tables for transparent routing between WAN and Ethernet devices
- VPN support for securing communications over WAN/modem connections
- Modbus master function for device data acquisition by polling up to 31 Modbus slaves

Additional protocols and functions supported:

- HTTP server for TSX ETG 30•• module setup, diagnostic and application monitoring
- FTP server for module firmware updates and application uploads/downloads
- SNMP agent for network management and administration
- NTP client for clock synchronization of TSX ETG 30•• modules, devices and computer systems over Ethernet
- SMTP client for email notification
- DNS client for IP address resolution of SMTP and database servers
- DHCP client for automatic reconfiguration on replacement of TSX ETG 30•• modules (FDR function)
- IP filtering to restrict connection to a list of authorized client IP addresses
- for securing communication over WAN
- Modem callback function: on an incoming telephone call, automatic callback of a telephone number using PSTN/GSM with automatic connection to the GPRS network

## Embedded Web Server Functions

An integrated HTTP server provides Web portal features to your remote installations and RTU applications. The memory of the TSX ETG 30•• consists of 32 MB of internal flash with memory extension such as Compact flash card or USB memory key up to 1 GB.

It includes:

- predefined Web pages for:
  - module setup and control
  - module and application diagnostic
  - application monitoring and control
- user customizable Web pages for:
  - graphic monitoring
  - HTML reporting/dashboard
  - storage of any user defined files such as PDF, docs, schematics, etc. for maintenance purposes
  - real time access to devices and application data from custom Web pages using a library of graphical Java applets

## Embedded HMI Functions

In addition to the gateway and the embedded Web server functions, the TSX ETG 30•• module can perform internal processing.

This processing includes the following HMI services (FactoryCast HMI active services):

- **Device service:** to configure data acquisition from devices (polling of device variables).
- **Calculation service:** to perform calculation scripts for custom data processing (scaling, thresholds, etc.).
- **Email service:** for alarm or report notifications via email or SMS on events, alarm triggers or thresholds exceeded, etc.
- **Datalogging service:** for automatic archiving of application information such as measures, events, alarms, device status, etc. The log files are saved as CSV files into the module memory (Flash memory, CF card, USB memory key).
- **Database connection:** for automatic archiving of application information into an external relational database (SQL Server, Oracle, MySQL).
- **Active pages:** for creating of custom dynamic HTML reports.
- **Recipe service:** for automatically managing recipe data files to initialize a group of variables set at predetermined values when a specified event occurs.
- **Recipe database service:** providing read/write database functions that can be used for recipe management by reading/writing recipe data inside a database (SQL Server, Oracle, MySQL).
- **Data Editor:** to create data tables associated with variables, to be able to determine their value when the table is animated.

- **Pocket Editor:** providing a version of the Data Editor that operates in a Pocket PC environment.
- **Graphic Viewer:** to visualize custom animated graphics for application monitoring.
- **Graphic Editor:** to create custom animated graphics for application monitoring.
- **Custom Web pages:** to create a fully customized Web interface to match customer needs.

## Compatibility

### Presentation

The following information concerns:

- compatibility with other Schneider Electric devices
- PC and software minimum configuration

### Compatibility with Other Devices

The TSX ETG 30•• modules are based on the TSX ETG 10•0 modules. They have new utilities and services and remain compatible with TSX ETG 10•0 modules.

A TSX ETG 30•• is able to replace a TSX ETG 10•0 and it provides the same level of interoperability as a TSX ETG 10•00. A TSX ETG 30•• can communicate with a TSX ETG 10•0 using either the PPP link or the Ethernet link.

**NOTE:** You cannot convert a FactoryCast HMI application in a TSX ETG 30•• application.

### PC and Software Configuration

You need to install the following software on your PC in order to interact with the TSX ETG 30•• modules:

- Web Designer for TSX ETG 30•• to fully configure the module
- *Java Runtime Environment (JRE)* 1.4.2 or above
- a Web browser such as Internet Explorer 6.0 (or above) or Mozilla Firefox 2.0 (or above)





---

# Security

# 2

---

## Scope of this Chapter

Do not use a TSX ETG 30•• to support critical functions. Before configuring your website, you must plan how to secure it. Unlike default website data, data from a custom website is write enabled.

Pay particular attention to the people with access rights to the site and to the data that can be changed. This chapter describes security problems and presents security mechanisms accessible to users of the Web utility.

## What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Internal Security	26
External Security	27
Variable Access Security, Symbol, Direct Address	29
Internet Communication	30

## Internal Security

### Presentation

The TSX ETG 30\*\* provides 2 mechanisms so that only authorized users can view and modify your data:

- password entry
- write restrictions

Anyone who has access to the configuration tool and to your embedded server can override your security settings and download new settings to the server.

Unauthorized or incorrect changes to data may change the behavior of your application in ways that may be undesirable or even hazardous.

### **WARNING**

#### **UNINTENDED OPERATION**

Restrict control of access to the embedded server:

- Change passwords monthly.
- Do not use simple user names and passwords.
- Disable default passwords before commissioning the module.

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

### Password Entry

Although you may add unprotected Web pages to the site, the default Web pages and any other pages you choose to protect can only be viewed by users who supply the correct user name and password.

### Restrictions

When you create a website and you want to protect it, you must place it in the folder called secure. The uploading of the custom website is subject to security conditions linked to an FTP password (*see page 253*).

## External Security

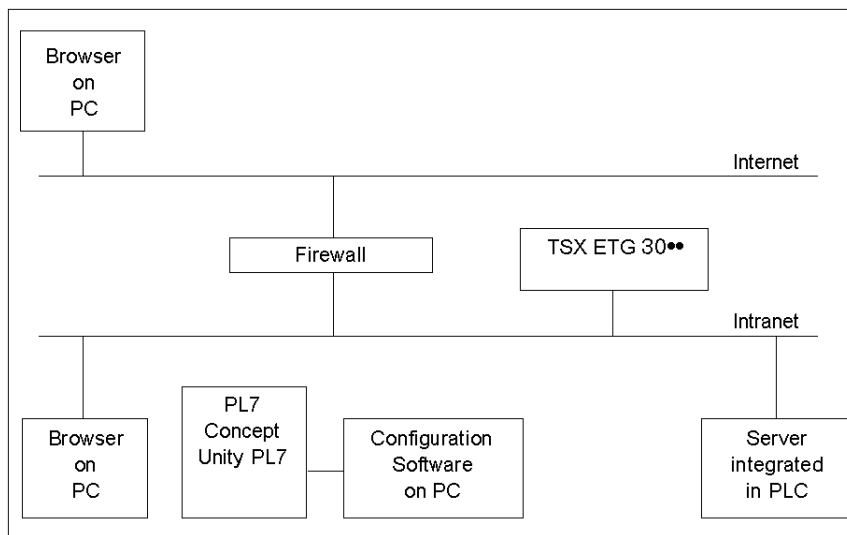
### Presentation

If your network has been configured to enable users to consult your Internet site, your security system is the same as that of an intranet site, only you have an additional security measure: a firewall.

### Architecture of a Firewall

A firewall forms a gateway between the Internet and your embedded server, as illustrated below. You can use a firewall to restrict or forbid access to your web site.

A firewall can be configured to authorize network connections to a limited range of ports, or to authorize traffic to or from certain IP addresses.



### Types of Firewalls

There are two types of firewall:

- network firewalls
- application firewalls

### Network Firewalls

Network firewalls are often installed between the Internet and a single entry point to an intranet or internal protected network.

## Application Firewalls

An application firewall works for an application, for example FTP. It intercepts all traffic sent to this application, and decides whether or not to transmit this traffic to the application. Application firewalls are located on individual host computers.

## Firewall Configuration

HTTP, FTP and Schneider Electric Modbus application protocol (MBAP) are required to access embedded server pages and files. If you want viewers to be able to access your site from the Internet and your embedded server is protected by a firewall, you must configure the firewall to authorize HTTP, FTP and MBAP traffic.

The following table shows the configuration port to authorize the traffic:

Port	Protocol	Use to access to...
21	FTP	Protected embedded server files
Higher than 1024		
80	HTTP	Web pages
502	MBAP	Operational data
500	UDP	VPN communication
50	IP	
51		

### NOTE:

- The default FTP username and password are userdata/userdata.
- The FactoryCast client follows the "Firewall Friendly FTP" standard, RFC 1579. It issues an FTP PASV command to the FactoryCast server before attempting to establish an FTP data connection.
- The online mode of the configuration tool is not operational if the module is protected by a firewall. The ports in this mode are dynamically assigned.

## Variable Access Security, Symbol, Direct Address

### Presentation

Users who enter the data password (*see page 79*) can only modify variables (symbols) and direct addresses which are write-enabled. When you create a Web-enabled database of variables and direct addresses, you can designate each element as read-only or write-enabled.

Unauthorized or incorrect modifications made to symbols and direct addresses may change the behavior of your application in ways that may be undesirable or even hazardous.

### WARNING

#### UNINTENDED EQUIPMENT OPERATION

- Carefully select the symbols and direct addresses you authorize to be modified online.
- Do not authorize online modification of variables of critical nature concerning human and material integrity.

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

## Internet Communication

### Presentation

The GPRS feature enables you to access the module over the Internet. It implies an enhancement of security controls in order to protect access to the module.

<b><i>NOTICE</i></b>
<b>UNINTENDED OPERATION - DISCLOSURE OF DATA</b> Set the confidentiality of your data using IP Control or VPN. <b>Failure to follow these instructions can result in equipment damage.</b>

TSX ETG 302• modules provide 2 security solutions:

- IP control: unknown IP addresses cannot connect to the module.
- VPN (*Virtual Private Network*) (*see page 199*): data traveling between two hosts is encrypted for privacy along with other security features. In this way, private data can safely pass over the Internet.

---

# Hardware Characteristics

# 3

---

## Scope of this Chapter

This chapter describes the hardware characteristics of the TSX ETG 30\*\* module.

## What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
3.1	Description	32
3.2	Installation of the Module	34
3.3	Connections	55
3.4	Diagnostics	68
3.5	Electrical Characteristics	70
3.6	Standards	71
3.7	Conditions of Use	72

## 3.1 Description

---

### Technical Details

#### Computing Power

The module uses a NEC VR4133 266 MHz CPU. The processor speed is 466 MIPS. It has 64 MB of SDRAM and 16 MB of programmable memory accessed by a 32 bits bus.

#### Storage Memory

The TSX ETG 30\*\* includes 32 MB of NAND flash memory accessed by a 8 bits bus. The module also has 512 KB of Static RAM (SRAM) and a RTC dedicated for memory backup.

**NOTE:** The battery enables backup of the SRAM and the RTC.

#### USB Interface

The TSX ETG 30\*\* implements a USB 1.1 host interface with a standard USB connector type A. You can connect a USB memory key up to 1 GB to the module. The USB interface provides power control for the USB memory key.

#### CF Card

Extension of storage memory is provided by a Standard Compact Flash slot Type II in which customer can insert a 1 GB compact flash card (available in the Telemanique catalogue).

#### I/O Board

An integrated digital I/O board is available on double size modules: TSX ETG 3010 and TSX ETG 302\*. The board provides 2 digital outputs (source type) and 6 digital inputs. It uses a specific connector provided with the module (reference: TSXETGACC01).



## PSTN Modem

The PSTN socket modem is part of FactoryCast Gateway TSX ETG 3010.

This socket is plugged in at the factory on the expansion I/O board of the product. It provides an interface to the line with a RJ11 connector available on the front panel of the product.

PSTN modem characteristics:

Characteristic	Value
Maximum transfer rate	33.600 bps (V34)
USA compliance	FCC part 68
FCC registration number	AU7USA-25814-M5-E
European compliance	TBR21:19981999/5/EC R&TTE directive

## GSM/GPRS Modems

The GSM/GPRS socket modem 900/1800 MHz is part of the TSX ETG 3021. This type of modem is specifically for the European market.

The GSM/GPRS socket modem 850/1900 MHz is part of the TSX ETG 3022. This type of modem is specifically for the North American market.

This socket is plugged in at the factory on the expansion I/O board of the product, and a coaxial cable is connected between the modem and the front plate of the product, providing an external connector for an antenna (FME male type).

European Compliance of GSM 900/1800 MHz embedded modem:

GSM	EN 301511 V7.0.1 covered by 3GGP TS 51.010 V5.0.0
EMC	EN 301489-1 V1.2.1 (2002-08) and EN 301489-7 V1.4.1 (2002-08)
Safety	EN 60950:2000
European Council	1999/519/EC recommendation and 1999/5/EC directive R&TTE

American Compliance of GSM 850/1900 MHz embedded modem:

FCC ID	AU792U03G23710
Industry Canada	125a-0009
Rules	FCC part 22.901(d) and FCC part 24E

## 3.2 Installation of the Module

---

### Overview

This section covers the installation of a TSX ETG 30\*\* module.

### What's in this Section?

This section contains the following topics:

Topic	Page
Dimensions and Mounting of the Module	35
Connecting the Power Cord	41
Changing the Battery	43
Inserting a SIM Card	47
CF Card Installation and Removal	51
Wiring to the I/O Connector	53

## Dimensions and Mounting of the Module

### Environmental Considerations

You must pay particular attention to the location where the module will be placed.

**NOTE:** The TSX ETG 30•• module is not certified for operation in explosive environment or where corrosive gases are present.

TSX ETG 302• includes a low power radio transmitter and receiver (GSM modem). When it is ON, it sends and receives radio frequency (RF) signals. GSM communication produces magnetic fields.

### WARNING

#### **DISTURBANCE OF ELECTRONIC DEVICES**

- Switch off the TSX ETG 302• module during air travel by disconnecting the power cable. Using GSM devices during air travel is illegal.
- Do not use the TSX ETG 302• module in hospitals or health care facilities that may be using equipment that could be sensitive to external RF energy (especially near pacemakers and hearing aids).

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

### WARNING

#### **EXPOSURE TO RADIO FREQUENCY ENERGY**

Keep to a minimum distance of 40 cm (15.74 in) from the antenna of the TSX ETG 302• module when it is transmitting.

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

### CAUTION

#### **DISTURBANCE OF ELECTRONIC DEVICES AND LOSS OF DATA**

- Do not place the module next to magnetic media such as floppy disks, tapes, etc.
- Do not operate the modem close to other electrical equipment such as televisions, telephones, radios and personal computers.
- Do not use the TSX ETG 302• while driving. In case of use on cars, check that electronic equipment is shielded against RF signal.

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

The TSX ETG 302• module, like all wireless devices, is subject to interference that may reduce its performance.

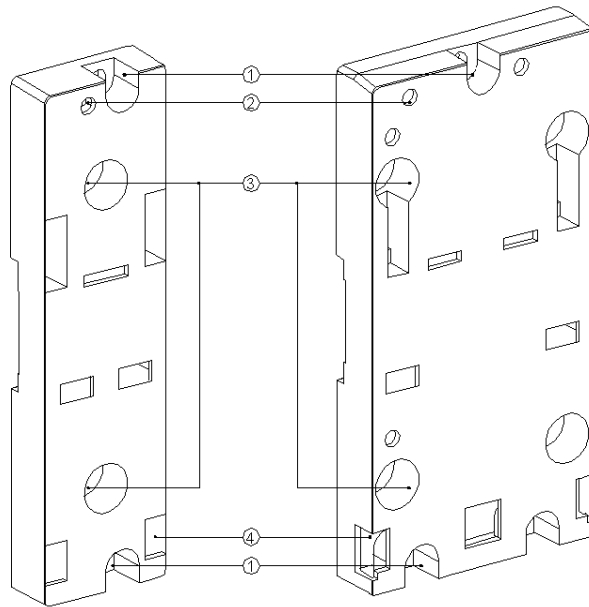
**NOTE:** The TSX ETG 30•• module must be located away from devices that might cause overheating or arc flashes.

### Pre-Slotted Plates

Each TSX ETG 30•• module is fixed to a support plate that can be mounted on either an AM1-DE200 or AM1-DP200 DIN rail.

TSX ETG 30•• can also be mounted either on a Telequick AM1-PA or on a Telequick AM2-PA pre-slotted plate depending on the size of the model:

- AM1-PA for TSX ETG 3000 (single size)
- AM2-PA for TSX ETG 3010, TSX ETG 3021 and TSX ETG 3022 (double size)



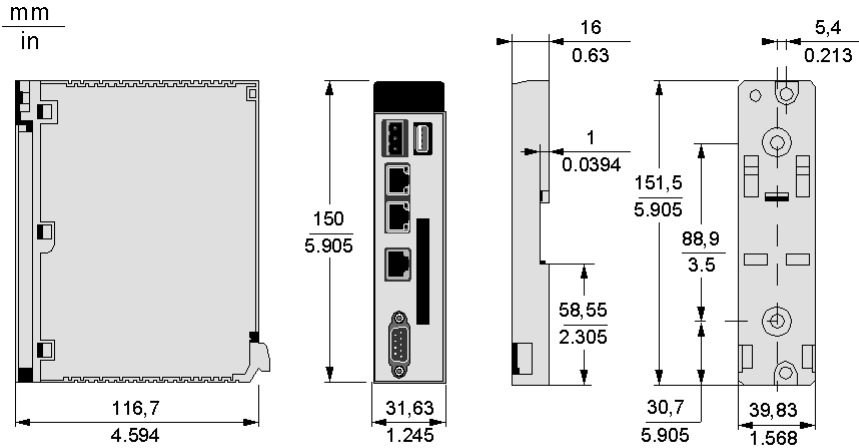
### Description of the Plate

Label	Meaning
1	Two 5.5 mm (7/32 in) holes for securing the plate to a panel or to an AM1-PA pre-slotted plate, with fixing centers of 140 mm (5.51 in, Micro fixing centers). Tightening torque: 1 to 1.2 N-m
2	M4 fixing hole for securing the TSX ETG 30•• module.

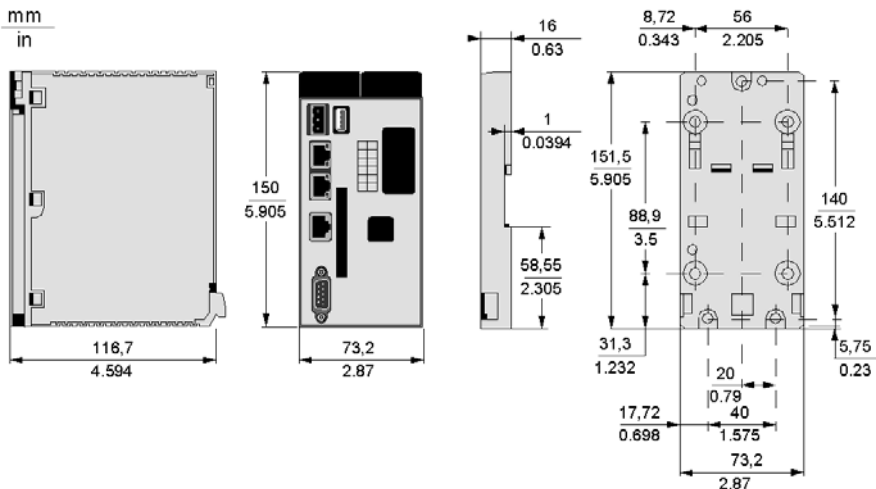
Label	Meaning
3	Two 6.5 mm holes for securing the plate to a panel or to an AM1-PA pre-slotted plate, with fixing centers of 88.9 mm (3.5 in, TSX Premium fixing centers).
4	Windows for marking the pins located on the base and rear of the module.

### Dimensions

The following figure shows the dimensions of a single size module:

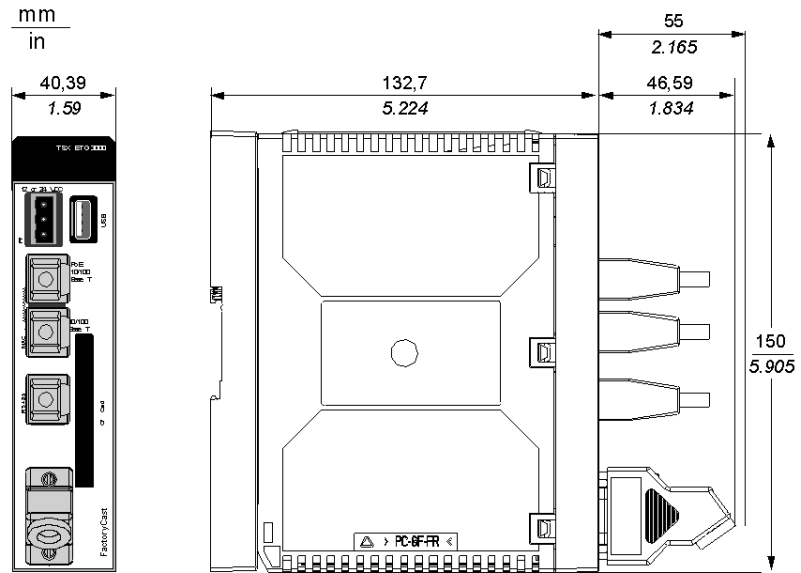


The following figure shows the dimensions of a double size module:



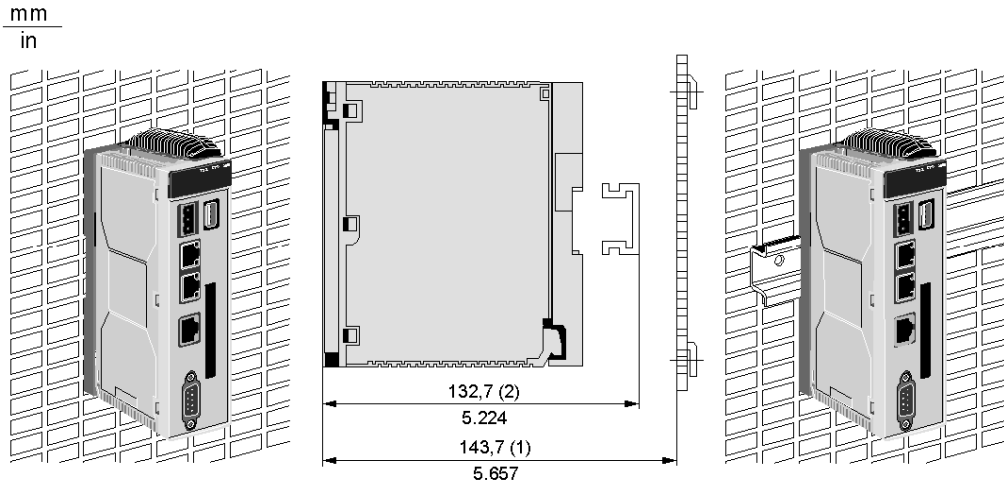
### Dimensions of Ready-Assembled Module

The following figure shows the dimensions of the ready-assembled module with its cables connected to the front panel:



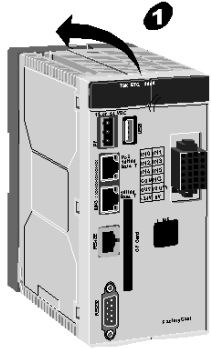
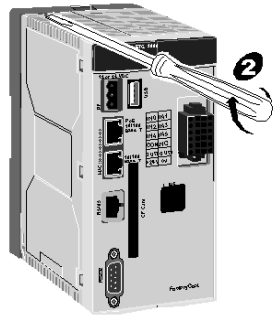
### Mounting on a Rail or Plate

The following figure shows the module mounted on an AM1-DE200 or AM1-DP200 rail or on an AM1-PA plate:



### Installing the Module

To install the module from its support plate, follow the steps below:


Step	Action	Illustration
1	Fix the support plate to the rail or the grid.	-
2	Clip the module to the bottom of the support plate and push the module upwards.	
3	Fasten the screw located at the top of the module to fix it to its support.	



## Connecting the Power Cord

### Introduction

Follow these instructions when supplying power to the TSX ETG 30\*\* module.


**DANGER**

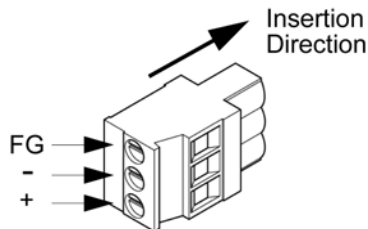
**HAZARD OF ELECTRIC SHOCK, EXPLOSION OR ARC FLASH**

- Remove the Ethernet cable that provides PoE (only for TSX ETG 3000 modules) to disconnect all power before installing or removing any hardware and cables.
- Always use a properly rated voltage sensing device to confirm power is off.
- Always use suitable insulation equipment when inserting or removing all or part of this equipment.

**Failure to follow these instructions will result in death or serious injury.**

### Power Plug Illustration

The following illustration shows the power plugs for the TSX ETG 30\*\* module:



Connection	Wire
+	24 Vdc
-	0 Vdc
<b>FG</b>	Grounded terminal connected to the TSX ETG 30** chassis.

### Grounding the Module

Functional Ground (**FG**) is available on the power connector in order to discharge high frequency disturbance. The module is grounded using the metal plate at the rear of the module. When the module is in place, this metal plate is in contact with the metal of the support plate. Shielded connectors (USB, RJ45 Ethernet, RJ45 Modbus, SubD9 and DIO) are also connected to the **FG**. However to avoid an electric shock caused by indirect contact, the **FG** must be linked to the protective ground.

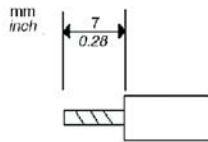
For more information on grounding the module, refer to the *Grounding and Electromagnetic Compatibility of PLC Systems Basic Principles and Measures User Manual*.

### Power Cord Preparation

**NOTE:**

- Wherever possible, use wires that are 0.2 to 2.5 mm (24 - 12 AWG) for the power cord, and twist the wire ends before attaching the terminals.
- The maximum length of the power cord is 10 m (393 in).
- Not properly twisting the wire ends may cause short loops. To avoid this, use D25CE/AZ5CE cable ends.
- The conductor type is solid or stranded wire.

Stripping length:



### Connecting the Power Cord

Step	Action
1	Remove the Ethernet cable that provides PoE (only for TSX ETG 3000 modules).
2	Remove the vinyl cover of each of the power cord's wires.
3	Twist the wire ends.
4	Connect the wires to the power plug using a flat-blade screwdriver (0.6 X 3.5).
5	Torque the mounting screws: 0.5...0.6 Nm 4.4...5.2 lb-in).
6	Replace the power plug to the power connector.

**NOTE:**

- Do not solder the wire directly to the power receptacle pin.
- The power supply cord should meet the specification shown above. Be sure to twist the power cords together, up to the power plug, for EMC reason.

## Changing the Battery

### Introduction

The TSX ETG 30\*\* uses one 3 Vdc, 1200 mAh lithium battery. Its reference is TSXETGACC02.

If power is lost, the TSX ETG 30\*\* uses battery power to back up data to the internal RAM. It also uses the battery to keep track of the Real Time Clock of the module. The battery is delivered with the TSX ETG 30\*\* and you must install it.

### Battery Power Status

The BAT LED indicator located on the front panel of the TSX ETG 30\*\* indicates the state of the battery. The BAT LED states are described in the following table:

LED State	Description
Extinguished	The battery is functioning normally.
Steady red	Indicates that either: <ul style="list-style-type: none"> <li>● the power of the battery is low (you must replace the battery as soon as possible), or</li> <li>● there is no battery installed in the battery compartment.</li> </ul>

### Installing and Replacing a Battery

The battery compartment is located on the back side of the TSX ETG 30\*\* case. To install or replace the battery, follow these steps:

#### **DANGER**

##### **HAZARD OF ELECTRIC SHOCK, EXPLOSION OR ARC FLASH**

- Remove the Ethernet cable that provides PoE (only for TSX ETG 3000 modules) to disconnect all power before installing or removing any hardware and cables.
- Always use a properly rated voltage sensing device to confirm power is off.
- Always use suitable insulation equipment when inserting or removing all or part of this equipment.

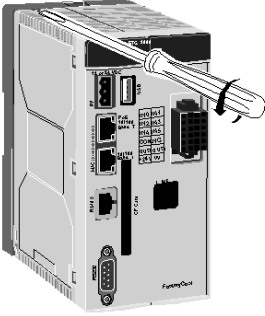
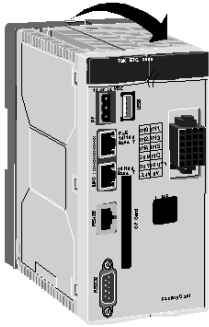
**Failure to follow these instructions will result in death or serious injury.**

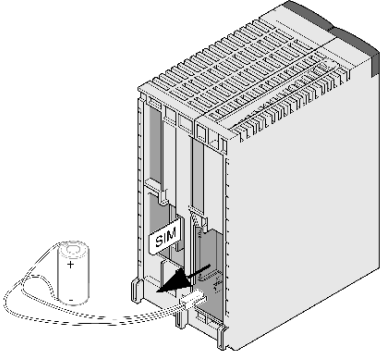
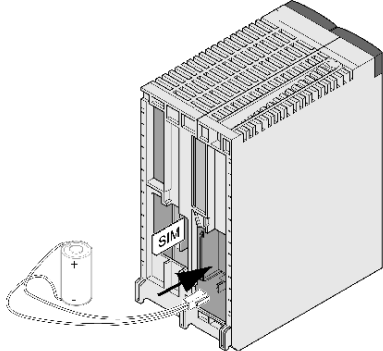

## ⚠ WARNING

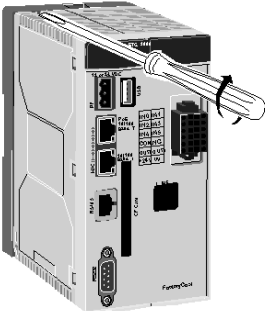
### EXPLOSION AND FIRE HAZARD

Replace the battery with part number TSXETGACC02 only.

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

Step	Action	Illustration
1	Remove the Ethernet cable that provides PoE (only for TSX ETG 3000 modules).	–
2	Remove the power cord from the power supply.	–
3	Remove the power plug from the TSX ETG 3000.	–
4	Unfasten the screw located at the top of the module to loosen it from its support.	
5	Pull the module downwards and unclip it from the bottom of the support.	

Step	Action	Illustration
6	Remove the used battery, if applicable, from the compartment. Unclip the connector of the battery.	
7	Plug the connector of the new battery. Insert the new battery vertically in its compartment.	
8	Clip the module to the bottom of the support and push the module upwards.	

Step	Action	Illustration
9	Fasten the screw located at the top of the module to fix it to its support.	 <p>The illustration shows a TSX ETG 30 module installed in a rack. A screwdriver is shown tightening a screw on the top edge of the module to secure it to the rack's support structure. The front panel of the module features various ports and labels, including 'TSX ETG 30', 'CPU', 'DI', 'DO', 'AI', 'AO', 'RS-485', 'RS-232', 'M12', 'M17', 'M23', 'M25', 'M27', 'M29', 'M31', 'M33', 'M35', 'M37', 'M39', 'M41', 'M43', 'M45', 'M47', 'M49', 'M51', 'M53', 'M55', 'M57', 'M59', 'M61', 'M63', 'M65', 'M67', 'M69', 'M71', 'M73', 'M75', 'M77', 'M79', 'M81', 'M83', 'M85', 'M87', 'M89', 'M91', 'M93', 'M95', 'M97', 'M99', 'M101', 'M103', 'M105', 'M107', 'M109', 'M111', 'M113', 'M115', 'M117', 'M119', 'M121', 'M123', 'M125', 'M127', 'M129', 'M131', 'M133', 'M135', 'M137', 'M139', 'M141', 'M143', 'M145', 'M147', 'M149', 'M151', 'M153', 'M155', 'M157', 'M159', 'M161', 'M163', 'M165', 'M167', 'M169', 'M171', 'M173', 'M175', 'M177', 'M179', 'M181', 'M183', 'M185', 'M187', 'M189', 'M191', 'M193', 'M195', 'M197', 'M199', 'M201', 'M203', 'M205', 'M207', 'M209', 'M211', 'M213', 'M215', 'M217', 'M219', 'M221', 'M223', 'M225', 'M227', 'M229', 'M231', 'M233', 'M235', 'M237', 'M239', 'M241', 'M243', 'M245', 'M247', 'M249', 'M251', 'M253', 'M255', 'M257', 'M259', 'M261', 'M263', 'M265', 'M267', 'M269', 'M271', 'M273', 'M275', 'M277', 'M279', 'M281', 'M283', 'M285', 'M287', 'M289', 'M291', 'M293', 'M295', 'M297', 'M299', 'M301', 'M303', 'M305', 'M307', 'M309', 'M311', 'M313', 'M315', 'M317', 'M319', 'M321', 'M323', 'M325', 'M327', 'M329', 'M331', 'M333', 'M335', 'M337', 'M339', 'M341', 'M343', 'M345', 'M347', 'M349', 'M351', 'M353', 'M355', 'M357', 'M359', 'M361', 'M363', 'M365', 'M367', 'M369', 'M371', 'M373', 'M375', 'M377', 'M379', 'M381', 'M383', 'M385', 'M387', 'M389', 'M391', 'M393', 'M395', 'M397', 'M399', 'M401', 'M403', 'M405', 'M407', 'M409', 'M411', 'M413', 'M415', 'M417', 'M419', 'M421', 'M423', 'M425', 'M427', 'M429', 'M431', 'M433', 'M435', 'M437', 'M439', 'M441', 'M443', 'M445', 'M447', 'M449', 'M451', 'M453', 'M455', 'M457', 'M459', 'M461', 'M463', 'M465', 'M467', 'M469', 'M471', 'M473', 'M475', 'M477', 'M479', 'M481', 'M483', 'M485', 'M487', 'M489', 'M491', 'M493', 'M495', 'M497', 'M499', 'M501', 'M503', 'M505', 'M507', 'M509', 'M511', 'M513', 'M515', 'M517', 'M519', 'M521', 'M523', 'M525', 'M527', 'M529', 'M531', 'M533', 'M535', 'M537', 'M539', 'M541', 'M543', 'M545', 'M547', 'M549', 'M551', 'M553', 'M555', 'M557', 'M559', 'M561', 'M563', 'M565', 'M567', 'M569', 'M571', 'M573', 'M575', 'M577', 'M579', 'M581', 'M583', 'M585', 'M587', 'M589', 'M591', 'M593', 'M595', 'M597', 'M599', 'M601', 'M603', 'M605', 'M607', 'M609', 'M611', 'M613', 'M615', 'M617', 'M619', 'M621', 'M623', 'M625', 'M627', 'M629', 'M631', 'M633', 'M635', 'M637', 'M639', 'M641', 'M643', 'M645', 'M647', 'M649', 'M651', 'M653', 'M655', 'M657', 'M659', 'M661', 'M663', 'M665', 'M667', 'M669', 'M671', 'M673', 'M675', 'M677', 'M679', 'M681', 'M683', 'M685', 'M687', 'M689', 'M691', 'M693', 'M695', 'M697', 'M699', 'M701', 'M703', 'M705', 'M707', 'M709', 'M711', 'M713', 'M715', 'M717', 'M719', 'M721', 'M723', 'M725', 'M727', 'M729', 'M731', 'M733', 'M735', 'M737', 'M739', 'M741', 'M743', 'M745', 'M747', 'M749', 'M751', 'M753', 'M755', 'M757', 'M759', 'M761', 'M763', 'M765', 'M767', 'M769', 'M771', 'M773', 'M775', 'M777', 'M779', 'M781', 'M783', 'M785', 'M787', 'M789', 'M791', 'M793', 'M795', 'M797', 'M799', 'M801', 'M803', 'M805', 'M807', 'M809', 'M811', 'M813', 'M815', 'M817', 'M819', 'M821', 'M823', 'M825', 'M827', 'M829', 'M831', 'M833', 'M835', 'M837', 'M839', 'M841', 'M843', 'M845', 'M847', 'M849', 'M851', 'M853', 'M855', 'M857', 'M859', 'M861', 'M863', 'M865', 'M867', 'M869', 'M871', 'M873', 'M875', 'M877', 'M879', 'M881', 'M883', 'M885', 'M887', 'M889', 'M891', 'M893', 'M895', 'M897', 'M899', 'M901', 'M903', 'M905', 'M907', 'M909', 'M911', 'M913', 'M915', 'M917', 'M919', 'M921', 'M923', 'M925', 'M927', 'M929', 'M931', 'M933', 'M935', 'M937', 'M939', 'M941', 'M943', 'M945', 'M947', 'M949', 'M951', 'M953', 'M955', 'M957', 'M959', 'M961', 'M963', 'M965', 'M967', 'M969', 'M971', 'M973', 'M975', 'M977', 'M979', 'M981', 'M983', 'M985', 'M987', 'M989', 'M991', 'M993', 'M995', 'M997', 'M999'. The 'FactoryDirect' logo is visible at the bottom right of the front panel.</p>
10	Connect the power plug to the module.	-
11	Connect the power cord to the power supply. Power up your TSX ETG 30** module immediately to preserve battery life.	-

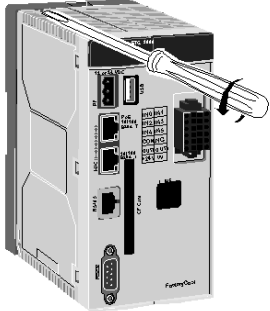

## Inserting a SIM Card

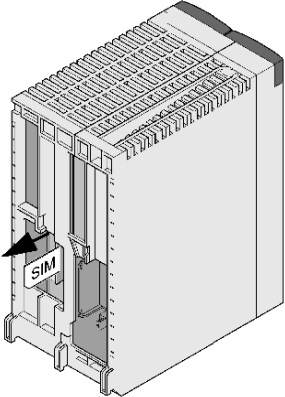
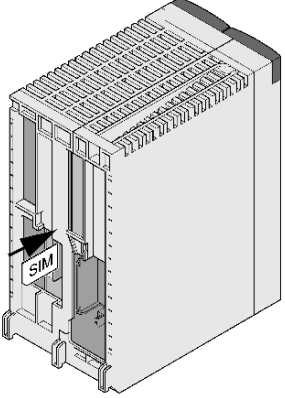

### Introduction

The TSX ETG 302• modules contain a GSM/GPRS modem. The use of a SIM card is mandatory in the GSM/GPRS connection. The following part shows you how to insert a SIM card into a TSX ETG 302• and how to find the IMEI number of the module.

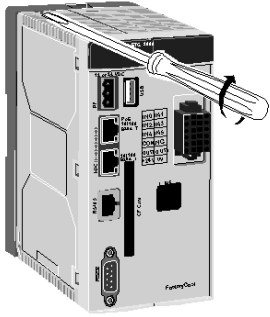
### Inserting a SIM Card

To remove the module from its support plate, follow the steps below:

Step	Action	Illustration
1	Remove the power cord from the power supply.	—
2	Remove the power plug from the TSX ETG 302•.	—
3	Unfasten the screw located at the top of the module to loosen it from its support.	
4	Pull the module downwards and unclip it from the bottom of the support.	

Step	Action	Illustration
5	Remove the old SIM card, if applicable, from the compartment.	
6	Insert the new SIM card in the compartment.	
7	Push the module upwards and clip it to the bottom of the support.	

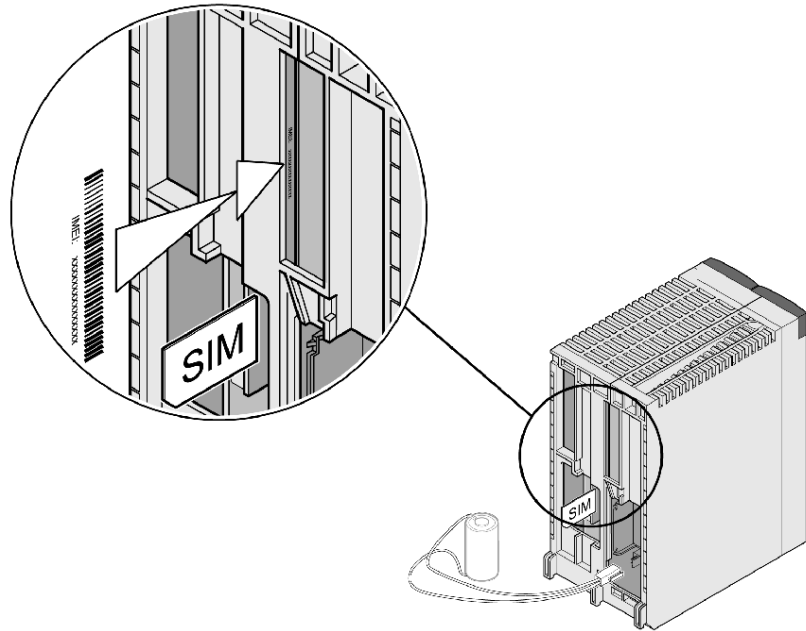


Step	Action	Illustration
8	Fasten the screw located at the top of the module to fix it to its support.	 A photograph of a grey industrial module, identified as 'Furphy-Data' at the bottom. The front panel features several ports: a USB port, two RJ45 ports, a power jack, and a terminal block. A screw is being inserted into a hole at the top of the module, secured by a metal bracket.
9	Connect the power plug to the module.	—
10	Connect the power cord to the power supply. Power up the TSX ETG 302• module immediately to preserve battery life.	—

## IMEI Number

The GSM/GPRS network uses the IMEI number to identify valid devices and therefore can be used to stop a stolen phone from accessing the network.

This number is printed on the GSM/GPRS board of the module. To find it, follow the preceding procedure. The number printed on the card can be read at steps 5 and 6.



## CF Card Installation and Removal

### Precautions for Using a CF Card

Use only CF cards manufactured by Schneider Electric. The performance of the TSX ETG 30\*\* cannot be guaranteed when using CF cards from another manufacturer.

#### NOTE:

- TSX ETG 30\*\* modules only support CF cards that have been formatted in FAT or FAT32 format.
- Schneider Electric recommends you:
  - do not bend the CF card
  - do not drop or strike the CF card against another object
  - do not touch the CF card connectors
  - do not disassemble or modify the CF card
  - use the push button to remove the CF card
  - keep the CF card dry

The TSX ETG 30\*\* module views the CF card as a hard disk.

If the CF card is inserted incorrectly, the CF card, its internal data, and the CF card unit may be damaged.

### CAUTION

#### LOSS OF DATA OR CF CARD MALFUNCTION

- Do not remove the CF card when the EXT LED is flashing.
- Prior to inserting the CF card, check that the rear and the front of the CF card are correctly oriented, and that the CF card connector position is correct.

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

### Inserting the CF Card

Step	Action
1	Disable or prevent the TSX ETG 30** from reading or writing to the CF card.
2	Insert the CF card firmly into the CF card slot, and check that the eject button pops out.

### Removing the CF Card

Step	Action
1	Disable or prevent the TSX ETG 30•• from accessing to the CF card.
2	Press the eject button in fully to remove the CF card from the CF card slot.

### Data Writing Restrictions

The CF card has a limitation of approximately 100,000 data writings. Therefore, be sure to back up all CF card data regularly to another storage media.

### Backup of the CF Card Data

Step	Action
1	Install the CF card into a CF card adaptor and insert the adaptor into a PC card slot on the personal computer.
2	Save the data from the CF card onto your personal computer.


## Wiring to the I/O Connector

### Required Items

- Screwdriver:
  - Recommended type: 1891348-1 <Tyco Electronics AMP.>
  - If another manufacturer is used, be sure the part has the following dimensions: point depth 1.5 mm (0.06 in) and point high 2.4 mm (0.09 in).
  - The point shape should be DIN5264A, and meet Security Standard DN EN60900.
  - The screwdriver tip must be flat in order to access the narrow hole of the connector.
- Wire:
  - The wire should be AWG24 to AWG18 thick, and twisted.
  - Applicable wire sizes are UL1015 and UL1007.
  - Be sure to strip at least 7.0 mm (0.28 in) of cover from the wire.

**Procedure**

Use the following procedure to connect the wires to the connectors:

 **DANGER**

**HAZARD OF ELECTRIC SHOCK, EXPLOSION OR ARC FLASH**

- Remove the Ethernet cable that provides PoE (only for TSX ETG 3000 modules) to disconnect all power before installing or removing any hardware and cables.
- Always use a properly rated voltage sensing device to confirm power is off.
- Always use suitable insulation equipment when inserting or removing all or part of this equipment.

**Failure to follow these instructions will result in death or serious injury.**

Step	Action
1	Disconnect the I/O connector from the module or keep it out for a first use.
2	Insert the screwdriver into the square-shaped hole. <b>Result:</b> it opens the wire's round-shaped hole.
3	Hold the screwdriver and insert the wire into the wire's round-shaped hole.
4	Take out the screwdriver from the square-shaped hole. <b>Result:</b> the round-shaped hole is closed and the wire is held securely in place.

**NOTE:** To remove the wire, re-insert the screwdriver into the square-shaped hole and when the wire spring clamp releases, pull the wire out.

---

## 3.3 Connections

---

### Overview

This section describes the electrical connections for the TSX ETG 30\*\* module.

### What's in this Section?

This section contains the following topics:

Topic	Page
Module Connectors	56
I/O Board	62
Connection Cables	66

## Module Connectors

### Ground Connection

Functional Ground (FG) is available on power connector in order to discharge high frequency disturbance. The module is grounded using the metal plate at the rear of the module. When the module is in place, these metal plates are in contact with the metal of the support plate. Shielded connectors (USB, RJ45 Ethernet, RJ45 Modbus, SubD9 and DIO) are also connected to the FG.

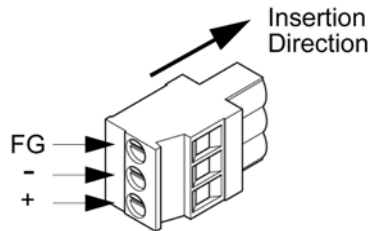
However to avoid an electric shock caused by indirect contact, FG must be linked to the protective ground.

For more information on grounding the module, refer to the *Grounding and Electromagnetic Compatibility of PLC Systems Basic Principles and Measures User Manual*.

### Power Supply Terminals

The power supply terminal is a 3-pin terminal located on the front panel of the module. Use the power plug delivered with the module to bring 12 Vdc or 24 Vdc power to the power supply terminal. The power plug accepts cables of up to 14 AWG (2.5 mm<sup>2</sup>).

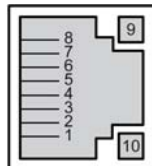
The following illustration shows the power plug for the TSX ETG 30\*\* module:



For replacement of the power plug, order part number TSXETGACC01.

### RJ45 Ethernet Connector

The following illustration shows the RJ45 shielded connector for the Ethernet link:





The following table shows the pinouts of the RJ45 shielded connector (without PoE):

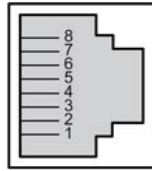
Pin	Signal
1	Tx(+)
2	Tx(-)
3	Rx(+)
4	–
5	–
6	Rx-
7	–
8	–
9	LED indicator: status of the link
10	LED indicator: communication activity

The following table shows the pinouts of the RJ45 shielded connector with PoE (only on the TSX ETG 3000 modules). Using mode A or mode B is transparent for the user:

Pin	Ethernet Signal	PoE mode A	PoE mode B
1	Tx(+)	Positive $V_{Port}$ , Negative $V_{Port}$	–
2	Tx(-)	Positive $V_{Port}$ , Negative $V_{Port}$	–
3	Rx(+)	Negative $V_{Port}$ , Positive $V_{Port}$	–
4	–	–	Positive $V_{Port}$ , Negative $V_{Port}$
5	–	–	Positive $V_{Port}$ , Negative $V_{Port}$
6	Rx-	Negative $V_{Port}$ , Positive $V_{Port}$	–
7	–	–	Negative $V_{Port}$ , Positive $V_{Port}$
8	–	–	Negative $V_{Port}$ , Positive $V_{Port}$
9	LED indicator: status of the link		
10	LED indicator: communication activity		

### RJ45 Modbus Link Connector

The following illustration shows the shielded non-insulated RJ45 connector for the RS485 Modbus link:



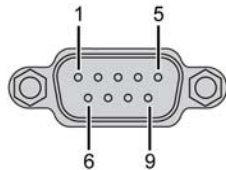
The following table shows the pinouts of the RJ45 Modbus link connector:

No.	Signal
1	NC
2	NC
3	NC
4	D1 (B/B')
5	D0 (A/A')
6	NC
7	NC
8	Common

**NOTE:** The pinout conforms to the Schneider Electric standard. The connector is master and includes the polarity on the line.

### RS232 Serial Modem Link Connector

The following illustration shows the 9-pin SUB D male connector for the RS232 serial modem link:



The following table shows the pinouts of the RS232 serial modem link connector:

Pin	Signal
1	DCD
2	RD
3	TD
4	DTR

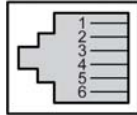
Pin	Signal
5	GND
6	DSR
7	RTS
8	CTS
9	RI

**NOTE:** This connector conforms to the PC standard.

### RJ11 Internal Modem Link Connector

Use the RJ11 connector (socket) to connect the telephone line to the internal modem of the TSX ETG 3010.

**NOTE:** The internal modem is featured on the TSX ETG 3010 module only.



The following table shows the pinouts of the RJ11 internal modem link connector:

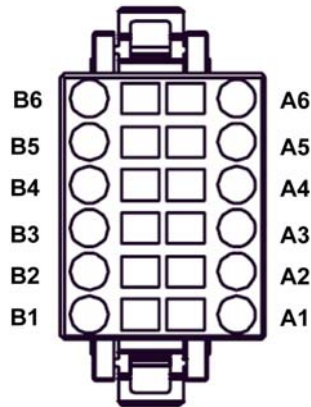
Pin	Signal
1	NC
2	NC
3	Ring
4	Tip
5	NC
6	NC

### Digital I/O Board

A digital I/O connector is provided with the TSX ETG 30\*\*. This interface is used to connect an external I/O device.

**NOTE:** When preparing the cable to connect the wiring, check the pin numbers inscribed on the digital I/O connector.

For replacement of the digital I/O connector, order part number TSXETGACC01.



The following table shows the pinouts of the digital I/O connector:

Pin	Signal	Pin	Signal
B6	IN0	A6	IN1
B5	IN2	A5	IN3
B4	IN4	A4	IN5
B3	COM	A3	NC
B2	OUT0	A2	OUT1
B1	+24 v	A1	0 V

**NOTE:**

- For more information concerning the I/O board characteristics, refer to the I/O board part (*see page 62*).
- For information concerning mounting the digital I/O connector, refer to the wiring to the I/O board part (*see page 53*).

**FME Male Connector**

An antenna for the embedded GSM modem is provided with the TSX ETG 302• module. Connect it to the FME male connector available on the front panel.

The characteristics of the antenna are:

Characteristics	
Type	Micromagnetic - 1/4 wave
Frequency	GSM/DCS Tri-band 900/1800/1900 MHz
Impedance	50 Ω
Gain	0 dB

Characteristics	
Weight	About 78 g (2.75 oz)
Base size	∅ 30 mm (1.18 in)
Height	7 cm (2.76 in)
Cable	RG174 ∅ 2.54 mm (0.1 in)
Cable length	250 cm ± 10 cm (98.4 in ± 3.94 in)
Connector	FME-female

**NOTE:** The antenna is mandatory if you want to use the GSM/GPRS modem.

### CF Card

The TSX ETG 30\*\* modules provides a standard CF card connector available on the front panel. It enables storage extension by inserting a CF card of 1 GB (available in the Schneider Electric catalog).

**NOTE:**

- TSX ETG 30\*\* modules only support CF cards that have been formatted in FAT or FAT32 format.
- Schneider Electric recommends you only use CF cards manufactured by Schneider Electric. The performance of the TSX ETG 30\*\* cannot be guaranteed when using CF cards from another manufacturer.

### USB Memory Key

The TSX ETG 30\*\* modules provides a standard USB type A (host) connector available on the front panel. It enables storage extension by inserting a USB memory key.

**NOTE:**

- USB memory keys do not have to comply to the same rigorous industrial standards and environmental constraints that the TSX ETG 30\*\* modules must meet.
- Schneider Electric recommends that you do not use USB memory keys for permanent operations. Instead, use the CF card feature, which offers greater stability.
- TSX ETG 30\*\* modules only support USB memory keys that have been formatted in FAT or FAT32 format.

## I/O Board

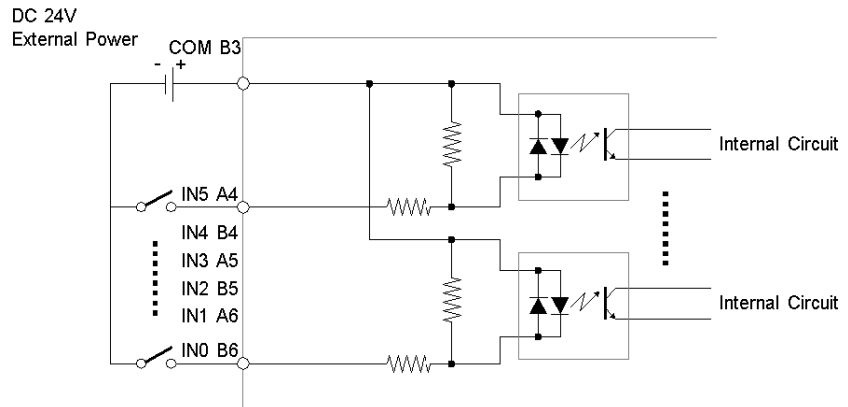
### Presentation

The I/O board is composed of 6 digital inputs and 2 digital outputs. You can read/write the value of the I/O using a Modbus/TCP request (*see page 110*) or using the calculation service (*see page 286*).

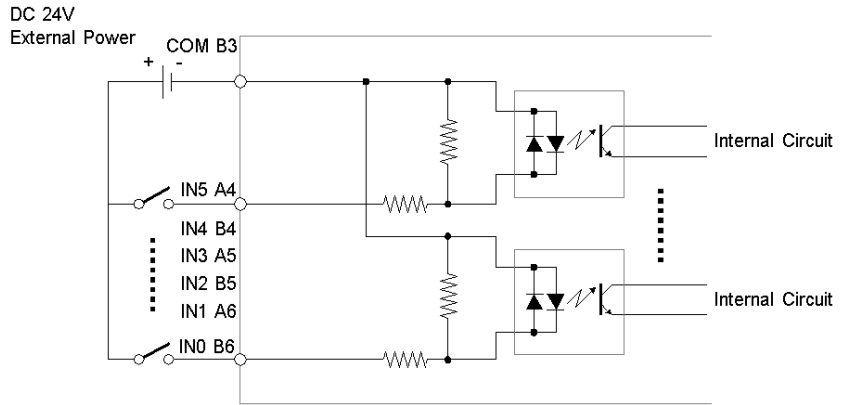
#### NOTE:

- The module maintains output values at module soft reboot. In case of hard reboot, the outputs are set to 0.
- The I/O board is only available with the TSX ETG 3010 and TSX ETG 302• modules.
- For more information concerning the I/O connector, refer to the module connectors part (*see page 59*).
- For information concerning mounting the digital I/O connector, refer to the wiring to the I/O board part (*see page 53*).

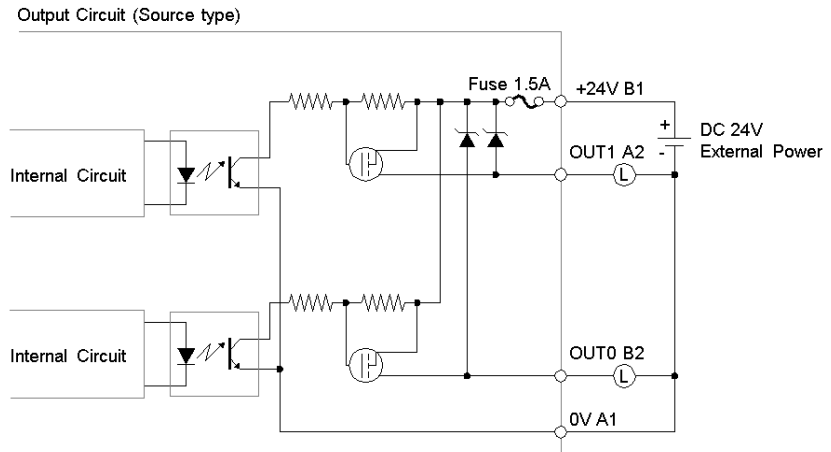
### Interface Input Circuit Using Sink Output



### Interface Input Circuit Using Source Output



### Interface Output Circuit



**NOTE:** Since the output terminals are not electrically protected, an output line could short circuit or a connection interruption could damage the TSX ETG 30•• module. Please install an applicable fuse to prevent an overload in the circuit if necessary.

## I/O Scanning

The I/O scanning service is a stand-alone communications task that exchanges register data between one device running the service and many remote devices on the network using Modbus TCP. The service requires only a simple configuration operation in the I/O scanner device; no special programming is required in either the I/O scanner device or the remote devices.

The I/O scanner reads and writes data repetitively in a user-configurable period ranging from 20 ms to 5 s. These read/write exchanges generate a load on the network. For this reason, the I/O scanner is best suited for critical periodic operations.

## Characteristics

The following tables show the characteristics of the I/O board:

<b>Input Characteristics</b>	
Rated voltage	24 Vdc
Maximum allowable voltage	28.8 Vdc
Input method	Source input
Rated current	5.7 mA (24 Vdc)
Input resistance	4.2 k $\Omega$
Operation range	ON voltage: 15 Vdc or more
	OFF voltage: 5 Vdc or less
Input delay time	OFF to ON: 1.5 ms or less
	ON to OFF: 1.5 ms or less
Common lines	1
Common design	6 points/1 common line
External connection	12 pins connector (used for output)
Input points	6
Input signal display	No LED indicators
Status display	None
Isolation method	Photocoupler isolation
Dielectric strength	1000 V
Isolation resistance	10 m $\Omega$
External power supply	24 Vdc



<b>Output Characteristics</b>	
Rated voltage	24 Vdc
Maximum allowable voltage	28.8 Vdc
Output method	Source output
Maximum load voltage	0.2 A/point, 0.4 A/common
Output voltage drop	1.5 Vdc or less
Output switching time	OFF to ON: 1.0 ms or less
	ON to OFF: 1.0 ms or less
Voltage leakage (when OFF)	0.1 mA or less
Clamp voltage	39 Vdc $\pm 1$ Vdc
Type of output	Transistor output
Common lines	1
Common design	2 points/1 common line
External connection	12 pins connector (used with input)
Output protection type	Unprotected
Internal fuse	1.5 A, 125 V DIP fuse (not replaceable)
Surge control circuit	Zener diode
Output points	2
Output signal display	No LED indicators
Status display	None
Isolation method	Photocoupler isolation
External power supply	24 Vdc

## Connection Cables

### Presentation

Various cables can be used for connecting the TSX ETG 30\*\* module. These cables can be obtained from the Schneider Electric catalog or can be constructed by the user.

### Ground Connection

Functional Ground (FG) is available on the power connector in order to discharge high frequency disturbances. The module is grounded using the metal plate at the rear of the module. When the module is in place, these metal plates are in contact with the metal of the support plate. Shielded connector (USB, RJ45 Ethernet, RJ45 Modbus, SubD9 and DIO) are also connected to FG. However to avoid an electric shock caused by indirect contact, FG must be linked to the protective ground.

For more information on grounding the module, refer to the *Grounding and Electromagnetic Compatibility of PLC Systems Basic Principles and Measures User Manual*.

### Ethernet Network Connection Cables

For connecting to the Ethernet network we recommend using cables with 100  $\Omega$  RJ45 STP (shielded twisted pair) connecto (10/100base-T interface) or category 5 Ethernet cables conforming to standard TIA/EIA-568A.

Power over Ethernet (PoE): TSX ETG 3000 can be supplied using Ethernet port 1. In this case, you can use an Ethernet switch compliant with PoE or a PoE injector (Reference: TCSEAV0100).

**NOTE:** Operating Temperature and EMC compliance for the PoE injector are not at the same level as those for the TSX ETG 3000.

**RS232 Cable to PC**

The RS232 cable to the PC is a standard commercial shielded DTE/DTE type crossed cable. This type of cable is also referred to as a "null modem" cable by certain suppliers.

Example of a suitable cable type: EYN257H-0006-FF from Black Box.

Connectors: DB9-S Sub D with UNC-4-40-2B screws

Wiring:

9-pin female Sub D			9-pin female Sub D		
2	RD		3	TD	
3	TD		2	RD	
4	DTR		6 and 1	DSR + DCD	
5	GND		5	GND	
6 and 1	DSR + DCD		4	DTR	
7	RTS		8	CTS	
8	CTS		7	RTS	
9	NC		9	NC	
Body	-	Shielding braid	Body	-	Shielding braid

**NOTE:** The DSR and DCD signals can be tied together to emulate on-line behavior.

**RS232 Cable to Modem**

An external modem can be connected on the RS232 port. You must use cable compliant with your external modem specifications.

Usually external modems use a straight cable Sub D9 female to Sub D9 male.

## 3.4 Diagnostics

### LED Indicators

#### Introduction

On the front panel of the module there are LED indicators used for performing a first level diagnostics on the TSX ETG 30\*\* module.

#### Front Panel

Illustration of the LEDs:



**ETH** Ethernet diagnostic

**RUN** Module's status

**ERR** Error indicator

**CH0** RS232 diagnostic

**CH1** Modem diagnostic

**CH2** Modbus diagnostic

**BAT** Battery level indicator

**EXT** CF or USB memory indicator

Meaning of the diagnostics LEDs:

ETH	RUN	ERR	CH0	CH1	CH2	BAT	EXT	Meaning
<b>ON = Permanently on, F = flashing, xF = x time flashing, OFF = Off, – = not significant.</b>								
<b>Hardware Status</b>								
OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF	No supply to module.
F	F	F	F	F	F	F	F	Module running self-test.
OFF	ON	OFF	OFF	OFF	OFF	OFF	OFF	Module ready.
OFF	OFF	ON	OFF	OFF	OFF	OFF	OFF	Firmware detected error.
–	–	–	–	–	–	ON	–	Battery level is low (it must be changed as soon as possible).
–	–	ON	–	–	–	–	–	PIN code detected error. See note below.

ETH	RUN	ERR	CH0	CH1	CH2	BAT	EXT	Meaning
<b>ON = Permanently on, F = flashing, xF = x time flashing, OFF = Off, – = not significant.</b>								
<b>Ethernet Status</b>								
ON	ON	–	–	–	–	–	–	Ethernet link ready using the IP address configured.
3F	ON	–	–	–	–	–	–	Ethernet cable disconnected.
4F	OFF	–	–	–	–	–	–	Duplicated IP address detected.
5F	ON	–	–	–	–	–	–	Module configured as a BOOTP client and is waiting for a BOOTP server response.
6F	ON	–	–	–	–	–	–	Configured IP address not found. The Ethernet link is ready but it uses the default IP address derived from the MAC address.
<b>Modem Status</b>								
–	ON	–	–	ON	–	–	–	PPP link established.
<b>RS232 Status</b>								
–	ON	–	F	–	–	–	–	RS232 communication. It flashes according to the communication speed.
<b>Modbus Status</b>								
–	ON	–	–	–	F	–	–	Modbus communication. It flashes according to the communication speed.
<b>External Status</b>								
–	–	–	–	–	–	–	F	CF card or USB access.

**NOTE:** Type the PIN code of the SIM card carefully. Entering an incorrect PIN code three consecutive times will lock the SIM card. If this occurs, contact your phone provider to unlock it.

### Ethernet Connector

There are 2 LEDs on the Ethernet connectors:

- the highest one indicates the status of the link,
- the lowest one indicates the connection activity.

## 3.5 Electrical Characteristics

### Electrical Characteristics

#### Introduction

The TSX ETG 30\*\* is a standalone module requiring a 12 or 24 Vdc power supply.

#### Characteristics

Parameter	Model	Minimum	Nominal	Maximum
Supply voltage	TSX ETG 30**	9 Vdc	12 or 24 Vdc	30 Vdc
Shutdown		–	–	1 ms
Ripple factor		–	–	5%
In rush current		–	–	25 A
Power consumption	TSX ETG 3000	–	5.7 W	6.8 W
	TSX ETG 3010		6.5 W	7.8 W
	TSX ETG 3021		7.2 W	8.6 W
	TSX ETG 3022		7.2 W	8.6 W

**NOTE:** The power supply input is protected against accidental polarity inversions.

---

## 3.6 Standards

---

### Standards

#### Compliance with Standards

The TSX ETG 30•• module complies with the following standards:

- International standard:
  - ISO/IEC 8802-3
  - ANSI/IEEE Std 802.3-2002
  - IEEE Std 802.3af-2003 as PD
- US standard:
  - UL 508
  - FCC Class A
- European standard:
  - RoHS Directive 2002/95/EC
  - CE Marking
  - R&TTE Directive 1999/5/EC
- Canadian standard:
  - CSA C22.2 No. 142

## 3.7 Conditions of Use

### Conditions of Use

#### Applicable Conditions

Characteristic		
Operating temperature	TSX ETG 3000	-25...75° C
	TSX ETG 3010	
	TSX ETG 3021	0...60° C
	TSX ETG 3022	
Storage temperature	TSX ETG 3000	-40...85° C
	TSX ETG 3010	
	TSX ETG 3021	-30...85° C
	TSX ETG 3022	
Relative humidity	TSX ETG 3000	10...95% (without condensation)
	TSX ETG 3010	20...90% (without condensation)
	TSX ETG 3021	
	TSX ETG 3022	
Altitude	TSX ETG 30**	0...2000 m
Vibration resistance	TSX ETG 30**	EN/IEC 60068-2-6 Fc
Impact resistance	TSX ETG 30**	EN/IEC 60068-2-27 Ea

**NOTE:** CF cards and SIM cards do not have to comply to the same conditions of use. Refer to the manufacturer documentation for the specific conditions of use of these devices.



---

# Module Setup

# 4

---

## Scope of this Chapter

This chapter describes how to setup a TSX ETG 30•• module.

## What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
4.1	Hardware Configuration	74
4.2	Topology Examples	81
4.3	Modbus Communication	92
4.4	Internal Module Registers	101
4.5	TCP/IP Messaging	111
4.6	SOAP / XML Communications	125
4.7	PSTN, GSM and External Modems	148
4.8	GPRS Modem Communications	161
4.9	GPRS Communications and NAT Routing Functions	176
4.10	GPRS Communications and VPN Functions	198
4.11	Common Modem Features	215
4.12	Ending a Connection	219
4.13	Remote Connection Availability: Modem Stand-by Function	223
4.14	DHCP (FDR) Service	228
4.15	DNS Service	233
4.16	NTP Service	237
4.17	SNMP server	240
4.18	Device Explorer	246
4.19	FTP Service	253
4.20	Power Over Ethernet	254
4.21	Daisy Chain	256

## 4.1 Hardware Configuration

---

### Overview

This section describes the hardware configuration (modem, Ethernet, Modbus...) of the TSX ETG 30\*\* using the embedded website of the module.

**NOTE:** You can also configure the TSX ETG 30\*\* using *Web Designer for TSX ETG 30\*\**.

### What's in this Section?

This section contains the following topics:

Topic	Page
Accessing the Module Configuration	75
Configuration of your Computer for Direct Connection by Serial Link	77
Changing Passwords	79

## Accessing the Module Configuration

### Presentation

When accessing the configuration pages for the first time, you can use the Ethernet link or the RS232 serial link. After configuration of modem connection, you will be able to access configuration pages also via modem.

### Configuration via Ethernet Link

To access the module configuration pages for the first time, via an HTTP link, follow the steps below:

Step	Action
1	Connect the module to the Ethernet network and power up.
2	Open your Internet browser on the PC.
3	<p>In the <b>Address</b> field enter the command: <code>http://&lt;default_ETG_IP_address&gt;</code>, then press <b>Enter</b>.</p> <p><b>Note:</b></p> <ul style="list-style-type: none"> <li>• The default IP address is derived from the module's MAC address, which is printed on the front panel (refer to the next paragraph for more information).</li> <li>• Your PC has to be in the same IP address range based upon the subnet mask.</li> </ul> <p><b>Result:</b> The home page appears on the screen.</p>
4	Click the <b>Setup</b> link.
5	The configuration index page appears: click the service you require.
6	Enter the default username <code>USER</code> and the default password <code>USER</code> (without quotation marks), then press <b>Enter</b> .

**NOTE:** For subsequent connections proceed in the same way, but enter the appropriate IP address, username and password if they have been changed.

### Default IP address

The factory\_default IP address of the module is 10.10.x.y, where x and y can be obtained by converting the two last digits of the MAC address, from hexadecimal to decimal.

If the module MAC address is 00.80.F4.01.53.CF, then the IP address of the module will be 10.10.83.207. In hexadecimal format 53 becomes 83, in decimal format, CF becomes 207.

### Configuration via RS232 Serial Link

For the first connection you must configure the components of your operating system (*see page 77*) in order to be able to communicate with the TSX ETG 30\*\* module. Once these components have been configured, follow the steps below:

Step	Action
1	Connect an RS232 crossed cable ( <i>see page 67</i> ) between a COM port on the PC and the 9-pin SUB D socket on the TSX ETG 30** module.
2	Establish the serial connection between the PC and the module.
3	Open your Internet browser on the PC.
4	In the Address field type: <code>http://10.10.0.2</code> , then press <b>Enter</b> .
5	The configuration index page appears: click the service you require.
6	Enter the default UserName <code>USER</code> and the default password <code>USER</code> (without inverted commas), then press <b>Enter</b> .

**NOTE:** For subsequent connections, the IP address to be entered is always the same, but enter the appropriate username and password if they have been changed.

### Configuration via GSM or PSTN Link

For the first connection you must configure the modem (*see page 157*) in order to be able to communicate with the TSX ETG 30\*\* module. Once this component has been configured, follow the steps below:

Step	Action
1	Connect your PC to the internet.
2	Open your Internet browser on the PC.
3	In the Address field type: <code>http://10.10.0.4</code> (in case of connection by request from a remote device), then press <b>Enter</b> . <b>Note:</b> if the website does not appear, make sure you are authorized to access the network with your computer ( <i>see page 27</i> ).
4	The configuration index page appears: click the service you require.
5	Enter the default UserName <code>USER</code> and the default password <code>USER</code> (without quotation marks), then press <b>Enter</b> .

## Configuration of your Computer for Direct Connection by Serial Link

### Presentation

This document is an example of your computer configuration in order to access the TSX ETG 30•• modules http server using its serial link. The following procedure shows you the configuration for Windows XP. For other Microsoft OS, refer to the *Configuration of your computer for TSX ETZ direct connection by serial link* manual.

Although reasonable care has been taken to provide accurate and authoritative information in this document, no responsibility is assumed by Schneider Electric for any consequences arising from the use of this document.

### Configuration of the Computer

The following procedure shows you how to configure your computer for direct connection using Windows XP:

Step	Action
1	Click <b>Start</b> → <b>Control Panel</b> → <b>System</b> . <b>Result:</b> The System Properties window appears.
2	Select the <b>Hardware</b> tab and click <b>Device Manager</b> . <b>Result:</b> The Device Manager window appears.
3	Expand the <b>Ports (COM &amp; LPT)</b> item.
4	Double click the <b>Communication Port (COM1)</b> item to open it. <b>Result:</b> The Communications Port (COM1) Properties window appears.
5	Select the <b>Port Settings</b> tab and enter the following settings: <ul style="list-style-type: none"> <li>● Speed of 57600 bps</li> <li>● 8 data bits</li> <li>● No parity</li> <li>● 1 stop bit</li> <li>● Without flow control</li> </ul>
6	Validate the settings by clicking <b>OK</b> . <b>Result:</b> You are back to the Device Manager window.
7	Expand the <b>Modem</b> item.
8	Double click the <b>Communication cable between two computers</b> item. <b>Result:</b> The Communications cable between two computers Properties window appears.
9	Select the <b>Modem</b> tab.
10	Select a Maximum Port Speed of 57600 bps.
11	Validate the settings by clicking <b>OK</b> .

## Creating a Connection

The following procedure shows you how to create a serial link connection

Step	Action
1	Click <b>Start</b> → <b>Control Panel</b> → <b>Network Connections</b> .
2	Click <b>Create a new connection</b> on the menu on the left side. <b>Result:</b> The New Connection Wizard appears.
3	Expand the <b>Ports (COM &amp; LPT)</b> item.
4	Double click the <b>Communication Port (COM1)</b> item to open it. <b>Result:</b> The Communications Port (COM1) Properties window appears.
5	Select <b>Set up an advanced connection</b> and click <b>Next</b> .
6	Select <b>Connect directly to another computer</b> and click <b>Next</b> .
7	Select <b>Guest</b> and click <b>Next</b> .
8	Type a name for this connection and click <b>Next</b> .
9	Select the device Communications cable between two computers (COM1) and click <b>Next</b> .
10	Select <b>My use only</b> and click <b>Next</b> .
11	Validate the settings by clicking <b>Finish</b> . <b>Result:</b> The connection that you have just created appears in the Network Connections window.

## Setting up the Connection

The following procedure shows you how to configure the properties of the connection:

Step	Action
1	Double click the connection that you have created in the Network Connections window. <b>Result:</b> The <b>Connect xxxx</b> window appears (xxxx corresponds to the name of the connection).
2	Type the username and password of the HTTP server (default: USER/USER).
3	Click <b>Properties</b> .
4	In the General tab, select <b>Communications cable between two computers (COM1)</b> and click <b>Configure</b> . <b>Result:</b> The Modem Configuration window appears.
5	Enter the following settings: <ul style="list-style-type: none"> <li>● Speed of 57600 bps</li> <li>● Uncheck the all options</li> </ul>
6	Validate the settings by clicking <b>OK</b> twice.

## Changing Passwords

### Introduction

The **Security** page enables you to change:

- The HTTP user name and password for accessing the home page.
- The Write Data password for accessing data in write mode.
- The Configuration password for accessing the configuration parameters setup.

#### NOTE:

- The maximum length of the user name and password is 15 characters (a-z, A-Z and 0-9).
- The default values for the username and password fields protecting Web page access are **USER/USER** (username and password are both case sensitive).

### Accessing the Security Page

Step	Action
1	Click the <b>Setup</b> link on the <b>Home</b> page.
2	Click the <b>Security</b> link on the <b>Setup</b> page.

### View of the Security Page

**SECURITY**

HTTP Password

Configuration Password

Data Password

### Modifying a Password

Firstly, you have to choose which password you want to change:

Button	Action
<b>HTTP</b>	Change the password for accessing the website.
<b>Configuration</b>	Change the password for accessing the configuration parameters.
<b>Data</b>	Change the password for accessing the data in write mode.

**Modifying HTTP Access Rights**

Step	Action
1	Enter the current username.
2	Enter the current password (field is case-sensitive).
3	Enter the new password.
4	Confirm the new password.
5	Confirm the change by clicking <b>Apply</b> . <b>Result:</b> a confirmation window appears.
6	Click <b>Reboot</b> for the changes to take effect in the module.

**Changing the Data Password**

Step	Action
1	Enter the current password (field is case-sensitive).
2	Enter the new password.
3	Confirm the new password.
4	Confirm the change by clicking <b>Apply</b> . <b>Result:</b> a confirmation window appears.
5	Click <b>OK</b> for the changes to take effect in the module. <b>Result:</b> a window appears to confirm that the password has been changed successfully.

**Changing the Configuration Password**

Step	Action
1	Enter the current password (field is case-sensitive).
2	Enter the new password.
3	Confirm the new password.
4	Confirm the change by clicking <b>Apply</b> . <b>Result:</b> a confirmation window appears.
5	Click <b>OK</b> for the changes to take effect in the module. <b>Result:</b> a window appears to confirm that the password has been changed successfully.



---

## 4.2 Topology Examples

---

### Overview

This section shows you TSX ETG 30•• topology examples and relevant configurations. Additional explanations can be found in the succeeding sections.

### What's in this Section?

This section contains the following topics:

Topic	Page
Remote Access Topologies	82
Network Topologies	84

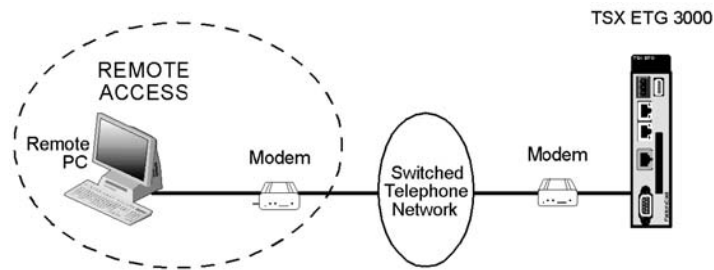
## Remote Access Topologies

### Presentation

The following topologies describe how to access remotely the TSX ETG 3000 modules using either embedded modems or external modems.

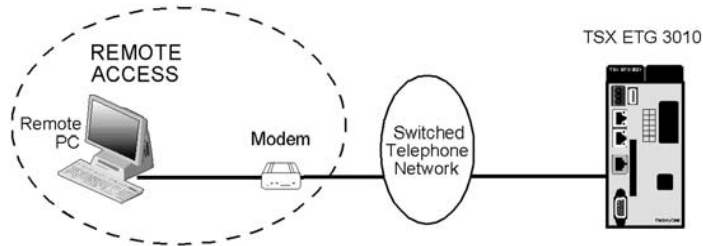
### External Modem

An external modem handles the communication between the module and the telephone network. If your module has an embedded modem, it must be disabled in order to use the external modem.



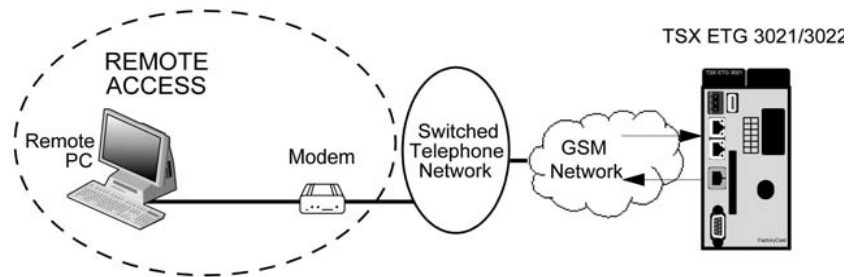
### Embedded PSTN Modem

The internal PSTN/RTC modem (TSX ETG 3010) handles the communication between the module and the telephone network.



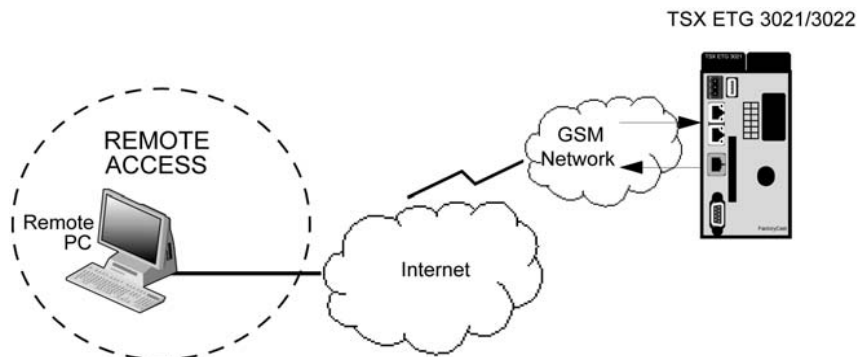
### Embedded GPRS Modem

The GPRS modem (TSX ETG 302•) handles the communication between the module and the telephone network.



### Embedded GPRS Modem

The GPRS modem (TSX ETG 302•) handles the communication between the module and the Internet.



## Network Topologies

### Overview

The TSX ETG 30\*\* offers various types of connection topologies options:

- for accessing device data
- for accessing maintenance
- for accessing the mail service
- for the RAS function
- for the routing function

**NOTE:** The following topologies show you examples of remote connections using a TSX ETG 3000. If you are using a TSX ETG 3010 or a TSX ETG 302\*, which provide internal modems, remote access (*see page 82*) to the module can be slightly different.

### Connection for Accessing Data

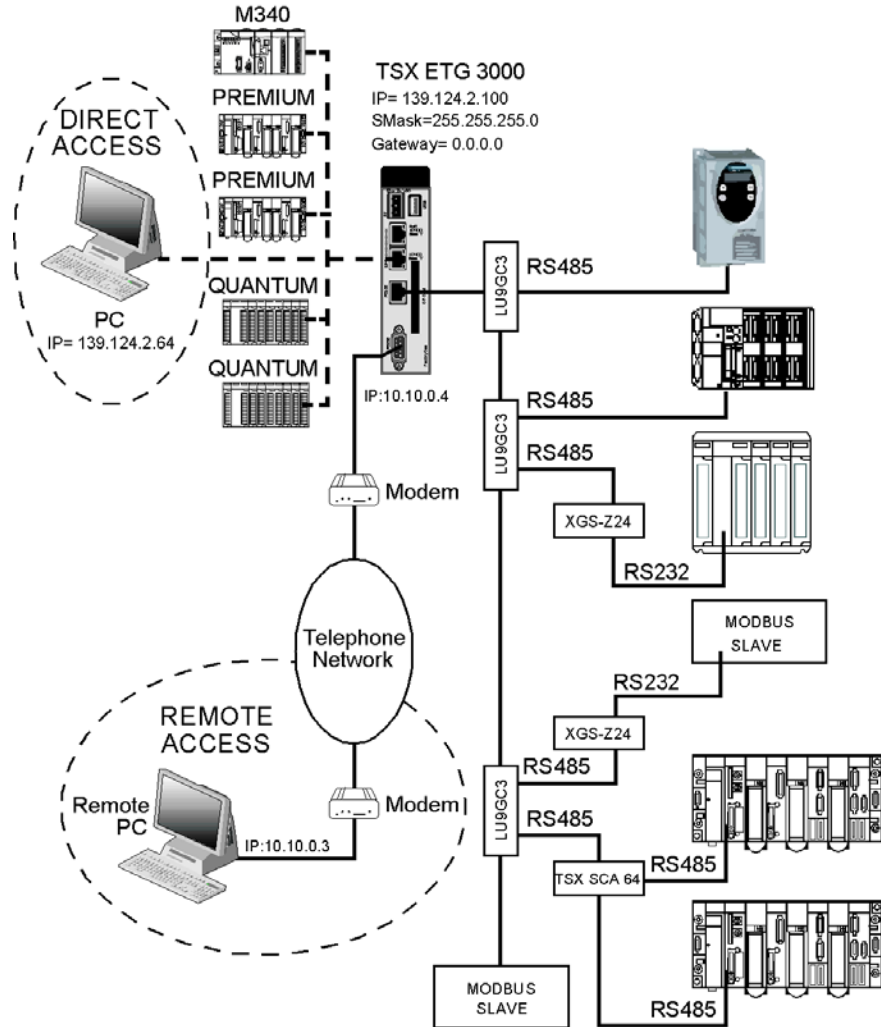
There are 2 ways of accessing remote device data:

- directly, where the PC is connected directly to the module via the Ethernet network or the serial link
- remotely, where the PC polls the module via the telephone network (via internal, external modem or GSM modem) and can therefore acquire device data

With either of these 2 connections, the user can access device data, internal registers, HTML pages, variable tables, animation tables, and screen pages.

**NOTE:** You can use the direct connection and the remote connection at the same time. In this case the TSX ETG 30\*\* has 2 IP addresses, one for the Ethernet network and another one for the PPP link.

The following figure shows the network topology for accessing registers:



**NOTE:** To connect LU9GC3 to a TSX ETG 3000, you must use cable reference VW3A8606R30 (RJ45-RJ45 cable).

For this example, you must configure the module as follows in the IP and modem configuration pages:

The image shows two configuration panels side-by-side. The left panel is titled 'Ethernet parameters' and contains three sections: 'Ethernet parameters' with 'Ethernet II' selected for frame format and '100-FD-Auto' for link speed; 'IP parameters' with 'Local' selected and IP address '139.124.2.100' and subnet mask '255.255.255.0'; and 'FDR parameters' with empty fields for device name and replication period. The right panel is titled 'Modem' and contains three sections: 'Modem' with 'External' selected; 'Server: local PPP Address' with 'Fixed' selected; and 'External Modem' with 'Commands HAYES', 'Time before connection (s)', 'Time detection', and 'Retry' as dropdowns, and 'RS232 baud rate' (57600), 'RS232 parity' (NONE), and 'RS232 stop bits' (1) as dropdowns. The bottom section is 'Line Management' with 'Idle time before hanging up' and 'Max time before hanging up' set to 'Never', and 'Hang-up after email' as an unchecked checkbox. Both panels have 'Apply' and 'Undo' buttons at the bottom.

### Connection for Email Services

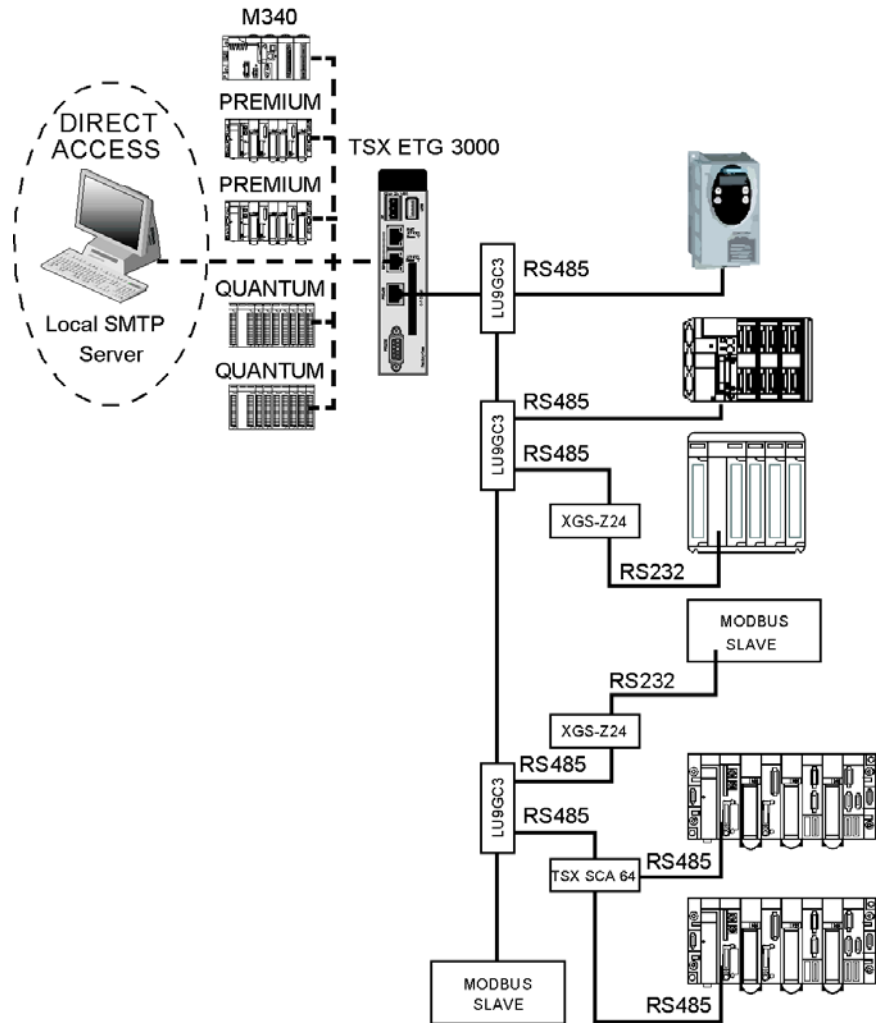
There are 2 ways of accessing email services:

- directly, where the SMTP server is connected directly to the module via the Ethernet network
- remotely, where the SMTP server polls the module via the telephone connection (via internal, external modem or GSM)

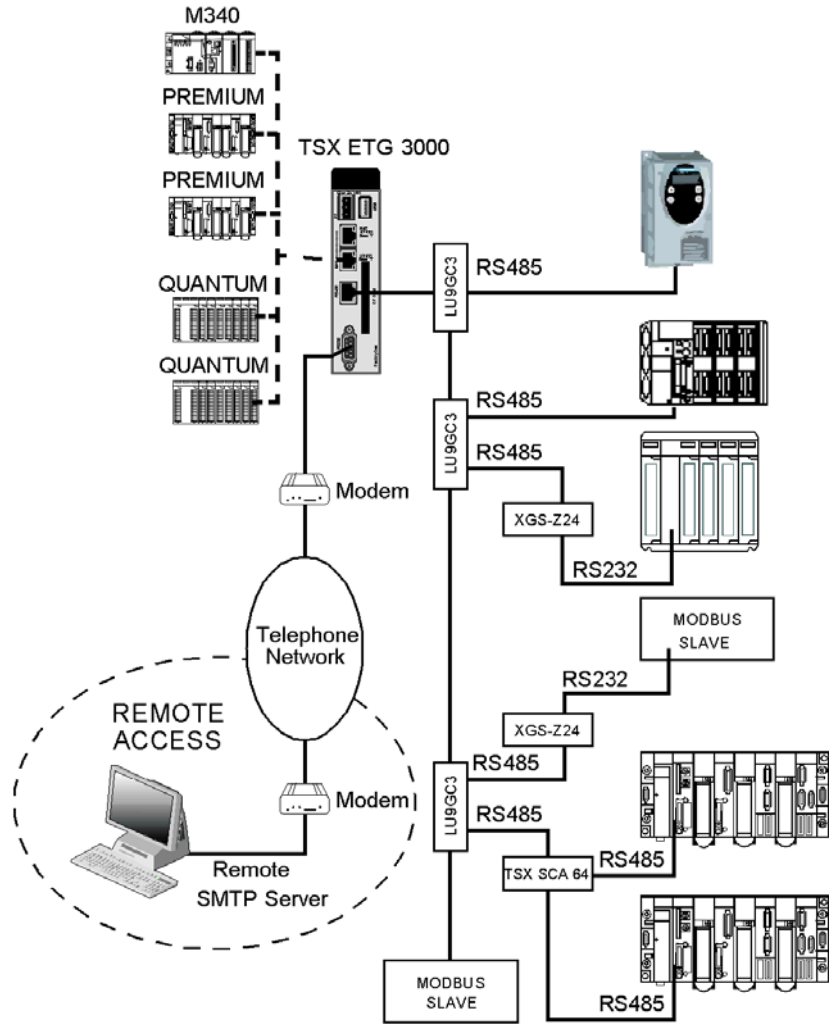
With either of these 2 connections, the SMTP server can receive all internal (module) or external (device) data or events (that have been set up) and then send a message to the suitable person.

**NOTE:** The 2 configurations cannot be combined because there can only be one SMTP server.

The following figure shows the network topology where the SMTP server is located on the same Ethernet network as the module:



The following figure shows the network topology where the SMTP server is connected to the module via a modem and the switched telephone network:

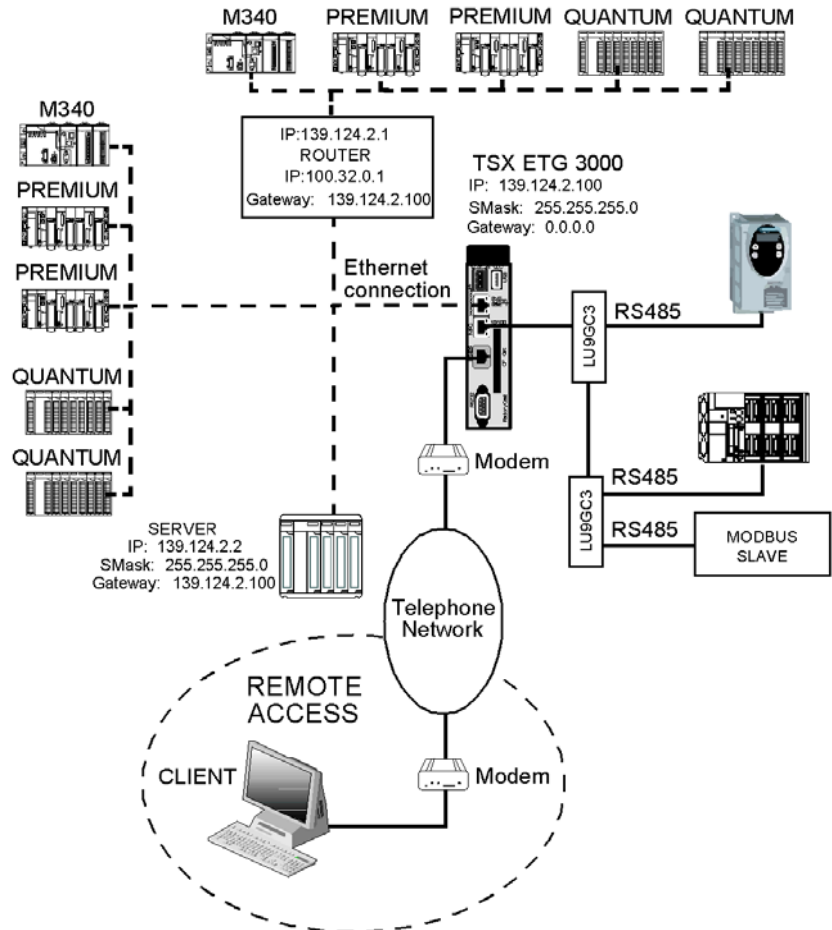




## Routing Connection

With this modem connection the user can access Modbus data, data from a remote server and data from another Ethernet network (via the router). In this case, any Ethernet configuration (fixed, server or using TCP/IP address) are suitable.

The following figure shows the network topology for a remote access:



**NOTE:** You must set the gateway address of the remote server and the gateway address of the router to the IP address of the TSX ETG 3000.

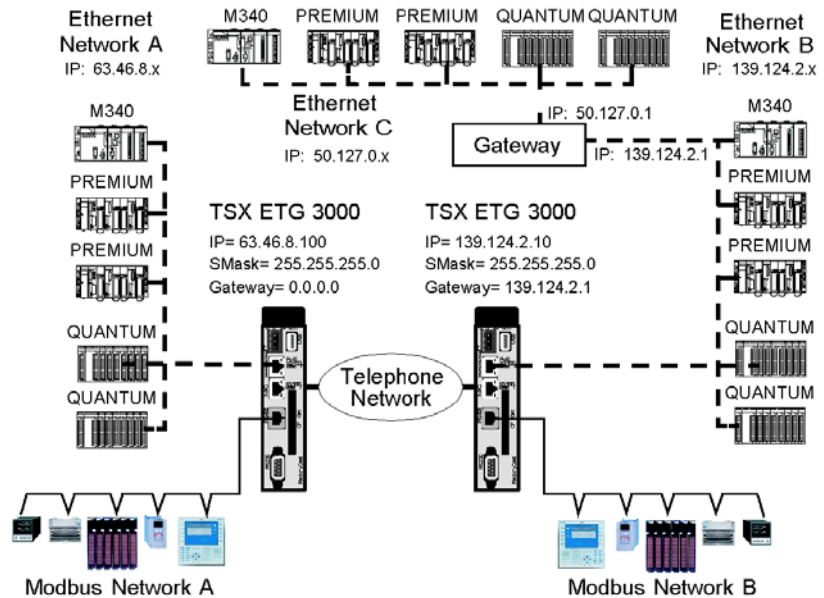
## Connection Between 2 TSX ETG 30\*\*

You can connect 2 TSX ETG 30\*\* modules via the PPP link and therefore connect networks together. There are 2 cases of connection between 2 TSX ETG 30\*\*:

- One of the Ethernet networks is connected to another network via a gateway.
- There is no gateway on both Ethernet networks.

**NOTE:** In this case, you can use a TSX ETG 1000 or TSX ETG 1010 instead of a TSX ETG 30\*\*.

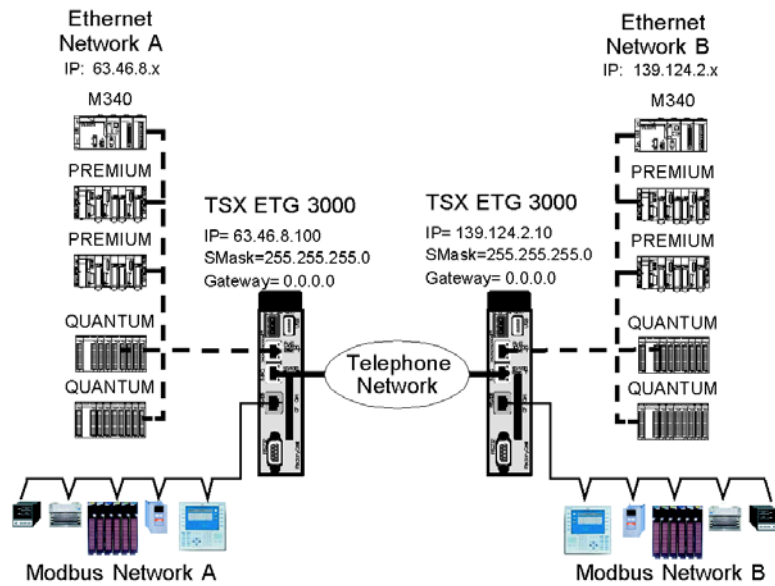
The following figure shows the network topology when there is a gateway on one of the Ethernet networks:



### NOTE:

- The TSX ETG 30\*\* module connected to Ethernet networks B and C must be client and any Ethernet configuration (fixed, server or using TCP/IP address) are suitable.
- The TSX ETG 30\*\* module connected to Ethernet network A must be server and it must use the same TCP/IP address for the PPP and the Ethernet link.

The following figure shows the network topology when there is no gateway on both Ethernet networks:



**NOTE:** In this case both TSX ETG 3000 modules can be either client or server.

## 4.3 Modbus Communication

---

### Overview

This section describes the Modbus communication function used by the TSX ETG 30\*\* module.

### What's in this Section?

This section contains the following topics:

Topic	Page
Modbus Communication	93
Modbus Server	96
Flat Addressing	98
Configuration Parameters for the Modbus Link	99
Modbus Performance	100

---

## Modbus Communication

### Presentation

Communicating via Modbus enables data exchange between all devices connected to the bus. The Modbus protocol is a protocol that creates a hierarchical structure (one master and several slaves). The master manages all exchanges in two ways:

- The master exchanges with the slave and awaits a response.
- The master exchanges with all the slaves without waiting for a response (general broadcast).

The TSX ETG 30\*\* module is able to manage:

- Modbus/TCP over Ethernet link,
- Modbus/RTU master over serial line.

#### NOTE:

- Modbus ASCII is not supported by the TSX ETG 30\*\*.
- Modbus/UMAS uses the same protocol than Modbus/TCP so the module can act as a gateway between Modbus/UMAS and Modbus/RTU.
- Serial line: the TSX ETG 30\*\* only manages Modbus/RTU.

### Principles

The TSX ETG 30\*\* module is a Modbus/TCP gateway for transmitting Modbus requests.

The TSX ETG 30\*\* module is the Modbus master. In order for the gateway to operate as Modbus master, the module has to be connected to a Modbus slave device.

As a Modbus master, the TSX ETG 30\*\* module manages exchanges with slave devices on the bus in a request/response manner. It sends a request to a slave device and awaits the response from that device. A transaction between the module and the devices on the Modbus network is always initiated by a request from the module.

Only one exchange can be made at a time: if the module is waiting for a response from a device, it cannot send a request. TCP requests are suspended on the network.

You can assign a Modbus address from 1 to 247 to the devices. The TSX ETG 30\*\* module is the Modbus server (*see page 96*) and has the Modbus address: 255.

The Modbus link (speed, parity, address, etc.) must be configured so that it is compatible with the slave link.

**NOTE:** If a Modbus device configured to be polled by the TSX ETG 30\*\* cannot be reached, the module considers the device absent during 30 s. In this way, the performance of the link do not decrease.

## Modbus TCP/IP Client to Modbus Serial Line

The table below shows a transaction between a Modbus/TCP client and a Modbus device over serial line:

Step	Action
1	The client over the Ethernet link sends a request to the TSX ETG 30** module (Modbus server).
2	The module performs a protocol conversion (Modbus/TCP to Modbus/RTU).
3	The module sends the request over serial line and it waits for a response.
4	On response arrival, the module performs a protocol conversion (Modbus/RTU to Modbus/TCP).
5	The module sends the response back to the Modbus device over the Ethernet link.

### NOTE:

- Incoming request from a device over Ethernet link are sent over serial line if the UnitID is not 255 (address of the TSX ETG 30\*\*).
- If the device over serial line does not reply to the request (step3), the TSX ETG 30\*\* module generates a Modbus exception response (0B hex).

## Modbus Broadcast

The master can address individual slaves, or can initiate a broadcast message to all slaves. Address 0 is used for the broadcast address, which all slave devices recognize. Responses are not returned to broadcast queries from the master.

As no response is expected, Modbus functions which imply a response are not supported by broadcast. The following table list these functions:

Function Code	Meaning
1	READ COILS
2	READ DISCRETE INPUTS
3	READ HOLDING REGISTERS
4	READ INPUT REGISTERS
7	READ EXCEPTION STATUS
8	DIAGNOSTIC
11	GET COM EVENT COUNTER
12	GET COM EVENT LOG
17	REPORT SLAVE ID
14/6	READ FILE RECORD
23	READ WRITE MULTIPLE REGISTERS

---

Function Code	Meaning
24	READ FIFO QUEUE
43/14	READ DEVICE IDENTIFICATION

Broadcast over serial line is disabled by default. You can enable it using the Modbus page of the website (*see page 99*) or using Web Designer for TSX ETG 30•• (refer to the *Web Designer for ETG•• User Manual*).

**NOTE:** If broadcast is disabled and the TSX ETG 30•• module receives a broadcast request (UnitID = 0), it generates a Modbus exception response (code: 0A hex) and it sends it to the client. Otherwise the TSX ETG 30•• module forwards the request to all the devices over serial line.

## Modbus Server

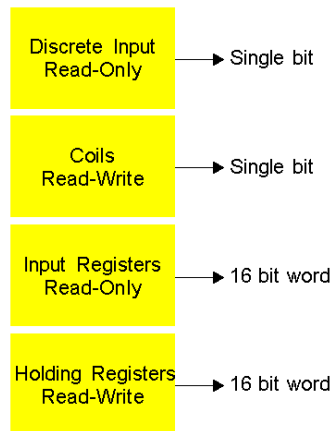
### Introduction

The TSX ETG 30\*\* module includes a Modbus server (regular class), that can be used to manage internal module variable addressing (via Modbus/TCP).

The Modbus address of TSX ETG 30\*\* module (Unit ID) is 255. This address enables you to access internal variables, in this case the module is seen as device.

### Modbus Addressing

The following figure shows the Modbus addressing model:



Memory images zones:

- The discrete Input block stores the memory image of the digital inputs.
- The coils block stores the memory image of the digital outputs.
- The holding Registers block stores the memory image of the internal registers.

You can read/write these variables using:

- the Data Editor of the embedded website,
- the Data Editor Lite of the embedded website,
- the Pocket Editor,
- the Graphic Editor/Viewer of the embedded website,
- the Custom Web Pages (widgets),
- any Modbus / TCP client.



## Modbus Requests

The Modbus server manages the following requests:

Function code	Value
FC 02	Read Discrete Input
FC 03	Read Multiple Registers
FC 05	Write Single Coil
FC 06	Write Single Register
FC 15	Write Multiple Coils
FC 16	Write Multiple Registers
FC 22	Write Mask Registers
FC 43/14	Read Device Identification

## Flat Addressing

### Introduction

TSX ETG 30\*\* modules follow a system of flat address mapping in Unity Pro. To work properly, each module requires a determinate number of bits and/or words. The IEC addressing system is equivalent to the 984LL register addressing. Use the following assignments:

- 0x is now %Mx
- 1x is now %Ix
- 3x is now %IWx
- 4x is now %MWx

**NOTE:** Modbus addresses start at 0 (400001 = 0)

The following table shows the relationship between 984LL notation and IEC notation.

Outputs and Inputs	984LL Notation Register Addresses	Data Type	Modbus Type	IEC Notation		
				System Bits and Words	Memory Addresses	I/O Addresses
output	0x	BOOL	Coil	System Bit	%Mx	%Qx
input	1x	BOOL	Discrete Input	System Bit	%Ix	%Ix
input	3x	16-bit INT	Input Register	System Word	%IWx	%IWx
output	4x	16-bit INT	Register	System Word	%MWx	%QWx

### Examples

The following examples show the relationship between 984LL register addressing, IEC addressing and Modbus addressing for Graphic and Data Editors (*see page 399*):

984LL Notation	IEC Notation	Modbus Type
000001	%M0	coil at address 0
100101	%I100	discrete input at address 100
301024	%IW1023	input register at address 1023
400010	%MW9	register at address 9

## Configuration Parameters for the Modbus Link

### Modbus Configuration Page

You can access the Modbus configuration page via the Setup menu of the website.

Serial line	
Baud rate	19200 ▼
Transmission mode	MODBUS RTU
Data bits	8
Parity	NONE ▼
Stop bits	1 ▼
Delay before new request (ms)	4 ▲▼
Response timeout (ms)	3000 ▲▼
Enable broadcast	<input type="checkbox"/>

### Parameters

Parameters	Value
Baud rate	Speed: 115200, 57600, 38400, 19200, 9600, 4800, 2400 baud
Transmission mode	Modbus RTU (cannot be changed)
Data bits	8 data bits (cannot be changed)
Parity	Even, odd, or no parity.
Stop bits	1 or 2 stop bits.
Delay before new request	1 to 999 ms
Response timeout	In milliseconds, from 1 to 10 seconds (default is 3 sec)
Enable broadcast	Check this box to enable broadcast on serial line
Edit buttons:	
Apply	Confirm the input
Undo	Cancel the current changes

**NOTE:** If broadcast is enable and a broadcast request is sent, all devices on serial line read the request but never send back a Modbus response.

## Modbus Performance

### Performance

The table below shows the Modbus performance. It shows the number of Modbus requests per second based on the bus speed and the request length.

<b>Requests/sec</b>	<b>Request length (in words)</b>					
<b>Modbus speed</b>	<b>64</b>	<b>32</b>	<b>16</b>	<b>8</b>	<b>4</b>	<b>2</b>
2,400 baud	2	3	5	7	8	9
4,800 baud	4	8	12	17	20	22
9,600 baud	9	16	22	28	34	38
19,200 baud	16	26	35	46	55	61
38,400 baud	28	42	59	61	88	95

Example:

35 Modbus slaves can be polled with 16-word requests if you set the Modbus speed to 19,200 baud.

---

## 4.4 Internal Module Registers

---

### Overview

This section describes the internal module registers of the TSX ETG 30\*\* module.

### What's in this Section?

This section contains the following topics:

Topic	Page
Local Registers	102
Date and Time Internal Registers	108
Input/Output Registers	110

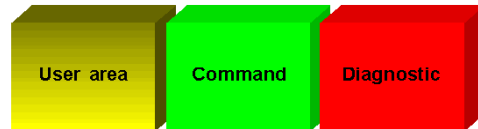
## Local Registers

### Register Areas

The TSX ETG 30\*\* module provides 3 register areas:

- user area (0 to 255)
- command area (1000 to 1005)
- diagnostic area (2000 to 2113)

The following figure shows the register areas of the TSX ETG 30\*\*:



#### NOTE:

- Services are able to access local registers using symbolic names. Symbolic names are defined using Web Designer for TSX ETG 30\*\*.
- All local registers are reset to initial values on module reboot.

### User Area

Use the user area to:

- store the services status values
- simulate a Modbus device

The user area (registers 0 to 255) is available for unrestricted use. You can read or write to this area using Modbus/TCP requests, the Data and Graphic Editor (data table/graphics objects), widgets or persistent variables.

The TSX ETG 30\*\* module manages the user area as registers of a Modbus/RTU device (UnitID = 255). The registers 0 to 255 have the following symbolic names: WORD0 to WORD255.

## Command Area

Use the command area (registers 1000 to 1005) to:

- open/close a PPP connection (command 1 or 2)
- open/close a PPP connection over leased line (command 3 or 4)

Register	Symbolic Name	Description
1000	STATUS	Command status for register 1: <ul style="list-style-type: none"> <li>● = 0 Command OK.</li> <li>● = FFFF hex (-1) Command in progress.</li> <li>● = FFEC hex (-20) No dial tone, line busy or no carrier.</li> <li>● = FFE8 hex (-24) Link down. The server has lost the connection.</li> <li>● = FFE7 hex (-25) Connection closed due to hang up timeout anomaly or idle time.</li> <li>● = FFCE hex (-50) Unknown telephone number.</li> <li>● = FFCD hex (-51) Different telephone number between the line disconnected and the line already connected.</li> <li>● = FFCC hex (-52) No modem detected.</li> <li>● = FFCA hex (-54) PPP link already established. Connection hex refused.</li> <li>● = FFC9 hex (-55) Anomaly while closing the line.</li> <li>● = FFC4 hex (-60) PPP connection error (parameters not compatible, user/password error).</li> </ul>
1001	COMMAND	Command: <ul style="list-style-type: none"> <li>● = 0 No command.</li> <li>● = 1 Open a PPP connection.</li> <li>● = 2 Close a PPP connection.</li> <li>● = 3 Open a VPN connection.</li> <li>● = 4 Close a VPN connection</li> <li>● = 5 Call a remote TSX ETG 302• module to open a remote GPRS connection.</li> </ul>
1002	CMD_PARAM1	Command = index of the telephone number in the telephone list or index of the device in the VPN list if you use VPN.
1003 to 1005	CMD_PARAM2...4	Reserved

### NOTE:

- If the command is equal to 1 or 2, the CMD\_PARAM1 corresponds to the index of the telephone table. The corresponding telephone number is composed automatically and is used for the PPP connection.
- You must fill the parameter registers area before starting to fill the command registers area.

## Diagnostic Area

The diagnostic area (registers 2000 to 2113) provides the following information:

- system statistics (*see page 104*)
- PPP statistics (*see page 104*)
- port 502 statistics (*see page 105*)
- serial line statistics (*see page 105*)
- GPRS statistics (*see page 106*)
- VPN statistics (*see page 107*)
- stand-by statistics (*see page 106*)
- read and write date and time (*see page 107*)

## System Statistics

Register	Symbolic Name	Description
2000	CPU_LOAD	CPU load in %. This value is computed every 60 seconds.
2001	ANIM_1	Changes its value (0 or 1) every 400 ms
2002	ANIM_4	Incremented by 1 every 400 ms. Displays value modulo 4
2003	ANIM_8	Incremented by 1 every 800 ms. Displays value modulo 8
2004	SECONDS	Seconds after minutes [0... 59]
2005	MINUTES	Minutes after hours [0...59]
2006	HOURS	Hours after midnight [0...23]
2007	DAY	Days of the month [1...31]
2008	MONTH	Month [1...12]
2009	YEAR	Year since 2000
2054	DAY_OF_WEEK	Day of the week (0=Sunday, 1=Monday...). This value is automatically calculated.

**Note:** The registers 2004 to 2009 give the date and time of the module. If the NTP service is not operational, the date and time are initialized by the real time clock of the TSX ETG 30•• (*see page 390*) at the first start.

## PPP Statistics

Register	Symbolic Name	Description
2010	STATUS_PPP	PPP connection status: <ul style="list-style-type: none"> <li>● = 0 for inactive</li> <li>● = 1 for client connection</li> <li>● = 2 for server connection</li> </ul>
2011 to 2014	PPP_IPi_ADD_REMOTE (where i = 1...4)	<ul style="list-style-type: none"> <li>● PPP: IP address of the remote device</li> <li>● if i = 0: PPP line closed</li> <li>● 2011: PPP_IP1_ADD_REMOTE</li> </ul>



Register	Symbolic Name	Description
2015 to 2018	PPP_IPi_ADD_ETG (where i = 1...4)	PPP: IP address of the FactoryCast gateway, if = 0 PPP line closed
2019 to 2022	PPP_IPi_ADD_FORCONNECT	IP address opened by a client PPP connection; if = 0, PPP line closed
2024 (LSB) to 2025 (MSB)	PPP_NB_FRAMES_SENT	Number of PPP frames sent
2026 (LSB) to 2027 (MSB)	PPP_NB_FRAMES_RECEIVED	Number of PPP frames received
2028 (LSB) to 2029 (MSB)	PPP_NB_CONNECT_OPEN_OK	Number of PPP connections opened correctly
2030 (LSB) to 2031 (MSB)	PPP_NB_CONNECT_OPEN_KO	Number of PPP connections opened with anomalies
2032 (LSB) to 2033 (MSB)	PPP_CONNECT_CLOSE_OK	Number of PPP connections closed correctly
2034 (LSB) to 2035 (MSB)	PPP_CONNECT_OPEN_KO	Number of PPP connections closed with anomalies
2036 (LSB) to 2037 (MSB)	PPP_CONNECT_SERVER_OPEN	Number of PPP server connections opened

### Port 502 Statistics

Register	Symbolic Name	Description
2038 (LSB) to 2039 (MSB)	PORT502_NB_CONN	Number of current connections opened
2040 (LSB) to 2041 (MSB)	PORT502_NB_MSG_SENT	Number of sent messages
2042 (LSB) to 2043 (MSB)	PORT502_NB_MSG_RECV	Number of received messages
2044 (LSB) to 2045 (MSB)	PORT502_NB_MSG_ERROR	Number of detected errors sent

### Serial Line Statistics

Register	Symbolic Name	Description
2046 (LSB) to 2047 (MSB)	RS485_NB_MSG_SENT	Number of sent messages
2048 (LSB) to 2049 (MSB)	RS485_NB_MSG_RECV	Number of received messages
2050 (LSB) to 2051 (MSB)	RS485_NB_MSG_ERROR	Number of CRC detected error

Register	Symbolic Name	Description
2052 (LSB) to 2053 (MSB)	RS485_NB_MSG_TIMEOUT	Number of timeouts
<b>Note:</b> Registers 2024 to 2053 must be displayed in int32Swap format in the Data Editor Lite.		
2060 to 2075	MODB_PRES_i_j	<p>Connection status of Modbus devices. Each register contains the state of 16 devices (1 bit per device).</p> <p><b>Example:</b> register 2061 corresponds to devices with UnitID = 16 to 31,</p> <ul style="list-style-type: none"> <li>● Bit = 1, the device is present.</li> <li>● Bit = 0, the device is not present.</li> </ul> <p><b>Example:</b> bit 15 of register 2061 = 0 -&gt; device with UnitID = 31 is not present.</p>
<b>Note:</b> Registers 2060 to 2075 are only updated on device request.		

### GPRS Statistics

Register	Symbolic Name	Description
2080 to 2083	GPRS_IPx_ADD_ETG with x from 1 to 4	GPRS IP address of theTSX ETG 302• ● 0: not connected
2084	GPRS_STATUS	GPRS connection status: ● 0: not connected ● 1: connected

### Stand-by Statistics

Register	Symbolic Name	Description
2095 (LSB) to 2096 (MSB)	SBY_NB_SWITCH	Number of switch
2097	SBY_ACTIVE	Active link: ● 0: Disable ● 1: Ethernet ● 2: GPRS
2098 (LSB) to 2101 (MSB)	SBY_LAST_SWITCH	Date and time of the last connection switch
2102 (LSB) to 2103 (MSB)	SBY_NB_TEST_OK	Number of modem connection test available
2104 (LSB) to 2105 (MSB)	SBY_NB_TEST_ERROR	Number of modem connection test not available
2106 (LSB) to 2109 (MSB)	SBY_LAST_TEST_KO	Date and time (RFC Modbus format) of the last modem detected connection error

## VPN Statistics

Register	Symbolic Name	Description
2085 to 2092	VPN_STATUS_x with x from 1 to 8	VPN Status: <ul style="list-style-type: none"> <li>● 0: Closed</li> <li>● 1: Open</li> </ul>
<b>Note:</b> VPN_STATUS_x for index x in the VPN setup list.		

## Read and Write Date and Time

Register	Symbolic Name	Description
2110	TIME_DATE_1	Years [0...99]
2111	TIME_DATE_2	Months [1...12] and days [1...31]
2112	TIME_DATE_3	Hours [0...23] and minutes [0...59]
2113	TIME_DATE_4	Milliseconds [0...59999]

For more information, refer to Date and Time Internal Registers (*see page 108*).

## IO Area

The IO area contains 2 registers:

- 3000: copy of the inputs of the module
- 3001: it enables you to read and write the outputs of the module

**NOTE:** The registers of the IO area do not have symbolic names.

Register	Symbolic Name	Description
3000	None	Bit 0 corresponds to input 0
		Bit 1 corresponds to input 1
		Bit 2 corresponds to input 2
		Bit 3 corresponds to input 3
		Bit 4 corresponds to input 4
		Bit 5 corresponds to input 5
3001	None	Bit 0 corresponds to output 0
		Bit 1 corresponds to output 1

## Date and Time Internal Registers

### Overview

The TSX ETG 30\*\* can synchronize date and time information with external tools (e.g. SCADA Vijeo Citect) by using internal registers.

Date and time information are defined on four consecutive registers:

- 2110: Year
- 2111: Months and days
- 2112: Hours and minutes
- 2113: Milliseconds

### Modbus RFC T1081 Standard

As a nonstandard date and time format could lead to interoperability problems, the format of the internal register follows the Modbus RFC T1081 standard (IEC 870-5).

### Read and Write Access

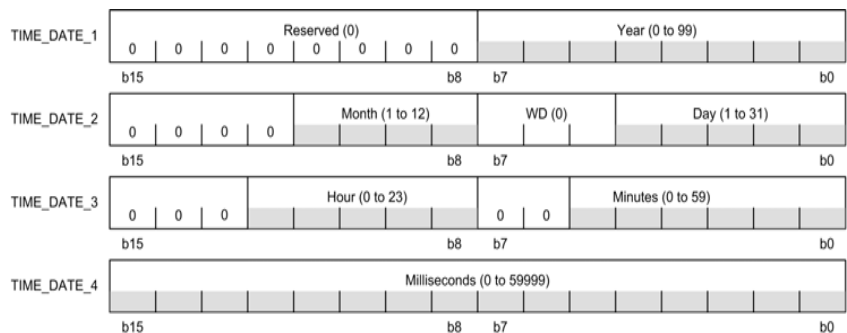
External tools can read or write the date and time information of the TSX ETG 30\*\*:

- Use the read register request (function 3 or 4) to read date and time information.
- Use the write register request (function 16) to set the TSX ETG 30\*\* slave to write the date and time information. This function uses 4 registers written in a single request.

### Operating Diagram

The date and time registers are organized in 4 consecutive words of 16 bits each.

The following diagram shows how the date and time registers are structured:



Registers	Symbolic Names	Description
2110	TIME_DATE_1	Years: 1 byte per year, varies from 0 to 99.
2111	TIME_DATE_2	Months: 1 byte per month, varies from 1 to 12.
		Days: 1 byte per day, varies from 1 to 31.
2112	TIME_DATE_3	Hours: 1 byte per hour, varies from 0 to 23.
		Minutes: 1 byte per minute, varies from 0 to 59.
2113	TIME_DATE_4	Milliseconds: 2 byte per millisecond, varies from 0 to 59999. <b>NOTE:</b> For devices that do not manage milliseconds, only seconds are taken into account.

Bits set to zero in the diagram are optional or not used and not generated by the slave. When not used, those bits are set to zero by the master:

Optional Date and Time Words	Description
WD	Day of the week: <ul style="list-style-type: none"> <li>● 0: Not used</li> <li>● 1: Sunday</li> <li>● 2: Monday...</li> <li>● 7: Saturday</li> </ul>
<b>NOTE:</b> For a write request, WD must be set to 0 (not used).	

## Input/Output Registers

### Memory Image of the Embedded Digital Inputs

Discrete Input Number	Symbolic Name	Value
0	INPUT0	Binary value of the digital input 0
1	INPUT1	Binary value of the digital input 1
2	INPUT2	Binary value of the digital input 2
3	INPUT3	Binary value of the digital input 3
4	INPUT4	Binary value of the digital input 4
5	INPUT5	Binary value of the digital input 5

### Memory Image of the Embedded Digital Outputs

Discrete Output Number	Symbolic Name	Value
0	OUTPUT0	Set/Reset output 0
1	OUTPUT1	Set/Reset output 1

**NOTE:**

For example, if OUTPUT0 or OUTPUT1 is set to:

- 1, the physical output is deactivated (OFF)
- 0, the physical output is activated (ON)

---

## 4.5 TCP/IP Messaging

---

### Overview

This section describes the TCP/IP messaging service of the TSX ETG 30\*\* module.

### What's in this Section?

This section contains the following topics:

Topic	Page
IP Addressing	112
TCP/IP Features	114
Default IP Address	115
Opening a Connection on the Ethernet Network	117
Retrieving the IP Address	120
Configuration Parameters for TCP/IP Services	121
IP Filtering Configuration	123

## IP Addressing

### Address Structure

An Internet Protocol (IP) address is a unique address that electronic devices use to identify and communicate with each other on a computer network. In other words, an IP address is the network address of a computer.

Each IP address consists of two elements, the network name and the device identifier. The network identifier identifies a network (or a site) and the device identifier identifies a device connected to this network. There are 3 classes of IP address.

### WARNING

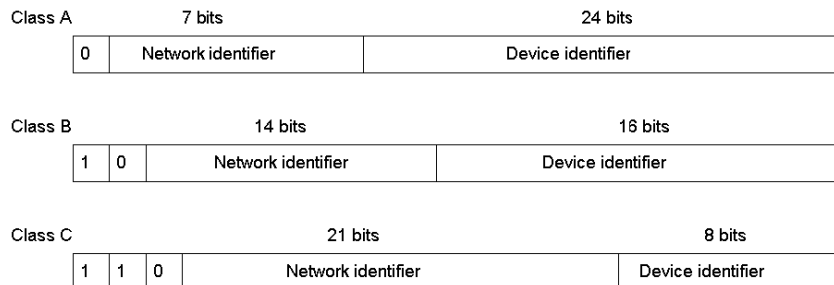
#### UNINTENDED OPERATION — DUPLICATE IP ADDRESS

- Make sure that this device will receive a unique IP address.
- Always obtain your IP address from your system administrator to avoid the possibility of duplicate addresses.

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

### Address Classes

The structure of the address classes is as follows:



Externally, an IP address of a device is represented by a string of four 8-bit values (0 to 255), separated by dots: "a.b.c.d".



Class	First address of the class	Last address of the class
A	0.0.0.1	127.255.255.254
B	128.0.0.1	191.255.255.254
C	192.0.0.1	223.255.255.254

## Sub-Addressing

The principle of sub-addressing is to divide the local part into a physical sub-network number and a device identification:

Format a	Internet part = network Id	Local part
----------	----------------------------	------------

Format b	Internet part	Physical sub-network number	Device identification
----------	---------------	-----------------------------	-----------------------

## Mask

A sub-network mask (Subnet Mask), coded in 32-bits, is used to define the bits of an IP address as the network part.

The mask bits are:

- Set to 1 if the bits corresponding to the IP address are to be interpreted as part of the network address
- Set to zero to identify the device

This system allows local internal networks to be addressed with a single attributed IP address.

Format a	Internet part = network Id	Local part
----------	----------------------------	------------

Format b	Internet part	Physical sub-network number	Device identification
----------	---------------	-----------------------------	-----------------------

Subnet(work) mask	Bits to 1	Bits to 0
-------------------	-----------	-----------

## Default Gateway

The default gateway parameter allows routing of messages to a device from a different network.

## TCP/IP Features

### Presentation

Ethernet communication is mainly responsible for the following applications:

- Coordination between PLCs
- Local or remote monitoring
- Communication with production management software
- Communication with remote I/O

The TCP/IP communication profile on Ethernet, supported by the TSX ETG 30\*\* module, allows communication via Modbus messaging

Acting as an agent, the TSX ETG 30\*\* module also supports management of the network monitoring standard SNMP.

### Communication Port

The communication port reserved for the TSX ETG 30\*\* module is Port 502 (reserved port for Schneider Electric). When a client device attempts to access the module, it requests that a connection be opened to this port.

### Timeout on TCP Connection

If a TCP connection is unable to be established (destination unavailable, for example), timeout is 80 seconds.

If the first exchange attempt was not successful, you can set the timeout for communication functions to a value greater than 80 seconds.

### Keep Alive Function

The Keep Alive function automatically generates a frame every 2 hours or so to check for broken connections.

## Default IP Address

### General

The TSX ETG 30•• module manages various IP addresses:

- one default IP address set in the factory
- one local IP address (*see page 121*) for the local Ethernet link (LAN) and configured by the user
- one IP address related to the modem connection (IP address used by the PPP protocol, or GPRS IP address provided by the GPRS provider)

#### NOTE:

- Use an orderly IP address assignment method so that duplicate address assignments do not occur.
- Always obtain your IP addresses from your system administrator to avoid the possibility of duplicate addresses.

### Default IP Address of the Ethernet Interface for the TSX ETG 30•• Module

The default IP address of the Ethernet interface for the TSX ETG 30•• module is derived from its MAC address:

10.10.xxx.yyy, where xxx and yyy are the last two numbers of the MAC address in decimal format.

#### Example:

The MAC address of the module (in hexadecimal format) is: 00 80 F4 01 **12 20**.

In this case, the default IP address (in decimal format) is: 10.10.**18.32**.

**NOTE:** The manufacturer defines the MAC address of the module (you cannot change this address). Its value appears in the front panel of the module.

### IP Address of the PPP Interface

The TSX ETG 30•• module manages one IP address per interface:

- the IP address of the Ethernet interface, configured by the user or pre-assigned (see above)
- the IP address of the PPP interface

The latter is assigned during connection negotiation by the PPP protocol. The TSX ETG 30•• module accepts any type of IP address during negotiation. Therefore any device that establishes a modem/PPP connection with the TSX ETG 30•• module should be configured to assign the IP address to the TSX ETG 30•• module.

However, if the remote device is configured to receive its IP address from the TSX ETG 30••, the IP addresses are:

- TSX ETG 30••: **10.10.0.4**
- remote device: **10.10.0.3**

If the connection is achieved by direct cable, the IP addresses are:

- TSX ETG 30\*\* : **10.10.0.2**
- remote device: **10.10.0.1**

**NOTE:** You must configure your computer (*see page 77*) in order to use the PPP interface.

### **GPRS IP Addresses Provided by the GPRS Service Provider**

The TSX ETG 30\*\* module can be configured using:

- GPRS static IP addresses
- GPRS dynamic IP addresses

The choice between static or dynamic IP addresses is defined by the GPRS contract (*see page 164*) established between you and your GPRS service provider. The dynamic GPRS address supports the DynDNS (*see page 167*) service.

---

## Opening a Connection on the Ethernet Network

### Presentation

A connection can be opened by a remote device wishing to communicate with the module in order to retrieve data via Modbus.

A connection is characterized by the module as follows:

Local TCP port, local IP address/remote TCP port, remote IP address.

The TSX ETG 30\*\* module is able to open a connection on request from a remote device using:

- TCP, or
- the modem.

#### **NOTE:**

- The maximum number of connections that can be open simultaneously is 64.
- Connection management is transparent for the user.

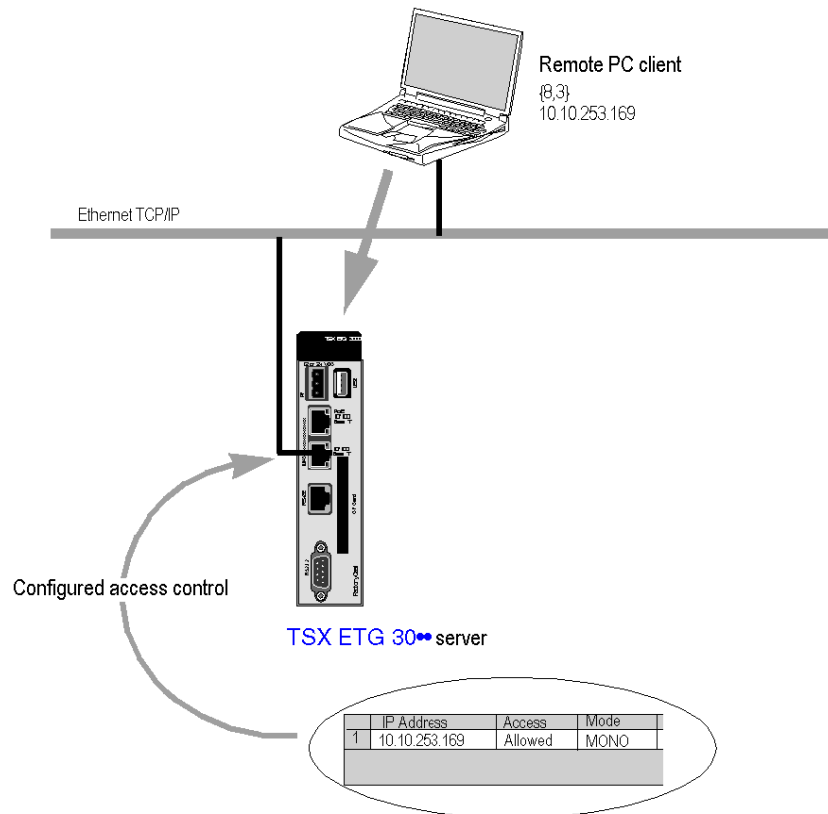
### Request from a Remote Device

When the TSX ETG 30\*\* receives a request from a remote device, the TSX ETG 30\*\* is the connection server.

On receiving a connection request from a remote device, the IP address of the remote device is verified (if access control is enabled).

The test checks that the address belongs to the list of remote devices authorized to connect. If it is, the module accepts the connection, otherwise it closes the connection.

The following illustration describes this connection process:

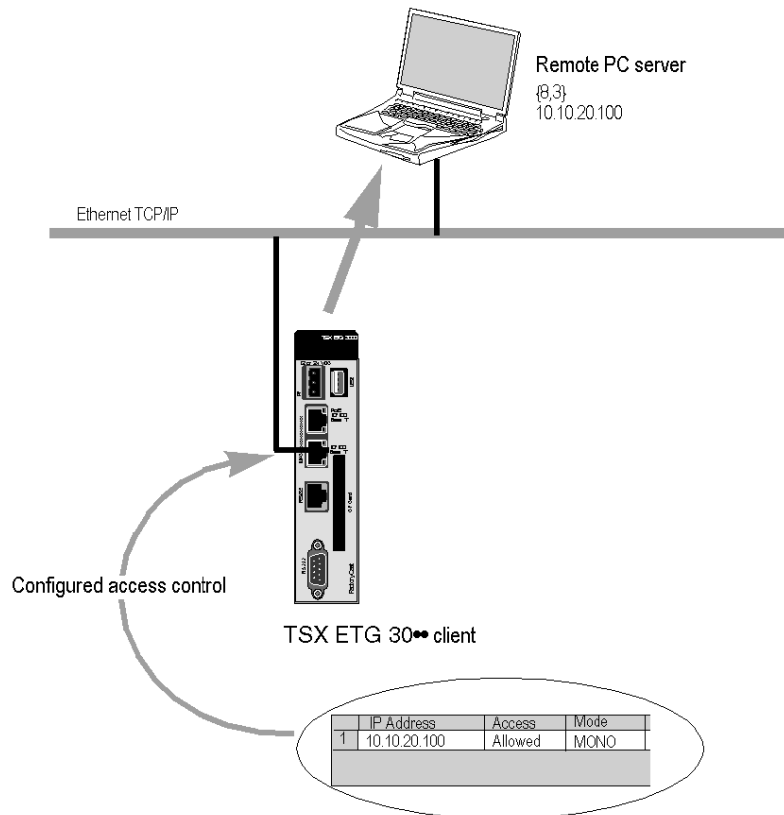


### Request to a Remote Device

When the TSX ETG 30 sends a request to a remote device, the TSX ETG 30 is the connection client (router function).

If there is no connection with the remote device, the TSX ETG 30 module automatically opens the TCP/IP connection to port 502 on the remote device. The remote device must belong to the IP configuration table.

The following illustration describes this connection process:



## Retrieving the IP Address

### Purpose

Use the direct cable feature to retrieve a lost IP address for the TSX ETG 30\*\* module. Configure your computer to communicate with the module via direct cable. Then, access the embedded website of the module to retrieve the IP address.

### Configuring your PC

For the first connection you must configure the components of your operating system (*see page 77*) to communicate with the TSX ETG 30\*\* module.

### Retrieving the IP Address

Once the preceding components have been configured, follow these steps:

Step	Action
1	Switch off the module.
2	Connect an RS 232 crossed cable between a COM port of the PC and the 9-pin SUB D socket of the TSX ETG 30**.
3	Start the module. The direct cable connection is established between the PC and the module (it takes about 40 seconds to be set up).
4	Establish the serial connection between the PC and the module.
5	Open a Web browser on the PC.
6	In the address bar, type: <code>http://10.10.0.2</code> , then click <b>Enter</b> . <b>Result:</b> the home page of the website appears.
7	Click <b>Setup</b> on the horizontal menu.
8	Click <b>IP Configuration</b> on the vertical menu. <b>Result:</b> the IP configuration page appears, the IP address of the module is displayed on this page.



## Configuration Parameters for TCP/IP Services

### Introduction

You can access the **IP Configuration** page via the **Setup** menu of the website. Having two devices with the same IP address can cause unpredictable operation of your network.

### WARNING

#### UNINTENDED OPERATION — DUPLICATE IP ADDRESS

- Obtain all IP addresses from your system administrator to avoid duplication.
- Assign a unique address to the device.

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

### View of the Ethernet Configuration Page

Ethernet parameters	
Ethernet frame format	Ethernet II
Link speed and duplex	100-FD-Auto
IP parameters	
<input type="radio"/> Served from MAC address	
<input type="radio"/> Served from device name	
<input checked="" type="radio"/> Local	
IP address	139 . 124 . 2 . 100
Subnet mask	255 . 255 . 255 . 0
Default gateway	139 . 124 . 2 . 1
FDR parameters	
Device name	<input type="text"/>
Replication period (mn)	<input type="text"/>
<input type="button" value="Apply"/> <input type="button" value="Undo"/>	

## Ethernet Parameters

The **Ethernet parameters** area describes the characteristics of the Ethernet communication:

- **Ethernet frame format** corresponds to the standard used for the Ethernet connection. Ethernet II is the most commonly used and complies to RFC 894 standard (this parameter cannot be changed).
- **Link speed and duplex** shows the characteristics of Ethernet connection. 100-FD-Auto means a connection speed up to 100 Mbits/s, Full Duplex with Automatic negotiation (these parameters cannot be changed).

## IP Parameters

The **IP parameters** area is used to define the IP address in three different ways:

- either the address is configured manually by clicking **Local**.
- or the address is provided by a DHCP server (*see page 228*) by clicking **Served from device name**, in which case the address input area is grayed out.
- or the address is deduced from the MAC address of the module by clicking on **Served from MAC address**, in which case the address input area is grayed out.

If you choose **Local**, you can then enter:

- the module IP address, **IP Address**,
- the subnetwork mask, **Subnet mask**,
- the address of the **Default Gateway**.

If you choose **Served from device name**, the IP address of the module is configured via a remote device acting as a DHCP (FDR) server. In this case, FDR parameters have to be specified.

## FDR Parameters

The **FDR parameters** area is used to configure an automatic connection:

- Enter the name of the module in the **Device Name** field.
- Enter the time (in seconds) for automatic comparison between the configuration of the TSX ETG 30•• and that stored on the DHCP server. If a difference is found, the TSX ETG 30•• automatically saves the configuration to the server.

### NOTE:

- If the DHCP (FDR) mode is selected, any modification of the configuration is automatically saved on the server.
- To switch to automatic mode when the module is configured for local mode, you must reinitialize the module with **Reboot**.

## IP Filtering Configuration

### Presentation

You can access the IP filtering configuration page via the **Setup** menu of the website. This function filters only the TCP Modbus communications on port 502 (not http, ftp,...).

### View of the IP Filtering Page

Access control

Connection configuration			
	IP address	Access	Mode
1	10.10.0.1	<input checked="" type="checkbox"/>	MULTI
2	10.10.0.30	<input type="checkbox"/>	MULTI
3	160.12.20.100	<input type="checkbox"/>	MONO
4		<input type="checkbox"/>	
5		<input type="checkbox"/>	
6		<input type="checkbox"/>	
7		<input type="checkbox"/>	
8		<input type="checkbox"/>	
9		<input type="checkbox"/>	
10		<input type="checkbox"/>	

Apply      Undo

## Parameters

General parameters:

Parameters	Value to be entered
<b>Access Control</b>	<p>This check box is used to enable or disable control of remote devices wishing to open a TCP connection to the module:</p> <ul style="list-style-type: none"> <li>• if the box is checked, access control management is activated and the <b>Access</b> column in the table is de-grayed (active). If the module operates in server mode, only remote devices selected by the <b>Access</b> box are authorized to connect as a client and to then communicate.</li> <li>• if the box is unchecked, control management is disabled and the <b>Access</b> column in the table is grayed out (not active). If the module is operating in server mode, third party remote devices can connect as client and then communicate with the module without being declared in the table.</li> </ul>

Entering data for remote devices connecting to the module:

Parameters	Values
<b>Input fields</b>	
<b>IP Address</b>	Enter the IP address of the device.
<b>Access</b>	If this box is checked, it indicates that the remote device is authorized to open a TCP connection.
<b>Mode</b>	<ul style="list-style-type: none"> <li>• <b>MONO</b>: The module only allows a single connection to a remote IP address.</li> <li>• <b>MULTI</b>: The module allows a single connection in client mode to a remote IP address and multiple connections in server mode to the same remote IP address.</li> </ul>
<b>Edit buttons</b>	
<b>Apply</b>	Confirm the input.
<b>Undo</b>	Cancel the current changes.

**NOTE:** If the module is used for routing, the Ethernet and modem IP addresses are identical and the connection screen applies to all connections (TCP/IP and PPP).

In server mode the module does not manage the password associated with the modem connection. All telephone connections are accepted. The **Access control** check box takes effect with an IP address for a remote device, as for the *PAP* (*Password Authentication Protocol*). The password must be managed at modem level.

---

## 4.6 SOAP / XML Communications

---

### Overview

This section describes the various SOAP / XML Communications implemented in TSX ETG 30•• Web servers. SOAP Web services are fully compliant with the W3C WS-I Web services standards. These services provide an open and standard communication means for control level devices to interact directly with information management applications using non proprietary SOAP protocol.

Web services are based on standards such as:

- SOAP (Simple Object Access Protocol), the exchange protocol carried out over the HTTP (HyperText Transfer Protocol) channel.
- WSDL (Web Services Description Language), in XML format.
- XML (eXtensible Markup Language), the universal data exchange standard.

TSX ETG 30•• SOAP Web services act as SOAP server interfaces. They allow developers to easily design client applications that can exchange data directly with TSX ETG 30•• Web servers. Applications such as Microsoft.NET, SQL Server, Microsoft Office, IBM (WebSphere), SUN (Java, Eclipse), Lotus, Oracle, SAP, MES, ERP and so forth can be interfaced directly with TSX ETG 30•• using SOAP Web services.

Two kinds of Web services are provided in TSX ETG 30•• modules as SOAP server interfaces:

- ModbusXMLDA: Web service to implement data access to Modbus variables,
- SymbolicXMLDA: Web service to implement Symbolic data access.

The Web services provided by TSX ETG 30•• are compatible with the WS-I basic profile 1.1.

**NOTE:** For more information on Web services, please refer to specialized publications on the subject.

## What's in this Section?

This section contains the following topics:

<b>Topic</b>	<b>Page</b>
Designing a SOAP Client Interface	127
ModBusXMLDA SOAP Services	128
SymbolicXMLDA SOAP Services	138
How to Optimize Massive Symbolic Requests	143
Visual Basic Examples	146
Online SOAP Documentation	147

---

## Designing a SOAP Client Interface

### Introduction

A server interface enables a SOAP client application to communicate directly with a TSX ETG 30•• Web server module. Exchanges are initiated by the SOAP client application. The TSX ETG 30•• server responds to these requests.

### Designing SOAP Client Interfaces

Stage	Action
1	Create the client application: The development environment (for example, Visual Studio.net) connects to a TSX ETG 30•• Web server module where it can access a list of available Web services. The Web server returns descriptions of the requested services as WSDL objects.
2	Develop the client application: The developer integrates the Web service APIs using the code retrieved in the previous stage as a Web reference and generates the client application.
3	Execute the client application: In run mode, the client application communicates in real time with the TSX ETG 30•• Web server module using the SOAP protocol.

## ModbusXMLDA SOAP Services

### Overview

The ModbusXMLDA Web Service is implemented to provide data access to Modbus variables using a syntax similar to the Modbus protocol.

This Web service offers the following APIs:

- ReadDeviceIdentification,
- ReadMultipleRegisters,
- WriteMultipleRegisters,
- ReadCoils,
- WriteMultipleCoils,
- ReadInt32,
- WriteInt32.

### Data amounts for API parameters

This table presents the maximum amount of data, according to the modules, that can be used for API parameters:

APIs	TSX WMY 100	140 NWM 100
ReadMultipleRegisters	124 (Register)	124 (Register)
WriteMultipleRegisters	119 (Register)	98 (Register)
ReadCoils	511 (Coils)	255 (Coils)
WriteMultipleCoils	510 (Coils)	254 (Coils)
ReadInt32	62 (Int32)	62 (Int32)
WriteInt32	59 (Int32)	49 (Int32)

### ReadDevice-Identification

The ReadDeviceIdentification request returns the entire device identification of the destination specified by the ID provided in the request.

### Parameters

Input: int UnitID

Output: string VendorName, ProductCode, MajorMinorRevision, VendorURL, ProductName, ModelName, UserApplicationName, TRImplementationClass, TRCommunicationServices



## Example

The following is an example of a SOAP request and its response:

### Request

```
POST /ws/ModbusXmlDa
HTTP/1.1 Host: 139.160.65.83:8080
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://www.schneider-
electric.com/ws/ModbusXmlDa/ReadDeviceIdentification"
<?xml version="1.0" encoding="utf-8" ?>
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ReadDeviceIdentification xmlns="http://www.schneider-
electric.com/ws/ModbusXmlDa/">
      <UnitID>int</UnitID>
    </ReadDeviceIdentification>
  </soap:Body>
</soap:Envelope>
```

### Response

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length <?xml version="1.0" encoding="utf-8"
?><soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ReadDeviceIdentificationResponse
xmlns="http://www.schneider-electric.com/ws/ModbusXmlDa/">
      <ReadDeviceIdentificationResult>
        <DeviceIdentification>
          <VendorName>string</VendorName>
          <ProductCode>string</ProductCode>
          <MajorMinorRevision>string</MajorMinorRevision>
          <VendorUrl>string</VendorUrl>
          <ProductName>string</ProductName>
          <ModelName>string</ModelName>
          <UserApplicationName>string</UserApplicationName>
```

```
        <TRImplementationClass>string</TRImplementa-
tionClass>
        <TRCommunicationServices>string</TRCommunication-
Services>
    </DeviceIdentification>
    </ReadDeviceIdentificationResult>
  </ReadDeviceIdentificationResponse>
</soap:Body>
</soap:Envelope>
```

## ReadMultiple-Registers

This is the ReadMultipleRegisters API used to implement a read multiple registers action.

The request provides three parameters:

- the destination of the read request,
- the starting address,
- the number of values to read (base index is provided by starting address).

In response, the list of values is provided.

## Parameters

Input: int UnitID, Address, Quantity

Output: array of int Result

## Example

The following is an example of a SOAP request and its response:

### Request

```
POST /ws/ModbusXmlDa
HTTP/1.1 Host: 139.160.65.83:8080
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://www.schneider-
electric.com/ws/ModbusXmlDa/ReadMultipleRegisters"
<?xml version="1.0" encoding="utf-8" ?>
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
```

```

    <ReadMultipleRegisters xmlns="http://www.schneider-
electric.com/ws/ModbusXmlDa/">
      <UnitID>int</UnitID>
      <Address>int</Address>
      <Quantity>int</Quantity>
    </ReadMultipleRegisters>
  </soap:Body>
</soap:Envelope>

```

### Response

```

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length <?xml version="1.0" encoding="utf-8" ?>
  <soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
    <soap:Body>
      <ReadMultipleRegistersResponse
xmlns="http://www.schneider-electric.com/ws/ModbusXmlDa/">
        <ReadMultipleRegistersResult>
          <int>int</int>
          <int>int</int>
        </ReadMultipleRegistersResult>
      </ReadMultipleRegistersResponse>
    </soap:Body>
  </soap:Envelope>

```

## WriteMultiple-Registers

This is the WriteMultipleRegisters API used to implement a write multiple registers action.

The request provides three parameters:

- the destination of the write request,
- the starting address,
- the values to write.

## Parameters

Input: int UnitID, Address, array of int Value

Output: none

## Example

### Request:

```
POST /ws/ModbusXmlDa
HTTP/1.1 Host: 139.160.65.83:8080
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://www.schneider-
electric.com/ws/ModbusXmlDa/WriteMultipleRegisters"
<?xml version="1.0" encoding="utf-8" ?>
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <WriteMultipleRegisters xmlns="http://www.schneider-
electric.com/ws/ModbusXmlDa/">
      <UnitID>int</UnitID>
      <Address>int</Address>
      <Value>
        <int>int</int>
        <int>int</int>
      </Value>
    </WriteMultipleRegisters>
  </soap:Body>
</soap:Envelope>
```

### Response

```
HTTP/1.1 200 OKContent-Type: text/xml; charset=utf-8
Content-Length: length <?xml version="1.0" encoding="utf-8" ?>
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <WriteMultipleRegistersResponse
xmlns="http://www.schneider-electric.com/ws/ModbusXmlDa/">
      </WriteMultipleRegistersResponse>
    </soap:Body>
  </soap:Envelope>
```

## ReadCoils

This is the ReadCoils API used to implement a read multiple coils action.

The request provides three parameters:

- the destination of the read request,
- the starting address,
- the number of values to read (base index is provided by starting address).

In response, the list of values is provided.

## Parameters

Input: int UnitID, Address, Quantity

Output: array of int Result

## Example

The following is an example of a SOAP request and its response:

### Request

```
POST /ws/ModbusXmlDa
HTTP/1.1 Host: 139.160.65.83:8080
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://www.schneider-
electric.com/ws/ModbusXmlDa/ReadCoils"
<?xml version="1.0" encoding="utf-8" ?>
  <soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ReadCoils xmlns="http://www.schneider-
electric.com/ws/ModbusXmlDa/">
      <UnitID>int</UnitID>
      <Address>int</Address>
      <Quantity>int</Quantity>
    </ReadCoils>
  </soap:Body>
</soap:Envelope>
```

### Response

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8" ?>
  <soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ReadCoilsResponse xmlns="http://www.schneider-
electric.com/ws/ModbusXmlDa/">
      <ReadCoilsResult>
        <int>int</int>
        <int>int</int>
      </ReadCoilsResult>
    </ReadCoilsResponse>
  </soap:Body>
</soap:Envelope>
```

## WriteMultipleCoils

The WriteMultipleCoils API implements a write multiple coils request.

The read request provides three parameters:

- the destination of the write request,
- the starting address,
- the values to write.

## Parameters

Input: int UnitID, Address, array of int Value

Output: none

## Example

### Request

```
POST /ws/ModbusXmlDa HTTP/1.1
Host: 139.160.65.83:8080
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://www.schneider-
electric.com/ws/ModbusXmlDa/WriteMultipleCoils"
<?xml version="1.0" encoding="utf-8" ?>
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <WriteMultipleCoils xmlns="http://www.schneider-
electric.com/ws/ModbusXmlDa/">
      <UnitID>int</UnitID>
      <Address>int</Address>
      <Value>
        <int>int</int>
        <int>int</int>
      </Value>
    </WriteMultipleCoils>
  </soap:Body>
</soap:Envelope>
```

### Response

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8" ?>
  <soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <WriteMultipleCoilsResponse xmlns="http://www.schneider-
electric.com/ws/ModbusXmlDa/">
      </WriteMultipleCoilsResponse>
    </soap:Body>
  </soap:Envelope>
```

### ReadInt32 SOAP request/response

The ReadInt32 request provides three parameters:

- The destination of the read request,
- the starting address,
- the number of values to read (base index is provided by starting address).

### Parameters

Input: int UnitID, Address, Quantity

Output: array of int Result

### Example

In response, the list of values is provided. The following is an example of a SOAP request and response:

#### Request

```
POST /ws/ModbusXmlDa
HTTP/1.1 Host: 139.160.65.83:8080
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://www.schneider-
electric.com/ws/ModbusXmlDa/ReadInt32"
<?xml version="1.0" encoding="utf-8" ?>
  <soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
```

```
<soap:Body>
  <ReadInt32 xmlns="http://www.schneider-
electric.com/ws/ModbusXmlDa/">
    <UnitID>int</UnitID>
    <Address>int</Address>
    <Quantity>int</Quantity>
  </ ReadInt32 >
</soap:Body>
</soap:Envelope>
```

### Response

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8" ?>
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    < ReadInt32Response xmlns="http://www.schneider-
electric.com/ws/ModbusXmlDa/">
      < ReadInt32Result>
        <int>int</int>
        <int>int</int>
      </ ReadInt32Result>
    </ ReadInt32Response>
  </soap:Body>
</soap:Envelope>
```

### WriteInt32 SOAP request/response

The WriteInt32 request provides three parameters:

- The destination of the write request,
- the starting address,
- the values to write.

### Parameters

Input: int UnitID, Address, array of int Value

Output: none



## Example

The following is an example of a SOAP request and response:

### Request

```
POST /ws/ModbusXmlDa
HTTP/1.1 Host: 139.160.65.83:8080
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://www.schneider-
electric.com/ws/ModbusXmlDa/WriteInt32"
<?xml version="1.0" encoding="utf-8" ?>
  <soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <WriteInt32 xmlns="http://www.schneider-
electric.com/ws/ModbusXmlDa/">
      <UnitID>int</UnitID>
      <Address>int</Address>
      <Value>
        <int>int</int>
        <int>int</int>
      </Value>
    </ WriteInt32 >
  </soap:Body> </soap:Envelope>
```

### Response

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8" ?>
  <soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    < WriteInt32Response xmlns="http://www.schneider-
electric.com/ws/ModbusXmlDa/">
      </ WriteInt32Response>
    </soap:Body>
  </soap:Envelope>
```

## SymbolicXMLDA SOAP Services

### Overview

The SymbolicXMLDA is the Web Service provided to implement a Symbolic Data Access based on a TSX ETG 30\*\* namespace. The services are:

- read,
- write,
- browse.

### Limitation on symbolic access

As symbolic access relies on low-level Modbus access for actual communication, only symbols mapped on data accessible through native Modbus requests (Read/Write registers and Read/Write Coils) are supported. Consequently, only data of these types are supported:

- bit
- 16-bit
- 32-bit double word

Unlocated data of all types are not supported.

### Read service

The aim of this request is to read one or several symbols. The symbols passed as request arguments are translated using the namespace to get the address of the value to read. When all addresses are known, a Modbus request is issued by contiguous addresses (i.e., if registers 10, 11, and 13 are requested, two requests are issued: one for 10 and 11, and one for 13) and by type of variable desired (coils, registers). Once all the values are retrieved, the SOAP response is built using the variable name, type and values.

### Parameters

Input: list of string ItemName

Output: array of Item ReadResult

## Example

### Request

```
POST /ws/SymbolicXmlDa HTTP/1.1
Host: 139.160.65.83
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://www.schneider-
electric.com/ws/SymbolicXmlDa/Read"
  <?xml version="1.0" encoding="utf-8" ?>
  <soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
    <soap:Body>
      <Read xmlns="http://www.schneider-
electric.com/ws/SymbolicXmlDa/">
        <Symbol>
          <Name>string</Name>
          <Name>string</Name>
        </Symbol>
      </Read>
    </soap:Body>
  </soap:Envelope>
```

### Response

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8" ?>
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ReadResponse xmlns="http://www.schneider-
electric.com/ws/SymbolicXmlDa/">
      <ReadResult>
        <Items>
          <Name>string</Name>
          <SymbolType>string</SymbolType>
          <IntValue>int</IntValue>
        </Items>
```

```
<Items>
  <Name>string</Name>
  <SymbolType>string</SymbolType>
  <IntValue>int</IntValue>
</Items>
</ReadResult>
</ReadResponse>
</soap:Body>
</soap:Envelope>
```

## Write service

The aim of this request is to write one or several symbols. The symbols, their types, and values are passed as request arguments. The names of the variables are translated using the namespace to get the address of the value to write. When all addresses are known, a Modbus request is issued by contiguous addresses (i.e., if register 10, 11, and 13 are requested, two requests are issued: one for 10 and 11, and one for 13) and by type of variable desired (coils, registers).

Once all the values are written, the SOAP response is built.

## Parameters

Input: list of (string ItemName, Value, Type)

Output: none

## Example

### Request

```
POST /ws/SymbolicXmlDa HTTP/1.1
Host: 139.160.65.83
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://www.schneider-
electric.com/ws/SymbolicXmlDa/Write"
<?xml version="1.0" encoding="utf-8" ?>
  <soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <Write xmlns="http://www.schneider-
electric.com/ws/SymbolicXmlDa/">
      <ItemList>
        <Items>
          <Name>string</Name>
          <SymbolType>string</SymbolType>
          <IntValue>int</IntValue>
```

```

        </Items>
    </Items>
    <Name>string</Name>
    <SymbolType>string</SymbolType>
    <IntValue>int</IntValue>
</Items>
</ItemList>
</Write>
</soap:Body>
</soap:Envelope>

```

### Response

```

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8" ?>
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
    <soap:Body>
        <WriteResponse xmlns="http://www.schneider-
electric.com/ws/SymbolicXmlDa/">
            </WriteResponse>
        </soap:Body>
    </soap:Envelope>

```

### Browse service

The aim of this request is to read a list of symbolic variables present on the namespace of the TSX ETG 30\*\* module.

This functionality does not interface with the Modbus layer since its only action is to return all the namespace items. The response contains all the symbols and their types.

### Parameters

Input: list of (string ItemName, Type)

Output: list of Descriptions BrowseResult

**Example****Request**

```
POST /ws/SymbolicXmlDa HTTP/1.1
Host: 139.160.65.83
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://www.schneider-
electric.com/ws/SymbolicXmlDa/Browse"
<?xml version="1.0" encoding="utf-8" ?>
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <Browse xmlns="http://www.schneider-
electric.com/ws/SymbolicXmlDa/">
      </Browse>
    </soap:Body>
  </soap:Envelope>
```

**Response**

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length
<?xml version="1.0" encoding="utf-8" ?>
  <soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <BrowseResponse xmlns="http://www.schneider-
electric.com/ws/SymbolicXmlDa/">
      <BrowseResult>
        <Descriptions>
          <Name>string</Name>
          <Address>string</Address>
          <SymbolType>string</SymbolType>
        </Descriptions>
        <Descriptions>
          <Name>string</Name>
          <Address>string</Address>
          <SymbolType>string</SymbolType>
        </Descriptions>
      </BrowseResult>
    </BrowseResponse>
  </soap:Body>
</soap:Envelope>
```

## How to Optimize Massive Symbolic Requests

### Overview

The symbolic read/write request allows you to perform requests of up to 128 symbols.

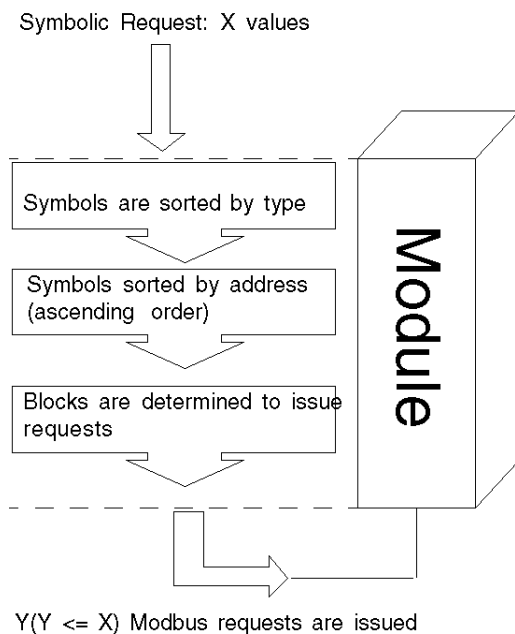
You can experience a lag when using a large amount of symbols. There are several rules you can apply to improve your application's speed, by reducing the number of internal requests.

The purpose of this topic is to help you create a coherent namespace and use symbolic requests efficiently.

### Symbolic Request Process

Symbolic requests enable the user to use symbols instead of addresses to handle TSX ETG 30 data. This request system uses the namespace on the module to resolve the addresses. Once the addresses are known, there is an optimization process to sort the addresses by type (coils, registers) in ascending order.

The sorted addresses should be grouped by coherent block to minimize the number of requests sent to the CPU. The graphic below shows the three steps of the symbolic request optimization:



## Optimization Method

To perform a massive symbolic request efficiently, it is necessary to pay close attention to the topology of the symbols:

Step	Action
1	Group symbols by type: do not mix coils and registers.
2	If you can, sort symbols by addresses.
3	Try to determine blocks of symbols (i.e., contiguous symbol addresses). These blocks will be processed in only one low-level request.

## Optimization Example

Considering a relatively simple namespace file:

```
|STATION:0.0
18
|Bit_mem1| %M0|0|0|1|1|1|false
|Bit_mem2| %M1|0|0|1|1|1|false
|Bit_mem3| %M2|0|0|1|1|1|false
|Bit_mem4| %M3|0|0|1|1|1|false
|Bit_mem5| %M4|0|0|1|1|1|false
|Bit_mem6| %M5|0|0|1|1|1|false
|Bit_mem7| %M6|0|0|1|1|1|false
|Engine_Brake_lvl| %MW0|0|0|4|1|1|false
|Engine_Brake_dsk| %MW1|0|0|4|1|1|false
|Engine_gauge_1| %MW2|0|0|4|1|1|false
|Engine_gauge_2| %MW3|0|0|4|1|1|false
|Engine_gauge_3| %MW4|0|0|4|1|1|false
|Engine_gauge_4| %MW5|0|0|4|1|1|false
|Engine2_Brake_lvl| %MW100|0|0|4|1|1|false
|Engine2_Brake_dsk| %MW101|0|0|4|1|1|false
|Engine2_gauge_1| %MW102|0|0|4|1|1|false
|Engine2_gauge_2| %MW103|0|0|4|1|1|false
|Engine2_gauge_3| %MW104|0|0|4|1|1|false
|Engine2_gauge_4| %MW105|0|0|4|1|1|false
0
1
false
```

There are values for two engines (engine and engine2).

There are also Boolean values (Bit\_mem).



If the end user asks for every symbol in their namespace order, they will issue 3 SOAP request with 18 values. As type and addresses (in ascending order) sort these values, the optimization will only determine blocks of addresses. Here we can see three blocks:

- coils from address 0 to 6,
- registers from address 0 to 5,
- registers from address 100 to 105.

There will be three low-level requests sent to the TSX ETG 30\*\*.

If the end user asks for five symbols in one SOAP request: Engine\_gauge\_3, Engine\_gauge\_1, Bit\_Mem1, Engine2\_gauge\_3, Engine2\_gauge\_1.

Step	Action				
1	Since symbols are mixed, the first optimization step is to sort between coils and registers.				
2	<p>In this step, symbols of the same type are sorted by address. As the SOAP request is not well ordered, this step has to process the data. At the end of this step we have two arrays: an array of one coil and an array of four registers:</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>Coils</th> <th>Registers</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">Bit_Mem1</td> <td style="text-align: center;">           Engine_gauge_1            Engine_gauge_3            Engine2_gauge_1            Engine2_gauge_3         </td> </tr> </tbody> </table>	Coils	Registers	Bit_Mem1	Engine_gauge_1 Engine_gauge_3 Engine2_gauge_1 Engine2_gauge_3
Coils	Registers				
Bit_Mem1	Engine_gauge_1 Engine_gauge_3 Engine2_gauge_1 Engine2_gauge_3				
3	<p>This step should find contiguous addresses. Since there are no contiguous addresses, we have five blocks of one address. The final result is that five low-level requests are sent to the CPU.</p>				

## In Brief

When you require intensive use (in number, in time constraints) of symbolic requests, these rules must be followed:

- group the symbols by types,
- sort the symbols by addresses,
- group the symbols by blocks of contiguous addresses.

## Visual Basic Examples

### Introduction

To help you start writing your application, here is a Visual Basic .NET (2003 or 2005) example of accessing SOAP requests.

### Preliminary Steps

Step	Action
1	In Visual Basic .NET, select <b>Project</b> → <b>Add WebReference</b> .
2	Type <code>http://Module @IP/ws/ModbusXmlDa</code> .
3	Select the desired function.
4	Click the <b>Add Reference</b> button.

### Example

The following example reads ten consecutive registers from register 5.

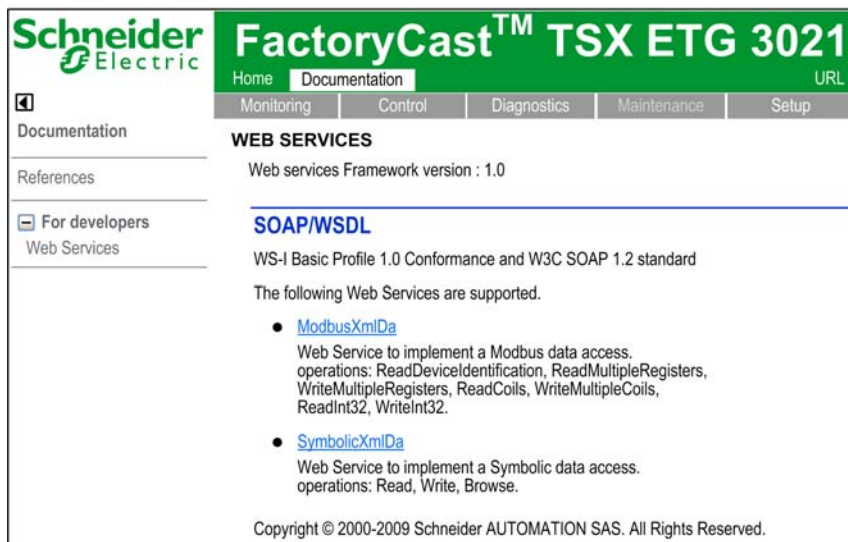
```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal  
e As System.EventArgs) Handles Button1.Click  
    Dim ws As New  
WindowsApplication26.WebReference.ModbusXmlDa  
    ListBox1.DataSource = ws.ReadMultipleRegisters(0, 5,  
10)  
End Sub
```

## Online SOAP Documentation

### Overview

When you open a TSX ETG 30\*\* Web server page in a browser, you can find a **Documentation** hyperlink.

This link takes you to SOAP online documentation, which describes the SOAP Web services available, explains the syntax of SOAP APIs and provides Visual Basic example.



The screenshot displays the Schneider Electric FactoryCast™ TSX ETG 3021 web interface. The top navigation bar includes links for Home, Documentation (selected), and URL. Below this, there are tabs for Monitoring, Control, Diagnostics, Maintenance, and Setup. The main content area is titled "WEB SERVICES" and indicates the Web services Framework version is 1.0. A section titled "SOAP/WSDL" states that the system conforms to WS-I Basic Profile 1.0 and W3C SOAP 1.2 standards. It lists two supported web services: ModbusXmlDa and SymbolicXmlDa, each with a brief description of their operations. The footer contains the copyright notice: Copyright © 2000-2009 Schneider AUTOMATION SAS. All Rights Reserved.

**Schneider Electric** **FactoryCast™ TSX ETG 3021**

Home **Documentation** URL

Monitoring Control Diagnostics Maintenance Setup

**WEB SERVICES**

Web services Framework version : 1.0

**SOAP/WSDL**

WS-I Basic Profile 1.0 Conformance and W3C SOAP 1.2 standard

The following Web Services are supported.

- [ModbusXmlDa](#)  
Web Service to implement a Modbus data access.  
operations: ReadDeviceIdentification, ReadMultipleRegisters, WriteMultipleRegisters, ReadCoils, WriteMultipleCoils, ReadInt32, WriteInt32.
- [SymbolicXmlDa](#)  
Web Service to implement a Symbolic data access.  
operations: Read, Write, Browse.

Copyright © 2000-2009 Schneider AUTOMATION SAS. All Rights Reserved.

## 4.7 PSTN, GSM and External Modems

---

### Overview

This section describes the embedded PSTN and GSM modems of the TSX ETG 30\*\* module. It also deals with external modems that can be connected to the module.

### What's in this Section?

This section contains the following topics:

Topic	Page
Opening a Serial Link Connection via a Modem	149
Modem PPP / Ethernet IP Addressing	153
Configuring the PSTN, GSM, or External Modem	157

---

## Opening a Serial Link Connection via a Modem

### Presentation

The internal modem is a specific feature of the TSX ETG 3010 (PSTN) and TSX ETG 302• (GSM). However, you can use the RS232 serial link to connect an external modem to all models.

You can open a serial link connection via a modem using one of the following methods:

- a request from a remote device (server mode)
- an email request from a module (client mode)
- an internal register command from a module (client mode)

You can disable the internal modem of the TSX ETG 3010 and TSX ETG 302• in order to use an external modem. Reserve this case for systems that require a specific modem.

**NOTE:** Client mode takes priority over server mode. If a remote device has established communication with the module in server mode, the connection will be closed by the TSX ETG 30•• if the module wishes to establish a connection to a remote device in client mode.

### PPP and PAP Protocols

The connection uses PPP. With this protocol, after a telephone connection has been established, the modem link is treated as a TCP/IP link at the application level.

A PPP connection uses PAP (*Password Authentication Protocol*) identification protocol for authentication. Any device that wants to communicate with the TSX ETG 30•• using the modem/PPP connection has to be configured using the PAP protocol.

**NOTE:** The TSX ETG 30•• does not support the CHAP (*Challenge-Handshake Authentication Protocol*) protocol.

If you want to open a connection to/from a remote device, you need to know the user name and the PAP password of the remote device. Before connecting the TSX ETG 30•• to the remote device, you must also configure the remote device to use the PAP protocol.

**NOTE:**

- If you use an external modem on the serial link, the modem must respond to AT commands in ASCII mode.
- The default user name and password for the PAP protocol are the same as those for the HTTP server (USER/USER).

### Diagnostics for the RS232 Modem Link

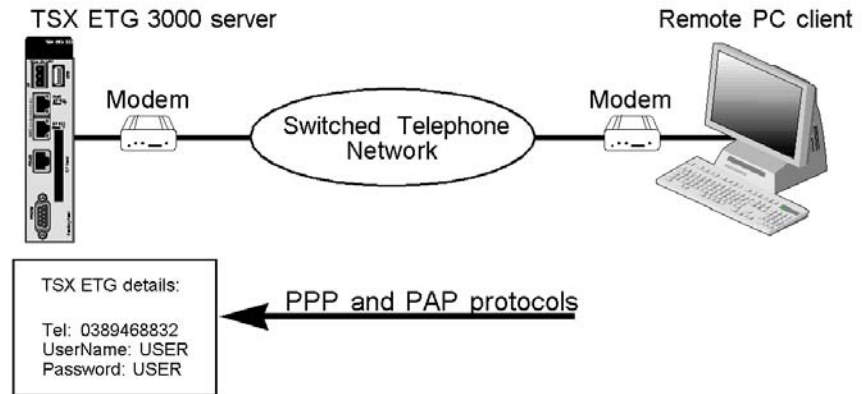
The modem/PPP log file page (*see page 370*) contains a log of the last 500 commands/diagnostics made.

### Connection by Request from a Remote Device

The TSX ETG 30\*\* module is the connection server. It waits for an incoming telephone connection request.

When a connection request arrives, the TSX ETG 30\*\* module verifies the user name and password (PAP) of the device. The TSX ETG 30\*\* module also checks that the IP address of the remote device appears in the list of remote devices authorized to connect (if access control has been activated). If identification is unsuccessful the module closes the connection.

The following illustration shows a connection by request from a remote device:



**NOTE:** The previous illustration shows the connection using a TSX ETG 3000 module. For the TSX ETG 3010, TSX ETG 302• products, the modem connected to the TSX ETG 3000 is optional (use of the internal modem). If you are using the TSX ETG 302• module, the communication goes through a GSM network instead of the switched telephone network.

### Connection by Request to a Remote Device

In this case, the TSX ETG 30\*\* is the connection client.

Two types of modem connection are available:

- The SMTP mail server is configured for modem connection; the module automatically opens the connection when an email is sent.
- On register command arrival (*see page 103*) from a module, the TSX ETG 30\*\* opens a connection.

**NOTE:** The TSX ETG 30\*\* Services (see page 257) take priority over the command:

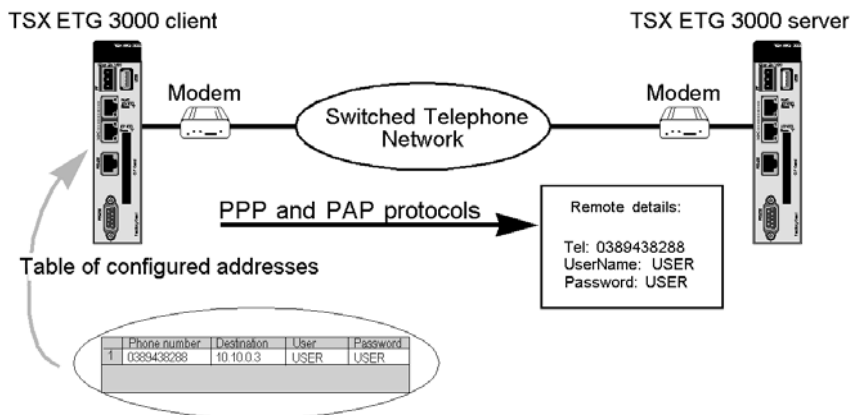
- If the IP address is not the same, the email connection closes the current command. In other words, the current connection is cut in order to send the email, and then the user needs to reestablish the previous connection.
- Otherwise, the command waits for the end of the Service connection (during the wait, the value of the register status is -1).

**NOTE:** If the user does not detect a dial tone and cannot dial a telephone number with internal mode (for example internal office line telephone, you can use Hayes command (AT X3: do not look for dialing tone, line busy, and dial). Refer to your modem documentation about Hayes commands supported.

The following table details the steps involved in the connection process:

Step	Action
1	The module dials the telephone number configured for this remote device.
2	The TSX ETG 30** automatically opens a TCP/IP connection to port 502 with the remote device.
3	The module checks if the remote device appears in the IP configuration table with its telephone number, name, and password.
4	The serial link connection is opened.

The following illustration describes the connection process:



**NOTE:**

- The previous illustration shows the connection between 2 TSX ETG 3000 modules. For the TSX ETG 3010, TSX ETG 302• products, the modem connected to the TSX ETG 3000 is optional (use of the internal modem). If you are using a TSX ETG 302• module, the communication goes through a GSM network instead of the switched telephone network.
- You can modify the telephone number for a remote device by clicking the **Setup** → **Phone List** link of the website or using Web Designer for TSX ETG 30••. You must restart the TSX ETG 30•• using the **Reboot** function for the new configuration to take effect.

**PPP Connection Priorities**

A remote device can open a modem (PPP) connection (server connection). The TSX ETG 30•• is also able to open a modem connection using the register commands (client connection). This page shows you the behavior of the module when it receives connection requests at the same time.

	<b>Existing Connection:</b>			
	<b>None</b>	<b>Remote device Server</b>	<b>Gateway - Command Client</b>	<b>Gateway Email Client</b>
New connection: Remote device server	<b>Server Connection</b> is opened.	The line is busy. <b>Impossible</b> to open a new connection	The line is busy. <b>Impossible</b> to open a new connection	The line is busy. <b>Impossible</b> to open a new connection
New connection: Gateway - Command client	<b>Client Connection</b> is opened.	The telephone connection is dropped. A <b>new connection</b> is opened.	The previous connection has not been closed. The telephone connection is dropped. A <b>new connection</b> is opened.	The new connection is put in a queue until the end of the previous PPP connection. Then the telephone connection is dropped. A <b>new connection</b> is opened.
New connection: Gateway -email client	<b>Client Connection</b> is opened.	The telephone connection is dropped. A <b>new connection</b> is opened.	The telephone connection is dropped. A <b>new connection</b> is opened.	The telephone connection is dropped if the previous PPP connection is closed. A <b>new connection</b> is opened.

**NOTE:** The telephone connection will be dropped if the IP address of the device is different from the currently connected device (client-client case).



## Modem PPP / Ethernet IP Addressing

### Presentation

The modem connection can use either its own IP address or the IP address of the Ethernet link. This page shows you how addressing your network devices in each case.

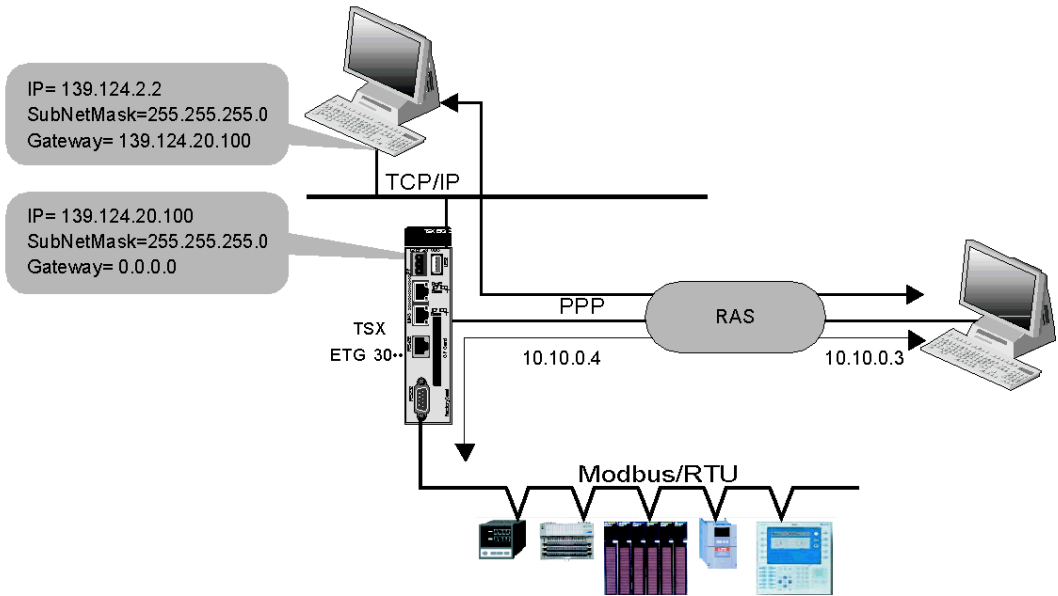
### Without the Ethernet IP Configuration

When connecting without using the Ethernet IP configuration, the TSX ETG 30\*\* uses 2 different IP addresses, one for the Ethernet link and another one for the modem link.

Stage	Description
1	The TSX ETG 30** uses the default PPP address (10.10.0.4) and another one for the Ethernet network (example: 139.124.20.100).
2	The gateway address of the TSX ETG 30** is set to 0.0.0.0 (direct link PPP/TCP-IP).
3	The device over PPP uses the default PPP address: 10.10.0.3.
4	The gateway address of the device over the Ethernet network is set to the value of the TSX ETG 30** IP address.

**NOTE:** Messages will be routed over the serial line if port 502 is used for incoming messages.

The following illustration shows how addressing when an Ethernet IP configuration is not used:



**NOTE:**

- The TSX ETG 30•• can access the Modbus devices with the IP address = 10.10.0.4 or 139.124.20.100 and with the UnitID = Modbus slave address.
- A remote device can access devices located on the same Ethernet segment as the TSX ETG 30•• through a PPP connection. In this case, you must set a gateway address of the remote device to the value of the TSX ETG 30•• address.

**Using the Ethernet IP Configuration**

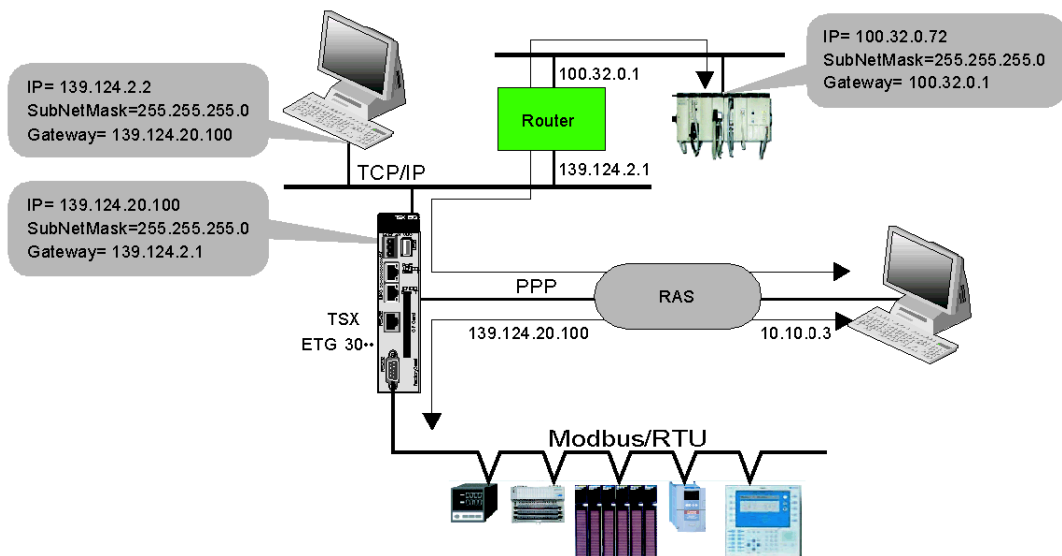
With an Ethernet IP configuration, the TSX ETG 30•• uses the same IP addresses for the Ethernet link and the modem link.

Once configured, a client can communicate over the modem line with devices on the serial line network or on the Ethernet network.

Step	Description
1	Set the gateway address to the value of the router address.
2	The gateway address of clients on the same Ethernet segment as the TSX ETG 30•• is set to the value of the TSX ETG 30•• IP address.
3	The gateway address of clients on a different segment than the one of the TSX ETG 30•• is set to the value of the router address.

**NOTE:** If there is only one Ethernet network (no router), the gateway address of the TSX ETG 30•• is set to 0.0.0.0.

The following illustration shows how addressing using an Ethernet IP configuration:



**NOTE:** A remote device can access devices located on the same Ethernet segment as the TSX ETG 30•• through a PPP connection. In this case, you must set a gateway address of the remote device to the value of the TSX ETG 30•• address.

## Served Addressing

The PPP address of the TSX ETG 30•• can be served. In this case, the remote device sets the PPP address of the TSX ETG 30•• during the connection negotiation.

## Address Management

The TSX ETG 30•• module can be:

- a PPP server: the IP address for the PPP link can be either fixed (10.10.0.4) or the IP address of the Ethernet link.
- a PPP client: the IP address for the PPP link can be fixed (10.10.0.3), served (the remote device set the IP/PPP address of the TSX ETG 30••) or the IP address of the Ethernet link.

## Routing Ethernet to Modbus

The TSX ETG 30•• transmits data traffic to the Modbus serial network if the destination address is the same than the TSX ETG 30•• address.

### **Routing Ethernet to PPP**

A remote device can make a connection request to a device through the TSX ETG 30•• and the Internet/PPP connection. In this case, the TSX ETG 30•• routes Ethernet to PPP.

## Configuring the PSTN, GSM, or External Modem

### Accessing the Configuration Page

You can access the Modem services configuration page via the Setup menu of the website.

### View of the Modem Configuration Page

The modem configuration page is different depending on the TSX ETG 30\*\* model and the configuration you are using. The following figures show the configuration page for each model and configuration.

Modem configuration page for TSX ETG 30\*\* modules using the external modem:

only for TSX ETG 3010 and TSX ETG 3021/3022

Modem	
<input type="radio"/> Internal	<input checked="" type="radio"/> External

Server: local PPP Address	
<input checked="" type="radio"/> Fixed	<input type="radio"/> TCP/IP address

External Modem	
Hayes Commands	AT
Time before connection (s)	1
Time detection	800
Retry	1
RS232 baud rate	57600
RS232 parity	NONE
RS232 stop bits	1

Line Management	
Idle time before hanging up	Never
Max time before hanging up	Never
Hang-up after email	<input type="checkbox"/>

Apply    Undo

Modem configuration page for a TSX ETG 3010 using the internal (PSTN) modem:

Modem	
<input checked="" type="radio"/> Internal	<input type="radio"/> External
Internal Modem	
Country	Default
Hayes command	
Server: local PPP Address	
<input checked="" type="radio"/> Fixed	<input type="radio"/> Use TCP/IP address
Callback	
Callback on ring Enable	<input checked="" type="checkbox"/>
Number of rings	3
Time before calling back (sec)	0
Phone list index	3
Line Management	
Idle time before hanging up	1mn
Max time before hanging up	Never
Hang-up after email	<input checked="" type="checkbox"/>
<input type="button" value="Apply"/> <input type="button" value="Undo"/>	

Modem configuration page for a TSX ETG 302• using the internal (GSM) modem:

Modem	
<input checked="" type="radio"/> Internal	<input type="radio"/> External
Server: local PPP Address	
<input checked="" type="radio"/> Fixed	<input type="radio"/> TCP/IP address
Internal Modem	
PIN code	*****
SMS Service Center	+447785020100
Hayes Commands	
Callback	
Callback on ring Enable	<input checked="" type="checkbox"/>
Number of rings	10
Time before calling back (sec)	0
Phone list index	1
Line Management	
Idle time before hanging up	1mn
Max time before hanging up	Never
Hang-up after email	<input checked="" type="checkbox"/>
<input type="button" value="Apply"/> <input type="button" value="Undo"/>	

GPRS	
GPRS enable	<input type="checkbox"/>
Connection	
Access Point Name (APN)	
Username	
Password	
IP publication	
DynDNS Hostname	
DynDNS Username	
DynDNS Password	

## General Parameters

Parameter	Description
Modem	
Internal	The embedded modem is used for the modem connection.
External	An external modem is used for the modem connection.
Server: local PPP Address	
Fixed	The default address is used (10.10.0.4).
Use TCP/IP address	The IP address of the modem interface is the same than that of the Ethernet interface.
Line Management	
Idle time before hanging up	If the time between two frames exceeds the value entered (in seconds) in this field, the connection is broken. If the time entered is Never, no control is performed.
Max time before hanging up	This field is used to define the maximum connection time (in seconds). At the end of this time the connection is broken. If the time entered is Never, no control is performed.
Hang-up after email	Select this box to close the PPP connection after an email has been sent. Use this feature in case the email service opens the connection.

## External Modem Parameters

Parameter	Description
Hayes Commands	Configuration by Hayes character string. You can use it to send commands to the modem in Hayes format*.
Time before connection	Specify the time before a connection is opened.
Time detection	Time before the modem sends an AT command.
Retry	Number of times the modem tries to open a connection.
RS232 Baud Rate	Speed of the RS232 link; the choice depends on the modem you are using: 2400, 4800, 9600, 19200, 38400 and 57400.
RS232 Parity	Even, odd or no parity.
RS232 Stop Bit	Number of stop bits.

\* Refer to your modem documentation to find out which Hayes commands are supported.

Example: AT&FS0=2

**Internal Modem Parameters**

Parameter	Description
Country	Select the country in which the module is used (only for PSTN modem).
SIM code	Enter the PIN code for the SIM card (only for GSM).
SMS service center	Enter the number of the SMS service center (only for GSM).
Hayes Commands	Configuration by Hayes character string. You can use it to send commands to the internal modem.

**NOTE:** Type the PIN code of the SIM card carefully: if you supply an incorrect PIN code 3 times, the SIM card is locked. If this happens, call your telephone operator to unlock it.

**Callback Parameters**

Parameter	Value
Callback on ring enable	Select this box to use the callback on ring service.
Number of rings	Specify the number of rings the module waits before launching the connection. Default: 2; minimum: 2; maximum: 20.
Time before calling back (sec.)	Specify the time to wait between the ring and the call back. Default: 0; minimum: 0; maximum: 240.
Phone list index	Phone list index of the number to call back (not used for GPRS). Default: 1; minimum: 1; maximum: 64.



---

## 4.8 GPRS Modem Communications

---

### Overview

This section describes the GPRS communications of the TSX ETG 30•• module.

### What's in this Section?

This section contains the following topics:

Topic	Page
GPRS Communications	162
Configuring the GPRS Modem	165
Opening an Internet Connection via a GPRS Modem	169
PPP Security: IP Control Configuration Page	174

## GPRS Communications

### Presentation

GPRS (*General Packet Radio Service*) communication is available on TSX ETG 302• modules (V1.1 version).

GPRS provides a cost effective solution for wireless remote connection to distributed installations over the Internet.

GPRS is a packet oriented data service based on GSM (*Global System for Mobile*). It offers the advantages to pay only for the total volume of data exchanged (in MB per month) regardless of the connection time, while data communication via traditional circuit switching (PSTN/GSM) is charged per minutes of connection time.

GSM connections are mainly used for on demand services such as sending SMS alarms or basic remote services such as diagnostics.

GPRS is more suitable for permanent access to remote installations providing:

- easy remote programming
- continuous remote monitoring and control
- transparent routing capabilities from the Internet to LAN networks or serial network devices connected to the TSX ETG 302• gateway

In addition, GPRS provides higher data exchange rates than GSM:

	Upload	Download
Theoretical	24 kbps	48 kbps
Typical	16 kbps	20 kbps

**NOTE:** These values depends on your service provider, the distance between your module and the base station, and the current traffic.

**NOTE:** If too many browsers are used with modem connection (GPRS, PSTN), performances are decreasing and could lead to a refresh page difficulties.

### GPRS Remote Access

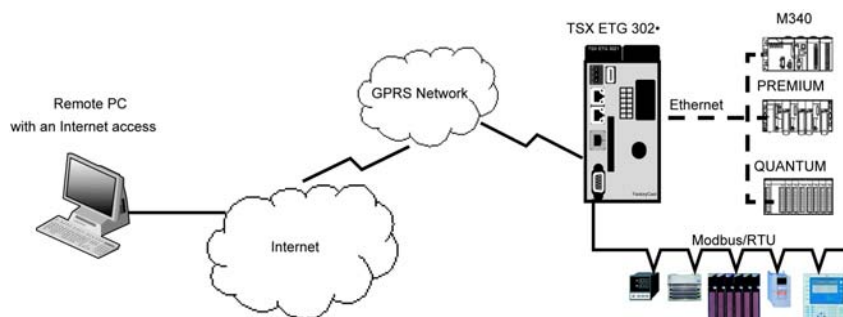
GPRS communication implies:

- The TSX ETG 302• module is connected to the Internet via the GPRS network.
- The remote PC or network is also connected to the Internet.

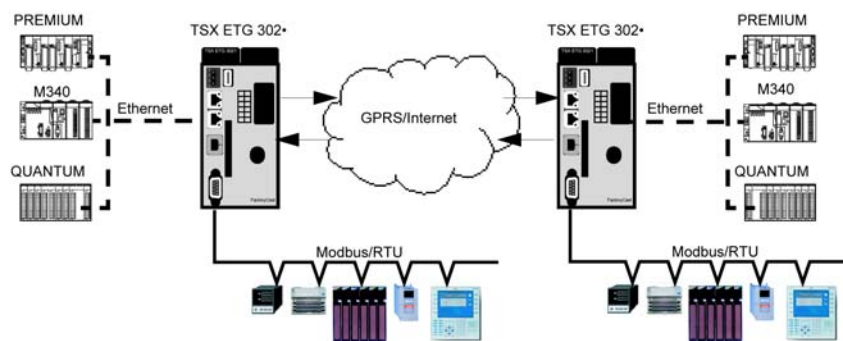
GPRS topologies can support:

- NAT (*see page 177*) routing tables for transparent routing to Ethernet devices
- security services such as IP address control (*see page 174*) or VPN tunnels (*see page 199*) for secured data exchange over the Internet

The following figure shows remote access to the network of the module:



The following figure shows inter-site communication by connecting 2 remote modules:



## Connection Principles

GPRS communication requires a SIM card and a specific GPRS contract (see page 164) with a service provider.

The GPRS connection is always initiated from the module to the GPRS network.

It is not possible for a client application to open a connection by dialing directly the TSX ETG 302\* module. Nevertheless, the module provides various solutions (see page 169) to connect to the GPRS network:

### Permanent Mode:

- Automatic connection at startup or reboot or after connection loss.

### On demand Mode:

- Callback function: opens the connection on reception of an incoming GSM or PSTN call.
- Autonomously on a process or application condition.

The module connects the APN (*Access Point Name*) of the service provider and receives an IP address back that can be static or dynamic.

The module supports both static and dynamic IP address. If the address is dynamic, it is necessary to inform the remote application of the new IP address (*see page 201*).

**NOTE:**

- GPRS uses the DNS server of the service provider, it replaces the DNS server configured in the module.
- The NTP service of the module does not work with GPRS.
- The default gateway set in the Ethernet configuration page (*see page 121*) of the module is not used with a GPRS connection. The default route of the GPRS connection is used instead. Thus, it is not possible to route through Ethernet when the module is connected to the GPRS network.

## **GPRS Contracts**

GPRS service providers offer dedicated services adapted to industrial applications, also called M2M (*Machine to Machine*).

Service providers offer GPRS contracts with different options. The most important options are:

- **public or private IP address:** You **must** choose a contract that gives you a public IP address to be reachable directly from the Internet.
- static or dynamic IP address
- incoming TCP ports blocked or not: Some providers offer only subscriptions with TCP ports blocked for security reasons, for instance, ports lower than 1024.

**NOTE:**

- For ease of use and configuration, you should choose a contract with no TCP ports blocked and with a static IP address.
- If your service provider blocks the public ports (< 1024), you must use a VPN and choose a contract that authorizes VPN traffic.

## Configuring the GPRS Modem

### Accessing the Modem Configuration Page

You can access the Modem services configuration page via the Setup menu of the module website.

**NOTE:** You can also configure the GPRS modem parameters using Web Designer.

### View of the Modem Configuration Page

The screenshot displays the Modem Configuration Page with the following sections and settings:

- Modem:** Radio buttons for  Internal and  External.
- Server Connection : local PPP Address:** Radio buttons for  Fixed and  Use TCP/IP address.
- Internal Modem:**
  - PIN code: \*\*\*\*\*
  - SMS Service Center: +447785020100
  - Hayes Commands: (empty field)
- Callback:**
  - Callback on ring Enable:
  - Number of rings: 10
  - Time before calling back (sec): 0
  - Phone list index: (empty field)
- Line Management:**
  - Idle time before hanging up: 1mn
  - Max time before hanging up: Never
  - Hang-up after email:
- GPRS:**
  - GPRS enable:
  - Connection: On demand
  - Access Point Name (APN): internet-company
  - Username: orange
  - Password: \*\*\*\*\*
- IP publication:**
  - DynDNS Hostname: etg3.dyndns.org
  - DynDNS Username: etg3021
  - DynDNS Password: \*\*\*\*\*

Buttons for **Apply** and **Undo** are located at the bottom of the form.

### General Parameters

Parameter	Description
Modem	
Internal	The embedded modem is used for the modem connection.
External	An external modem is used for the modem connection.
Line Management	
Idle time before hanging up	If the time between two frames exceeds the value entered (in seconds) in this field, the connection is broken. If the time entered is Never, no control is performed.
Max time before hanging up	This field is used to define the maximum connection time (in seconds). At the end of this time the connection is broken. If the time entered is Never, no control is performed.
Hang-up after email	Select this box to close the PPP connection after an email has been sent.

## Internal Modem Parameters

Parameter	Description
SIM code	Enter the PIN code for the SIM card.
SMS service center	Enter the number of the SMS service center.
Hayes Commands	Configuration by Hayes character string. You can use it to send commands to the internal modem.

**NOTE:** Type the PIN code of the SIM card carefully: if you supply an incorrect PIN code 3 times, the SIM card is locked. If this happens, call your service provider to unlock it.

## Callback Parameters

Parameter	Description
Callback on ring enable	Select this box to use callback on ring.
Number of rings	Specify the number of rings the module waits before launching the connection. Default: 2; minimum: 2; maximum: 20.
Time before calling back (sec)	Specify the time to wait between the ring and the callback. Default: 0; minimum: 0; maximum: 240.

## GPRS Parameters

Parameter	Description
GPRS Enable	Select this box to use the GPRS service.
Connection	Select the operating mode of the connection: <ul style="list-style-type: none"> <li>● Permanent: the module is always connected. Automatic connection at boot and after the loss of a connection.</li> <li>● On demand: the module is connected only on a user action (call back on ring, Modbus register command, or email).</li> </ul>
Access Point Name (APN)	Name of the access point given by the service provider for GPRS. The APN is mandatory when using GPRS. <b>Note:</b> Depending on your service provider, you might have to add an AT command. Add it in the APN field after the APN using “;” as separator (i.e. Internet-entreprise;;AT=CGQREQ=1,3,4,3,0,0) The module sends this command and checks the response before launching the connection.
Username	Username of the APN given by the service provider (can be empty).
Password	Password of the APN given by the service provider (can be empty).

## IP Publication

Parameter	Description
DynDNS Hostname	Hostname (URL) corresponding to the IP address of the module. The module informs the DynDNS server of its IP address at connection start. The server translates the hostname into the corresponding IP address. If you do not use DynDNS, keep this field empty.
DynDNS Username	Username to access the DynDNS service.
DynDNS Password	Password to access the DynDNS service.

## Working with Dynamic GPRS IP Addresses

If you have a dynamic IP address with your GPRS contract, this means that this IP address is renewed periodically by the service provider.

TSX ETG 302• modules provides 2 solutions to work with dynamic IP addresses:

- Use DynDNS (*Dynamic DNS*) service to work with an URL (i.e etg1.dyndns.org) that remains the same instead of the dynamic IP address.
- Use the email service to publish the new IP address to the end-user.

## IP Publication Using Emails

The module can send an email with its new IP address at GPRS connection start-up. Create an email service (*see page 279*) using Web Designer. The module only uses the GPRS\_STATUS register to trigger the email.

The following figure shows an example of IP publication using emails:

SendSMS

E-mail description


Identifier  Trigger  ... Type

Destination  Subject

Contents

My new IP address is:

```
http://{device.Device1.GPRS_IP1_ADD_ETG}.{device.Device1.GPRS_IP2_ADD_ETG}.
{device.Device1.GPRS_IP3_ADD_ETG}.{device.Device1.GPRS_IP4_ADD_ETG}
```

 Media  Path

Identifier	Destination	Subject	Trigger	Type	
email IP ...	remoteUser...	IP adresse...	email IP ...	RE	

## IP Publication Using DynDNS

TSX ETG 302• modules provides a solution to work with dynamic IP addresses through URL (DNS name). TSX ETG 302• modules support DynDNS name service supplied by DynDNS company. ([www.dyndns.com](http://www.dyndns.com)).

The DynDNS service allows a domain name to be mapped with IP addresses. When the GPRS provider changes the IP address of the module, the module sends queries to the DynDNS server to update its mapping table with the new address.

Create a DynDNS account by visiting the <http://www.dyndns.com> website. Then, set the URL you desire to use (i.e [etg1.dyndns.org](http://etg1.dyndns.org)) as a DNS name for your TSX ETG 302• module, and select the 'refreshed every minute' option. A username and a password are associated to the account. You can use them to protect access to the DynDNS service.

### NOTE:

- It is recommended to purchase an account that does not require reconfiguration after a period of inactivity.
- DynDNS updates the IP addresses every minutes, wait one minute before accessing the gateway.



## Opening an Internet Connection via a GPRS Modem

### Opening a GPRS Connection via the Integrated GPRS Modem

The GPRS service provides 2 modes to open a GPRS connection:

- **Permanent:** the module is always connected. Automatic connection (*see page 169*) at bootup and after a connection loss.
- **On demand:** the module only connects to GPRS network on demand by user actions or process event such as:
  - Connection/disconnection (*see page 170*) by a command register variable.
  - Callback on ring (*see page 171*): the module opens a GPRS connection after receiving a GSM call.
  - Sending an email request (*see page 172*): the module automatically opens a GPRS connection on sending an email.

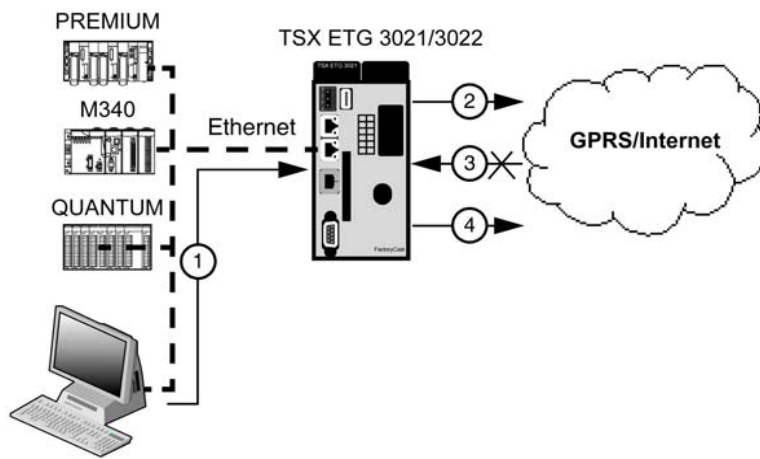
### Opening GPRS Connection on Remote Sites (Remote TSX ETG 302•)

This function allows a local TSX ETG 302• to request a remote TSX ETG 302• module to connect to the GPRS network for inter-site communication.

The local TSX ETG 302• calls the remote TSX ETG 302• in GSM mode, using its GSM number. The remote TSX ETG 302• hangs up (callback function enabled) and connects itself to the GPRS network. Then, the local TSX ETG 302• connects to the GPRS network and the communication is initiated. For more information, refer to Remote TSX ETG 302• GPRS Connection (*see page 173*).

### Permanent Connection

The module is always connected to the GPRS network. It connects to the GPRS network on reboot of the module or if the connection is lost:



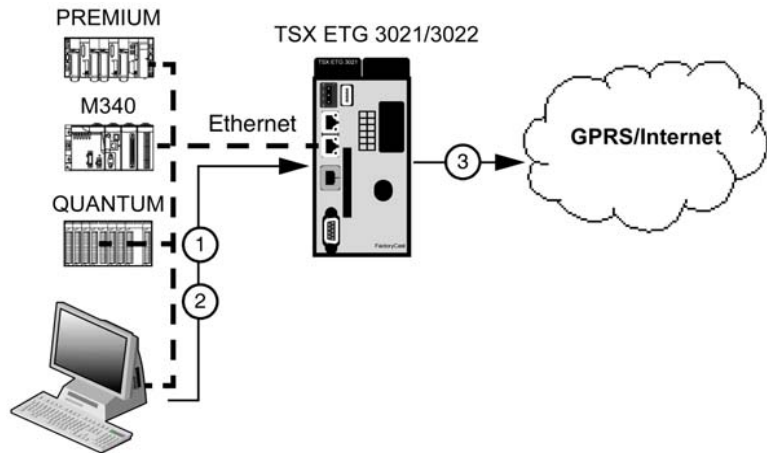
Step	Description
1	Configure a Permanent GPRS connection.
2	The module connects to the GPRS network.
3	The connection is lost or closed.
4	The module reconnects automatically to the GPRS network.

**NOTE:**

- If the module cannot connect to the GPRS network in the permanent mode, after 5 minutes a new connection attempt is made automatically.
- If you have a dynamic IP address, your service provider may close the connection. The module reconnects automatically but with a new IP address. You can avoid this disconnection by using the Line Management feature.

**Connecting by Registers Command**

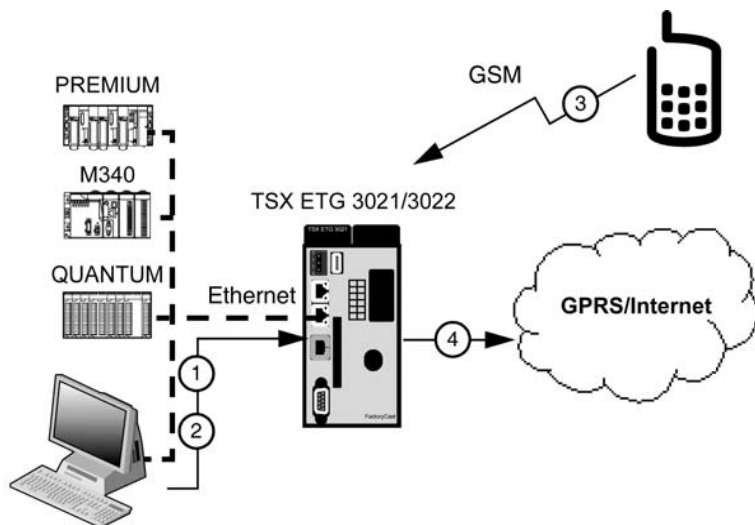
Write a specific register command to open the GPRS connection:



Step	Description
1	Configure an On Demand GPRS connection.
2	Write 1 to the COMMAND register (1001) and wait a few seconds (until the value of the STATUS register (1002) changes to 0). <b>Note:</b> GPRS does not use the CMD_PARAM1 register (1002).
3	The module connects to the GPRS network.

## Callback on Ring

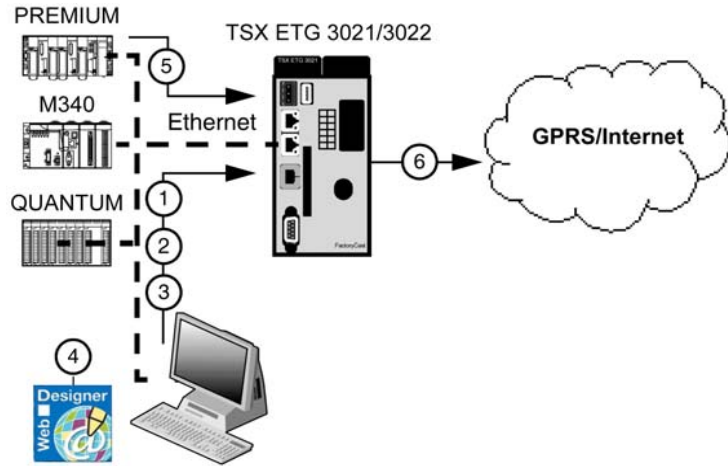
The module waits for an incoming telephone call to connect the GPRS network. When it receives a call, it hangs up after a number of rings and connects to GPRS network.



Step	Description
1	Configure an <b>On Demand</b> GPRS connection.
2	Select <b>Callback on ring Enable</b> in the callback panel and fill in: <ul style="list-style-type: none"> <li>the <b>Number of rings</b> field (i.e. 5, the module connects the GPRS network after 5 rings)</li> <li>the <b>Time before calling back (sec.)</b> field (i.e. 5, the module connects the GPRS network 5 seconds after the number of rings set previously)</li> </ul>
3	Call the module number (provided with the SIM card) using a landline or mobile.
4	After the number of rings and the delay configured in step 2, the module connects the GPRS network and hangs up the telephone call.

### Connecting to Send an Email

The module connects automatically to the GPRS network to send an email. GPRS network disconnection is possible after the email has been sent.

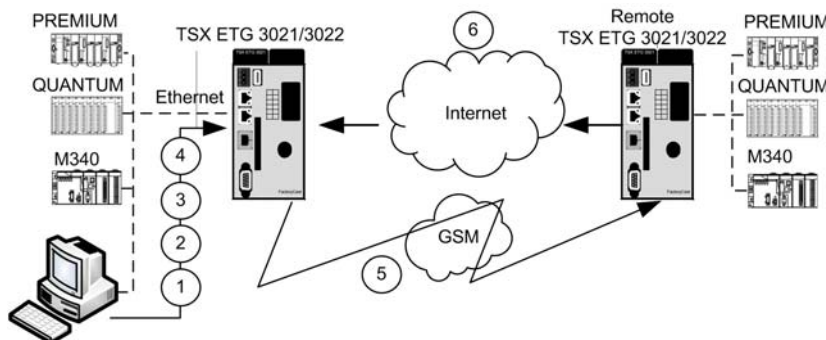


Step	Description								
1	Configure an <b>On Demand</b> GPRS connection. <b>Note:</b> Select the <b>Hang-up after Email</b> box if you want the connection to be closed after the email has been sent.								
2	Access the telephone list configuration page: click <b>Setup</b> → <b>Modem - Phone list</b> .								
3	Create a new record with the following parameters: <table border="1" data-bbox="504 1019 847 1084"> <thead> <tr> <th>Phone number</th> <th>Destination</th> <th>User</th> <th>Password</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>smtp.orange.fr</td> <td>GPRS</td> <td></td> </tr> </tbody> </table>	Phone number	Destination	User	Password	1	smtp.orange.fr	GPRS	
Phone number	Destination	User	Password						
1	smtp.orange.fr	GPRS							
4	Create an email service using Web Designer with the following parameters: <table border="1" data-bbox="504 1172 847 1237"> <tr> <td>SMTP server</td> <td></td> </tr> <tr> <td>SMTP server address</td> <td>smtp.orange.fr</td> </tr> </table> <p><b>Note:</b> The SMTP server address configured in Web Designer must match the one configured in the <b>Phone list</b> in step 3. Transfer the Web Designer project to the module.</p>	SMTP server		SMTP server address	smtp.orange.fr				
SMTP server									
SMTP server address	smtp.orange.fr								
5	The event configured to trigger the email in step 4 occurs.								
6	The module connects to the GPRS network and sends the email. <b>Note:</b> If Hang-up after email has been selected, the module disconnects automatically after the email has been sent.								

## Remote TSX ETG 302• GPRS Connection

The local TSX ETG 302• calls the remote TSX ETG 302• module using the GSM mode. With the callback on ring function activated, the remote TSX ETG 302• module automatically connects itself using the GPRS network.

The following diagram shows the connection between a local and a remote site:



Step	Description
1	Configure the remote TSX ETG 302• module with the callback on ring ( <i>see page 171</i> ) function.
2	Add the remote TSX ETG 302• module in the telephone list configuration ( <i>see page 216</i> ). <b>NOTE:</b> <ul style="list-style-type: none"> <li>After you have set new configuration parameters, reboot the module for them to take effect.</li> <li>The destination (<i>see page 216</i>) field is a comment string not used.</li> <li>The username and password are not mandatory.</li> </ul>
3	Enter the index number from the <b>Phone list</b> into the CMD_PARAM1 register ( <i>see page 103</i> ).
4	Use the command number 5 of the COMMAND ( <i>see page 103</i> ) register to open the GPRS connection of the remote module. <b>Results:</b> <ul style="list-style-type: none"> <li>The local TSX ETG 302• module calls the GSM number and waits until the line is disconnected before setting the COMMAND register to 0, see number 5 in the diagram.</li> <li>The remote module is connecting to the GPRS network using the callback on ring function, see number 6 in the diagram.</li> </ul>

## PPP Security: IP Control Configuration Page

### Introduction

TSX ETG 302• modules enable access to the module over the Internet.

Therefore, for increased security of remote access, the TSX ETG 302• module implements the following security services:

- VPN (*Virtual Private Network*) (*see page 199*) provides a higher level of security by encrypting the data transfer between two hosts.
- IP control provides a basic level of security by preventing devices with unknown IP addresses to connect to the module.

### IP Control

This control enables you to specify a list of IP address range authorized to connect to the module. Unauthorized IP addresses are rejected on connection attempt. IP control only protects PPP connections.

### IP Control Setup

To access the **IP control** configuration page, click **Modem/PPP Security** in the **Setup** menu of the website.

## View of the PPP Security Configuration Page

The choice between IP Control and VPN is exclusive. Only one type of security can be selected at the same time. The table not selected is grayed out.

IP control enable

IP Range authorized		
	From IP address	To IP address
1	62.44.4.0	62.44.1.255
2	80.10.23.100	80.10.23.100
3	62.117.0.0	62.117.255.255
4		

VPN enable

VPN Connections						
	Remote add...	Pre shared key	Mode	Remote LAN	Subnet mask	ETG client...
1	ETG.dyndns.o.	*****	Transport			
2	62.12.123.100	*****	Tunnel	192.168.2.0	255.255.255.0	High
3						
4						

Apply      Undo

© 2000-2009 Schneider Electric. All Rights Reserved.

## IP Control Parameters

Parameter	Description
<b>IP control enable</b>	Select this box to activate the IP control service. It enables you to authorize only one IP address or a range of IP addresses to communicate with the module. <b>Note:</b> If only one IP address must be allowed, enter the same IP address in both the <b>From/To IP address</b> columns.
<b>From IP address</b>	Start of the IP address range.
<b>To IP address</b>	End of the IP address range.

## VPN Parameters

For more information on the VPN Parameters, refer to PPP Security: VPN Configuration Page (*see page 213*).

**NOTE:** If VPN is checked and the table is empty. The module does not allow GPRS connections.

---

## 4.9 GPRS Communications and NAT Routing Functions

---

### Overview

This section describes the GPRS and NAT routing functions of the TSX ETG 30\*\* module.

### What's in this Section?

This section contains the following topics:

Topic	Page
NAT Routing Functions	177
Remote Access to TSX ETG 30** and Devices Using NAT	179
Intersite Communication Using NAT Routing	182
NAT On Port Routing Function	186
NAT On Port Configuration	188
NAT On Port Limitations	190
Modbus UnitID Routing Function	191
Modbus UnitID Routing Configuration	194
Port 502 Forwarding	195
<b>Port 502 Forwarding</b> Configuration	197



## NAT Routing Functions

### Overview

The Network Address Translation (NAT) protocol is used to enable the communication between remote LANs through a WAN using routing tables. The NAT protocol translates an IP address range from one LAN into another IP address range for the second LAN.

The NAT protocol is part of a router or firewall, NAT routing tables are created to map the IP address range through the local to global network and global to local network as shown in the following diagram:



### Routing Using NAT

The NAT protocol and its relation to ClassLess InterDomain Routing (CIDR) improves IP address depletion.

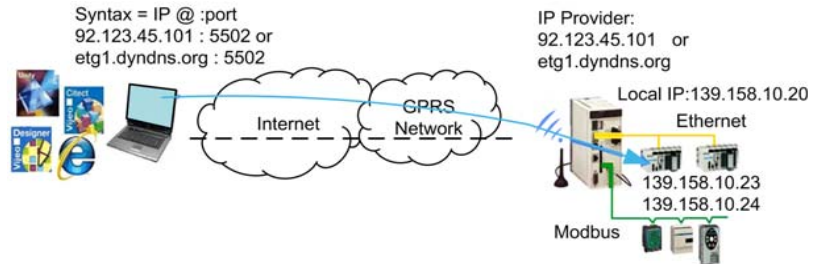
The main functions of NAT are to:

- provide routing capabilities from LAN to WAN and WAN to LAN
- reduce a significant amount of known public IP addresses by separating known public and private addresses
- allow a single device (for example a router) to control the network traffic between public and private networks
- provide a firewall by hiding private IP addresses
- increase the number of private IP addresses available
- combine multiple ISDN connections into a single Internet connection

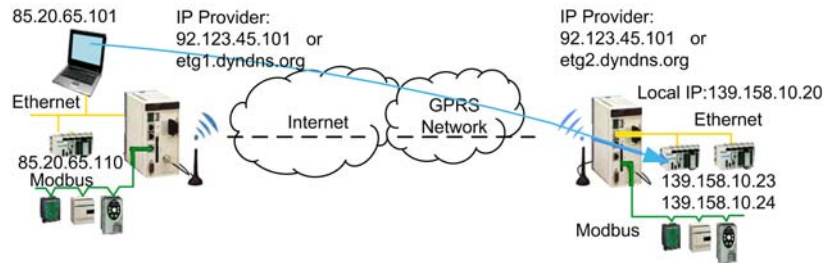
### Using NAT and the TSX ETG 302• Module

The TSX ETG 302• module offers NAT routing solutions described in the following principle diagrams:

A computer with an Internet connection can access a remote TSX ETG 302• module and associated devices:



A TSX ETG 302• module can access another remote TSX ETG 302• module and associated devices:



## NAT Routing Capabilities

The TSX ETG 302• module provides three types of transparent routing capabilities, based on:

- NAT On Port routing tables (*see page 186*), designed for IT and automation software as well as applications compliance
- Modbus UnitID routing tables (*see page 191*), designed for all legacy automation software and applications
- Port 502 Forwarding (*see page 195*) that provides quick and simple access to a single Ethernet device

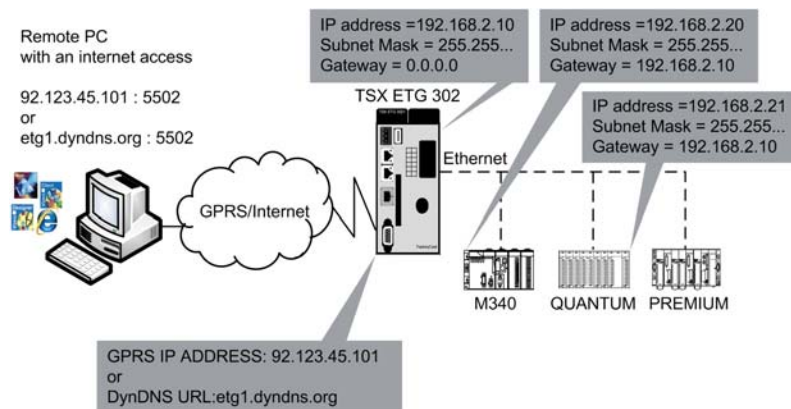
## Remote Access to TSX ETG 30\*\* and Devices Using NAT

### Presentation

This page describes how to access a remote TSX ETG 30\*\* module and devices using a PC connected to the Internet using the NAT service. The following examples describe various configurations and topologies.

### NAT On Port Topology Example

The following diagram shows an example of NAT On Port topology:



The following screenshot provides the NAT On Port routing table associated with the example:

Enable

Network Address Port Translation				
	Protocol	Source Port	Destination Address	Destin. Port
1	TCP ▼	5502	192.168.2.20	502
2	TCP ▼	5503	192.168.2.21	502

Apply Undo

The gateway address of each remote devices is mandatory and must be configured using the programming software of the devices. As shown in the previous diagram, the gateway address is the IP of the TSX ETG 302\*.

Remote access to the TSX ETG 302\* and devices:

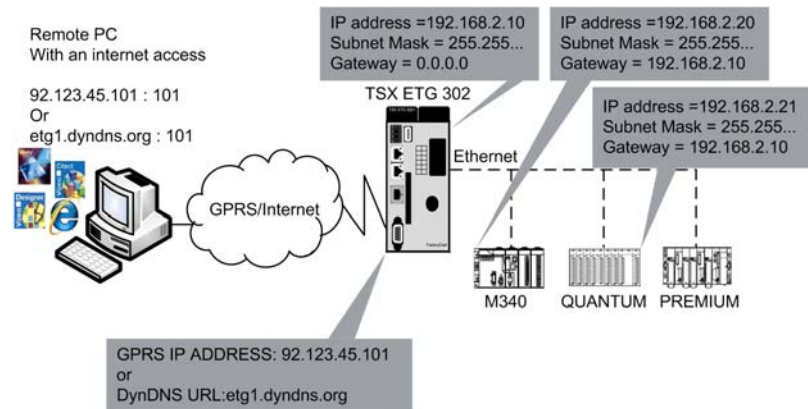
Remote access to the Ethernet devices is done with the following IP addresses or URL:

For the M340 PLC	For the Quantum PLC
92.123.45.101:5502	92.123.45.101:5503
etg1.dyndns.org:5502	etg1.dyndns.org:5503

**NOTE:** The notation of the IP addresses follows the syntax used with Unity Pro software (Version 4.1 or later), or other automation software such as Twido Soft or Power Suite, etc.

### Modbus UnitID Topology Example

The following diagram shows an example of Modbus UnitID topology:



The following table provides the Modbus UnitID configuration associated with the previous diagram:

UnitID Routing			
	Source UnitID	Destination IP address	Destin. UnitID
1	101	192.168.2.20	255
2	102	162.168.2.21	255

The gateway address of each remote devices is not mandatory when using the Modbus UnitID.

Remote access to the TSX ETG 302• and devices:

Remote access to the Ethernet devices is done with the following IP addresses or URL:

For the M340 PLC	For the Quantum PLC
92.123.45.101:101	92.123.45.101:102
etg1.dyndns.org:101	etg1.dyndns.org:102

**NOTE:** The notation of the IP addresses follows the syntax currently used with Unity Pro software (Version 4.1 or later), Vijeo Citect SCADA or third party SCADA.

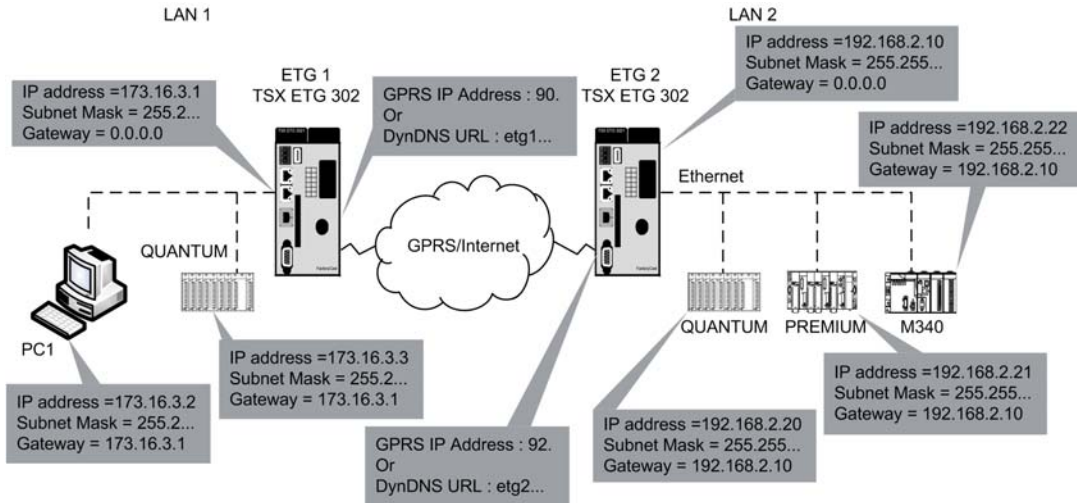
## Intersite Communication Using NAT Routing

### Presentation

This page describes how to create an intersite communication using the NAT On Port and Modbus UnitID functions.

### Example 1: Intersite Communication Using NAT On Port Routing Tables

The following diagram shows an example of intersite communication using the NAT On Port protocol:



### Configuring the Intersite Communication

The following screenshot provides the NAT On Port routing table associated with the ETG1:

Enable

Network Address Port Translation					
	Protocol		Source Port	Destination Address	Destin. Port
1	TCP	▼	5502	176.16.3.3	502
2					

Apply      Undo

The following screenshot provides the NAT On Port routing table associated with the ETG2:

Enable

Network Address Port Translation				
	Protocol	Source Port	Destination Address	Destin. Port
1	TCP ▼	5502	192.168.2.20	502
2	TCP ▼	5503	192.168.2.21	502

**NOTE:** To activate the NAT On Port service, you must select the **Enable** checkbox.

### Accessing a Device Using NAT On Port

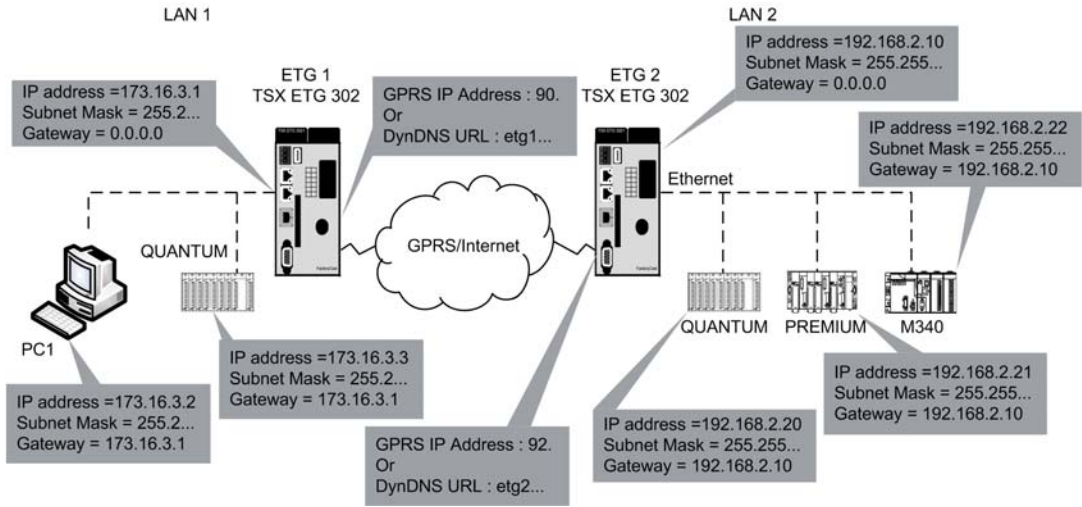
From PC1, the remote access to the Ethernet devices is done with the following IP addresses or URL:

For the Quantum PLC	For the Premium PLC
92.123.45.101:5502	92.123.45.101:5503
etg2.dyndns.org:5502	etg2.dyndns.org:5503

**NOTE:** The notation of the IP addresses follows the syntax currently used with the Unity Pro software.

### Example 2: Intersite Communication Using Modbus UnitID and NAT Routing Tables

The following diagram shows an example of intersite communication using the Modbus UnitID protocol:



### Configuring the Intersite Communication

The following table provides the NAT On Port routing table associated with the ETG1 in the previous diagram:

Enable

Network Address Port Translation				
	Protocol	Source Port	Destination Address	Destin. Port
1	TCP	5502	176.16.3.3	502
2				

**NOTE:** To activate the NAT On Port service, you must select the **Enable** checkbox.

The following table provides the Modbus UnitID configuration associated with the ETG2 in the previous diagram:

UnitID Routing			
	Source UnitID	Destination IP address	Destin. UnitID
1	101	192.168.2.20	255
2	102	162.168.2.21	255



---

### Accessing a Device Using Modbus UnitID and NAT

From the PC1 in the diagram, the remote access to the Ethernet devices is done with the following IP addresses or URL:

For the Quantum PLC	For the Premium PLC
92.123.45.101;101	92.123.45.101;102
etg2.dyndns.org;101	etg2.dyndns.org;102

**NOTE:** The notation of the IP addresses follows the syntax currently used with the Unity Pro software.

## NAT On Port Routing Function

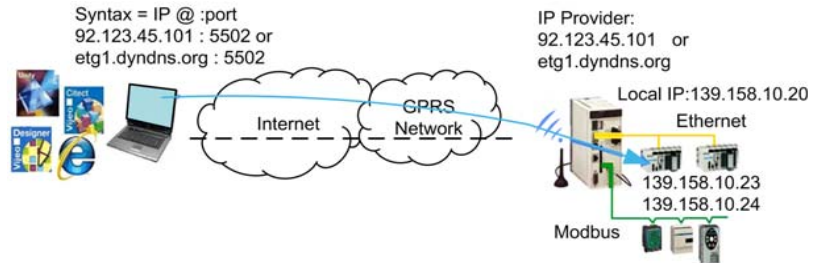
### Overview

The NAT On Port function allows routing between WAN and local Ethernet network through the TSX ETG 302• module (GPRS).

The TSX ETG 302• module can route a TCP or UDP/IP request based on the IP/port address translation. You must configure a routing table that indicates all the IP/port address translation related to each device or services to access.

### Configuration Example

The following diagram shows a configuration where a PC connected to the Internet accesses a remote device using the TSX ETG 302• via the NAT on Port service and the GPRS network:



In the TSX ETG 302•, configure a routing table for each device or service to access by providing the following parameters:

- TCP or UDP Protocol to be used
- incoming source port
- destination IP address resulting from the NAT translation
- destination port resulting from the NAT translation

For a complete description of each element of the routing table, refer to NAT On Port Configuration (*see page 188*). Find more information on the NAT On Port topology (*see page 179*) in the remote access to TSX ETG 302• and devices.

The following figure shows the NAT routing table of the previous configuration (see page 186) example:

Enable

Network Address Port Translation				
	Protocol	Source Port	Destination Address	Destin. Port
1	TCP ▼	5502	139.158.20.23	502
2	TCP ▼	5080	139.158.20.23	80
3	TCP ▼	6502	139.158.20.24	502
4	▼			
5	▼			
6	▼			
7	▼			
8	▼			
9	▼			
10	▼			

You can access the remote device using automation software such as Unity Pro (version 4.1 or later), SCADA (Vijeo Citect) or a Web browser when the routing table is completed.

In this example, the syntax for accessing the remote device changes depending whether the user wants to access with an automation software or via Web browser:

Unity Pro V4.1 Syntax	Web Browser Syntax
92.123.45.101:5080	http://92.123.45.101:5080
etg.dyndns.org:5080	http://etg.dyndns.org:5080

For more information on how to access a remote device with Unity Pro V4.1, refer to Accessing a PLC via a Network (see *Unity Pro, Operating Modes*) available in the Unity Pro Operating Mode Manual (see *Unity Pro, Operating Modes*).

## NAT On Port Configuration

### Introduction

The NAT On Port function is configured via a **Setup** webpage from the embedded website or via the Web Designer software.

**NOTE:** Write access is controlled by a security password with the default value **USER**.

### Accessing the NAT Setup Page

Step	Action
1	Click the <b>Setup</b> link on the <b>Home</b> page.
2	Click the <b>NAT</b> link on the <b>Setup</b> page.

### View of the NAT Setup Page

Enable

Network Address Port Translation				
	Protocol	Source Port	Destination Address	Destin. Port
1	TCP	5502	139.158.20.23	502
2	TCP	5080	139.158.20.23	80
3	TCP	6502	139.158.20.24	502
4				
5				
6				
7				
8				
9				
10				

**NOTE:** After you have set new configuration parameters, reboot the module for them to take effect.

---

**NAT On Port Parameters**

<b>Parameter</b>	<b>Action</b>
<b>Enable</b>	Click the check box to select the <b>NAT On Port</b> function.
<b>Protocol</b>	Specify the protocol to use: <b>TCP</b> or <b>UDP</b> .
<b>Source Port</b>	Enter the translated input port. The value must be between 0 and 65535. When possible, use a private port with a value higher than 1024.
<b>Destination Address</b>	Enter the IP address (using dot notation only) that is the result of the translation.
<b>Destin. Port</b>	Enter the port that is the result of the translation. The value must be between 0 and 65535.

**NOTE:** Enter no more than 32 lines when configuring the Modbus UnitID routing function.

## NAT On Port Limitations

### Presentation

This page describes the NAT On Port limitations with standard port redirection. It is recommended that you use private ports (>1024) for the NAT source to avoid the following limitations using the standard ones.

### Limitations

The standard port redirection (for ports 502 and 80) has the following limitations:

- The standard port redirection changes the behavior of the Data and Graphic Editors for port 502: the values from the NAT's destination equipment display instead of the values from the module.
- The standard port redirection changes the web server behavior for port 80: the website of the NAT's destination equipment displays instead of that of the module.

**NOTE:** This same behavior occurs with other standard ports such as FTP, NTP, and SNMP.

### ***NOTICE***

#### **RISK OF INCORRECT DATA**

Do not use the write function in the Graphic and Data Editors for nonpersistent and direct variables when the NAT On Port is enabled and port 502 or 80 is redirected.

**Failure to follow these instructions can result in equipment damage.**

---

## Modbus UnitID Routing Function

### Overview

The Modbus UnitID function allows routing between WAN and LAN through the GPRS modem connection (TSX ETG 30\*\*).

The TSX ETG 30\*\* module can route an Modbus TCP request to the 502 port (Modbus) of a remote device using its UnitID information.

### Advantages and Limitations

The main advantages offered by the Modbus UnitID routing are the following:

- **Compatibility:** The UnitID is always available with the Modbus TCP protocol and configurable with Unity Pro (Version 4.1 or later) and with legacy Schneider Electric tools and software such as, Concept, Twido Suite, Power Suite, Vijeo Designer, etc.
- **Simplified implementation:** A Modbus UnitID routing table must be configured indicating the destination IP address routing associated with the UnitID. The TSX ETG 30\*\* controls the connection using the Ethernet configured address, so that a device on Ethernet is seen as a device on Modbus serial.
- **Ethernet device accessibility:** The Modbus UnitID function offers access to Ethernet devices for services based on Modbus such: SOAP services, graphic and remote data editor for non persistent variables.

The service also presents some limitations:

- **Availability on port 502 connections:** The Modbus UnitID uses port 502 connections for messaging communication (64 maximum). The oldest connection is disconnected when none are available.
- **Diagnostics:** Diagnostics are not provided with Modbus UnitID. However, the TCP/IP Messaging (*see page 369*) screen can be used to control the traffic of the Modbus UnitID service.
- **Device Explorer:** The Ethernet equipment available in the Modbus UnitID routing table are detected via the device explorer (*see page 247*) discovery function from the Modbus serial link with the UnitID present in the routing table.

### Before Configuring the Modbus UnitID Routing

For Modbus requests, the Modbus UnitID routing table has the highest priority over the Modbus serial requests. Before configuring the Modbus UnitID routing, consider this priority rule as the Modbus Unit ID values must not already be used by the Modbus serial devices.

For the Modbus requests, the Modbus UnitID routing table has the highest priority over the Modbus serial requests.

## ⚠ WARNING

### UNINTENDED EQUIPMENT OPERATION

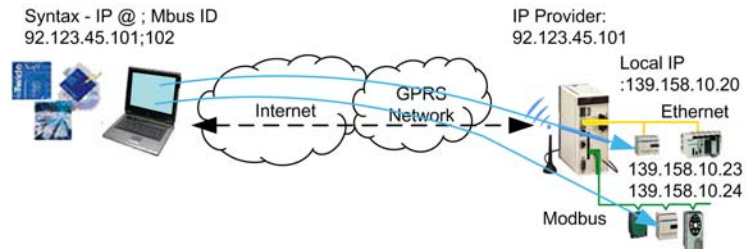
Before configuring the Modbus UnitID Routing, give the Modbus UnitID routing table priority over all Modbus serial requests.

Do not use values for the Modbus UnitID that are already being used by the Modbus serial devices.

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

### Configuration Example

The following diagram shows a configuration where a PC connected to the Internet accesses a remote device using a GSM TSX ETG 302• module via the Modbus UnitID function and the GPRS network:



Configure a Modbus UnitID table to set the connection by providing the:

- **Source UnitID** to be routed on Ethernet
- **Destination IP** address
- **UnitID** destination of the request

For a complete description of each element of the routing table, refer to Modbus UnitID Routing Configuration (*see page 194*) page. Find more information on the Modbus UnitID topology (*see page 180*) in remote access to TSX ETG 30•• and devices.



The following figure shows the UnitID Routing table of the previous configuration (see page 192) example:

UnitID Routing			
	Source UnitID	Destination IP address	Destin. UnitID
1	101	139.168.10.20	255
2	102	139.168.10.23	255
3	103	139.168.10.24	255
4			
5			
6			

You access the remote device using automation software as Unity Pro, Vijeo Designer, Twido Suite, Concept, Power Suite, etc.

In this example, the syntax for accessing the remote device with Unity Pro Version 4.1 is as follows: 92.123.45.101;102

For more information on how to access a remote device with Unity Pro (Version 4.1 or later), refer to *Accessing a PLC via a Network* (see *Unity Pro, Operating Modes*) available in the *Unity Pro Operating Modes Manual* (see *Unity Pro, Operating Modes*).

## Modbus UnitID Routing Configuration

### Introduction

The Modbus UnitID function is configured via a **Setup** web page from the embedded website or via Web Designer software.

**NOTE:** Write access is controlled by a security password whose default value is **USER**.

### Accessing the Modbus UnitID Routing Setup Page

To access the Modbus UnitID configuration screen, click the **UnitID Routing** link in the **Setup** menu of the website

### View of the Modbus UnitID Routing Setup Page

UnitID Routing			
	Source UnitID	Destination IP address	Destin. UnitID
1	101	139.168.10.20	255
2	102	139.168.10.23	255
3	103	139.168.10.24	255
4			
5			
6			

**NOTE:**

- After you have set new configuration parameters, reboot the module for them to take effect.
- If the table is empty, the Modbus UnitID Routing service is not activated.

### Modbus UnitID Routing Parameters

Parameters	Action
<b>Source UnitID</b>	Enter the UnitID value to route on the ethernet. The UnitID must be an integer value set between 0 and 255.
<b>Destination IP Address</b>	Enter the IP address destination (using dot notation only) of the TCP request.
<b>Destin. UnitID</b>	Enter the UnitID of the destination request. The UnitID must be an integer value set between 0 and 255.

**NOTE:** Enter a maximum of 32 lines when configuring the Modbus UnitID routing function.

## Port 502 Forwarding

### Overview

The Port 502 Forwarding function allows routing between WAN to LAN through the TSX ETG 30•• range.

The Port 502 Forwarding function is similar to the NAT On Port function. However, it can only be used to transfer request to a single Ethernet remote device at a time.

**NOTE:** The NAT On Port function is exclusive of the Port 502 Forwarding function. The latter cannot be activated if NAT On Port is activated.

### Advantages and Limitations

The Port 502 Forwarding function does not require any configuration and provides the following advantages:

- Quick configuration and activation/deactivation of the function.
- Quick switching between remote devices.
- Instant connection to the remote device.

This function also presents some limitations:

- Port 502 Forwarding routes to one remote device at a time, hence, all incoming requests are routed to this device.
- The Address setting will be lost if the module is rebooted.
- For permanent routing, Modbus UnitID or NAT functions are more appropriate.
- The data editor cannot be used with the Port 502 Forwarding and non persistent or direct variables. The displayed values are the values from the destination equipment of the Port 502 Forwarding.

### ***NOTICE***

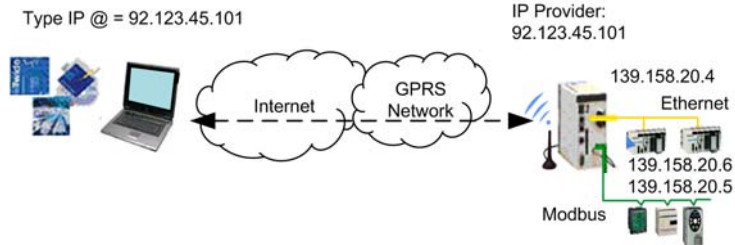
#### **RISK OF INCORRECT DATA**

Do not use the write function in the Graphic and Data Editors for non persistent and direct variables when the Port 502 Forwarding function is activated.

**Failure to follow these instructions can result in equipment damage.**

### Configuration Example

The following diagram shows a configuration where a PC connected to the Internet accesses a remote device using a TSX ETG 30•• via the Port 502 Forwarding function and the GPRS network:



You must provide the IP address of the remote device.

For a complete description of the Port 502 Control window, refer to Port 502 Forwarding Configuration (*see page 197*).

The following figure shows the Port 502 Forwarding control window of the previous configuration (*see page 196*) example:

Remote device									
IP address	139	.	158	.	20	.	6	: Port 502 (TCP)	
Apply		Undo							

The remote device is accessible as soon as you enter the IP address and press the **Apply** button.

## Port 502 Forwarding Configuration

### Introduction

The Port 502 Forwarding function is configured by applet available from the embedded website or from the Web Designer software.

**NOTE:** Write access is controlled by a security password whose default value is **USER**.

### Accessing the Port 502 Forwarding Control Page

Step	Action
1	Click the <b>Control</b> menu available from the <b>Home</b> page.
2	Click the <b>Port 502 Forwarding</b> link on the <b>Control</b> page.

### View of the Port 502 Forwarding Control Page

**Remote device**

IP address  .  .  .  : Port 502 (TCP)

**NOTE:** The address setting will be lost if the module is rebooted.

### Port 502 Forwarding Parameters

To configure and activate the **Port 502 Forwarding** function, the user must enter the IP address of the remote device.

The table below shows how to connect/disconnect to a remote device:

Step	Action
1	Check that the GPRS connection is activated.
2	Enter the IP address of the remote device.
3	Click <b>Apply</b> to confirm the address change. <b>Result:</b> The connection to the remote de device is instantaneous.
4	Delete the IP address to deactivate the connection with the remote device.

## 4.10 GPRS Communications and VPN Functions

---

### Overview

This section describes the GPRS and VPN services of the TSX ETG 30\*\* module.

### What's in this Section?

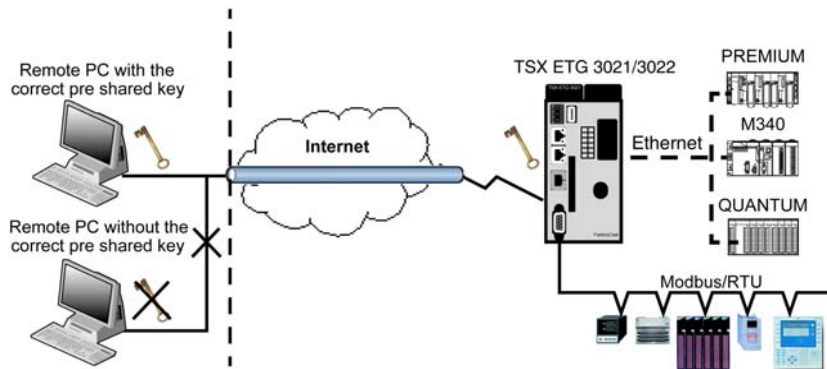
This section contains the following topics:

Topic	Page
Increasing Communication Security with a VPN	199
GPRS and VPN Features	201
Remote Access to TSX ETG 302• and Devices	202
Intersite Communications with VPN	209
PPP Security: VPN Control Configuration Page	213

## Increasing Communication Security with a VPN

### Presentation

VPN enables TSX ETG 302• modules to establish connections via private, bi-directional, encrypted tunnels over the Internet between the central sites and your remote devices.



VPN increases the level of security of the communication:

- Between a PC connected to the Internet and the TSX ETG 302• gateway. The remote devices connected to the gateway appear virtually in your LAN. Once the tunnel is established, your programming software and monitoring tools can access the remote devices transparently as if they are in the same LAN.
- Between two sites via VPN tunnels between two TSX ETG 302• gateways. Any device from a site can access the devices of the remote site. This capability can be used as an alternative solution by replacing leased lines between sites.

The VPN service is based on:

- the IPsec (*Internet Protocol Security*) protocols that improves security by creating secured tunneled connections
- IKE (*Internet Key Exchange*) for session authentication with preshared key exchange

**NOTE:** VPN in tunnel mode is mandatory for using WAN-to-LAN routing to access transparently devices connected to a TSX ETG 302• gateway.

### VPN Limitations

In VPN mode, after any reboot or power off of the remote TSX ETG 302• module, the VPN client software must be restarted on the client side (PC or ETG).

## IPsec Presentation

IPsec is a suite of protocols that improves security of Internet Protocol (IP) communications by authenticating and/or encrypting each IP packet in a data stream. IPsec also includes protocols for cryptographic key establishment. IPsec operates in 2 modes:

- Transport (end-to-end) mode increases the level of security of packet traffic by having the end-point computers do the security processing.
- Tunnel (portal-to-portal) mode improves communication security for the data exchanged. A single node provides security to several machines (even to whole LANs).

## VPN/IPsec Services

- Tunnel mode: the entire IP packet (data plus the message headers) is encrypted and/or authenticated. The tunnel mode is used for network-to-network communications.
- Transport mode: only the payload (the data you transfer) of the IP packet is encrypted and/or authenticated. The transport mode is used for host-to-host communications.
- Data authentication: uses the AH (*Authentication Header*) protocol to provide different levels of hash functions (from SHA to MD5).
- Data encryption: uses the ESP (*Encapsulating Security Payload*) protocol to provide different levels of encryption (from 3DES to DES lite).

## VPN Clients

When connecting a PC to a remote TSX ETG 302• module, the module acts as a VPN server. A VPN client interface must be running on the client side (PC or other TSX ETG 302•).

When connecting a TSX ETG 302• client to a remote TSX ETG 302•, the TSX ETG 302• client can act as a VPN client. TSX ETG 302• VPN setup includes this VPN client configuration.

The following VPN client can be used with a TSX ETG 302•:

- TheGreenBow VPN client software (recommended)
- VPN client service provided by Windows operating systems 2000, XP, Vista
- other VPN client software compliant with IPsec/IKE protocols

## Limitations

- VPN enables you to secure only PPP connections. It does not work for Ethernet connections.
- Only 8 VPN connections can be configured and active at the same time.

**NOTE:** You can use VPN with many different modem connection types. However, it is recommended to use it only for GPRS and PSTN connections.



---

## GPRS and VPN Features

### VPN Clients Overview

To communicate with the module using VPN, a VPN client software must be installed on your PC.

We recommended you use the following VPN clients utilities with TSX ETG 302• modules:

- Windows VPN client batch file (*bat* file) provided in the CD ROM in the VPN folder
- TheGreenBow VPN client software (recommended VPN client software)

### Windows VPN Client Batch File

We provide a sample Windows VPN client batch file (.bat file). This batch file can open a VPN connection using the Windows IPsec service.

The Windows IPsec service is available on Windows 2000/XP/Vista, however you must check that it is installed and enabled. If it is not already installed, you can add it by installing the "Windows support tools" and enable the IPsec service. Refer to your operating system documentation for more information.

Use the following commands to check if the IPsec service is installed and enabled on your operating system:

Under Windows, click **Start** → **Run...** and type 'cmd' in the Run window:

- Windows 2000: type `ipsecpol` in the command prompt.
- Windows XP: type `ipseccmd` in the command prompt.
- Windows Vista: type `netsh ipsec` in the command prompt.

If the following message appears 'Windows cannot find...', you must install or activate the IPsec service. Refer to your operating system documentation for more information.

### TheGreenBow VPN Client Software

TheGreenBow VPN Client is a software that allows a user to establish a VPN client connection to the module over the Internet. TheGreenBow VPN Client supports the ESP protocol only.

Refer to [www.thegreenbow.com](http://www.thegreenbow.com) for more information about the software.

## Remote Access to TSX ETG 302• and Devices

### Internet Access to the TSX ETG 302• from a Remote PC

The following configuration scenarios use various characteristics of the GPRS connection:

- The GPRS IP address is static or dynamic.
- TCP ports are blocked or not by the service provider.
- VPN is needed or not.

### Limitations

The default route must be used to send a request through a GPRS connection. If the Ethernet link has a default gateway configured, this route may become overloaded during the GPRS connection. It is not possible to route through Ethernet when GPRS is connected.

Generally, the GPRS connection automatically provides a DNS at connection. However, in the case above, it replaces the DNS configured during the GPRS connection.

The component NTP must get a return in 6 ms maximum (1 s in GPRS) to work correctly through the GPRS connection.

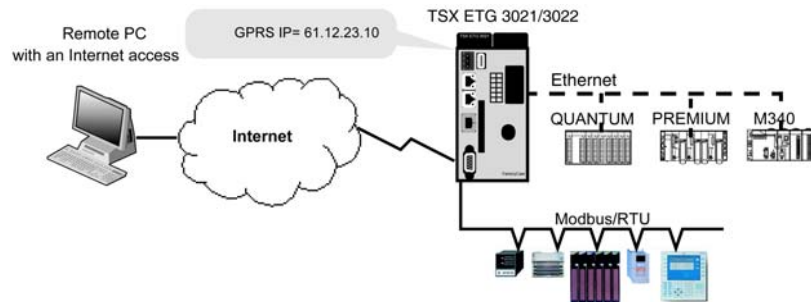
When a GPRS connection is open, attempting to send an SMS will close the GPRS connection. If the module is in permanent connection, it will reconnect with the GPRS after the SMS has been sent.

**NOTE:** You can use the Phone List configuration (*see page 216*) to transmit email (*see page 172*), but under no other circumstances should you use it with GPRS.

### Example 1

You are configuring remote access to TSX ETG 302• and devices with:

- public ports not blocked by the service provider
- SIM card with a static GPRS/internet IP address (i.e 61.12.23.10)
- VPN not used (you can use the IP control function)



Access the TSX ETG 302• and devices remotely:

You can access the TSX ETG 302• website by typing the GPRS IP address of the module (61.12.23.10) in the address bar of a Web browser or using Web Designer.

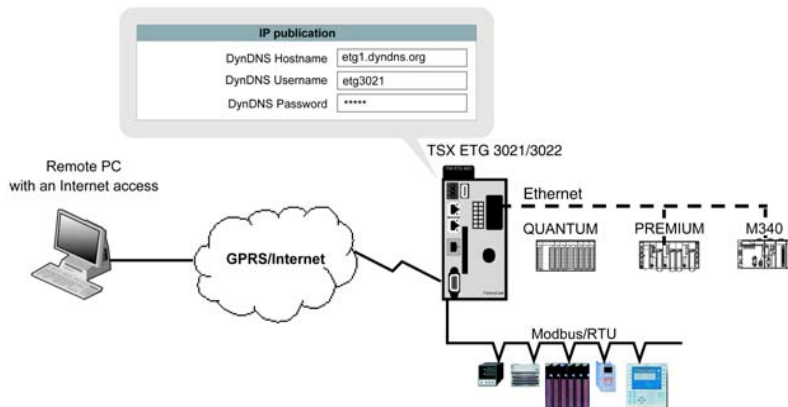
Modbus devices can be accessed remotely from any PLC programming software or SCADA using their Modbus slave ID.

Remote access to Ethernet devices requires a VPN tunnel (*see page 199*) configuration setup.

## Example 2

You are configuring remote access to TSX ETG 302• and devices with:

- public ports not blocked by the service provider
- SIM card with a dynamic GPRS IP address (requires use of DynDNS URL instead of IP address)
- VPN not used (you can use the IP control function)



Access the TSX ETG 302• and devices remotely:

You can access the TSX ETG 302• website by typing the URL of the module (etg1.dyndns.org) in the address bar of a Web browser.

Modbus devices can be accessed remotely from any PLC programming software or SCADA using their Modbus slave ID.

Remote access to Ethernet devices requires a VPN tunnel (*see page 199*) configuration setup.

**NOTE:**

- You can also use the email service of the module to publish the public IP address of the module (*see page 277*). Use both (DynDNS and email) to have an alternative method if one service does not work.
- A GPRS IP address can be known inside the modem diagnostic Web page of the TSX ETG 302• module.
- You can ping the URL to retrieve its IP address if the client application does not support URL addressing.

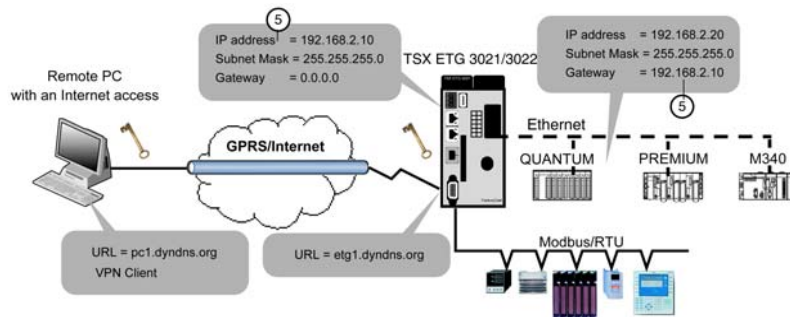
**Example 3**

You are accessing the Ethernet devices connected to the gateway using VPN in Tunnel mode with:

- SIM card with a GPRS dynamic IP address
- VPN in Tunnel mode (VPN ports not blocked by the service provider). The tunnel mode is mandatory to access Ethernet devices.
- VPN ports not blocked by the service provider
- remote PC with a dynamic IP address (use DynDNS address)

The following parameters must be the same in 2 or more places. If not, the connection may not work

Legend	Parameter	Setting
1	TSX ETG 302• GPRS dynamic address	etg1.dyndns.org (use DynDNS address)
2	PC Internet address	pc1.dyndns.org (DynDNS address) or 10.10.0.10 (virtual address)
3	PC and TSX ETG 302•	Configure the same Preshared key on both sides.
4	PC and TSX ETG 302•	Configure VPN Tunnel on TSX ETG 302• side and VPN client utility on PC side.
5	TSX ETG 302• local address Quantum gateway address	192.168.2.10 192.168.2.10
6	Remote LAN network	192.168.2.* or 192.168.2.0 + subnet mask
7	Virtual address of the remote network (only used for TheGreenBow)	10.10.0.10



Access the TSX ETG 302• and devices remotely:

You can access the TSX ETG 302• module by typing the Ethernet IP address of the module (192.168.2.10) in the address bar of a Web browser.

You can also access the Ethernet devices connected to the module as if you are on the same LAN by typing the IP address of the Ethernet device (i.e. 192.168.2.20) in the address bar of a Web browser.

Modbus devices can be accessed remotely from any PLC programming software or SCADA using their Modbus slave ID.

**NOTE:** When accessing a remote device via a Quantum PLC, the gateway address of the Quantum must match the address of the module to be accessed.

There are two ways you can configure the VPN client:

- Using the *.bat* file and Windows XP:  
Setup parameters on the TSX ETG 302• side:

VPN Connections						
	Remote address	Pre shared key	Mode	Remote LAN	Subnet mask	ETG client encryption
1	pc1.dyndns.org	*****	Tunnel			

2 3 4

Setup parameters on the PC side:

```

1
ipseccmd -l 3DES-SHA-2 -n AH[MD5] -f 0=192.168.2.* -t
3 6 4
etg1.dyndns.org -a PRESHARE:"etg1presharedkey" -p
"ETGTunnel" -r "PCToTarget" -lk 3600s -w reg -x

2
ipseccmd -l 3DES-SHA-2 -n AH[MD5] -f 192.168.2.*=0 -t
3 6 4
pc1.dyndns.org -a PRESHARE:"etg1presharedkey" -p "ETGTunnel"
-r "TargetToPC" -lk 3600s -w reg -x

```

- Using the TheGreenBow software, you must configure the client as follows:  
Setup parameters on the TSX ETG 302• side:

VPN Connections						
	Remote address	Pre shared key	Mode	Remote LAN	Subnet mask	ETG client encryption
1	pc1.dyndns.org	*****	Tunnel	10.10.0.10		

②
③
④
⑦

Setup parameters on the PC side (TheGreenBow) VPN client software:

### Phase 1 (Authentication)

Name:

Interface:

Remote Gateway:  ①

Preshared Key:  ③

Confirm:

Certificate:

IKE

Encryption:

Authentication:

Key Group:

### Phase 2 (IPSec Configuration)

Name:

VPN Client Address:  ⑦

Address type:

Remote host address:  ⑥

Subnet Mask:

ESP

Encryption:

Authentication:

Mode:  ④

PFS Group:

**NOTE:**

- You can use the email service of the module to retrieve the public IP address of the module (see page 277). Use both DynDNS and email as an alternative method if one service does not work.
- You can also ping the URL to retrieve its IP address if the client application does not support URL addressing.
- When using DynDNS for the remote PC, the TSX ETG 302• module checks each minute if the public IP address has changed. Thus, it exchanges about 100 kB of data every day for this function. To prevent useless exchange of data, use an On Demand connection.

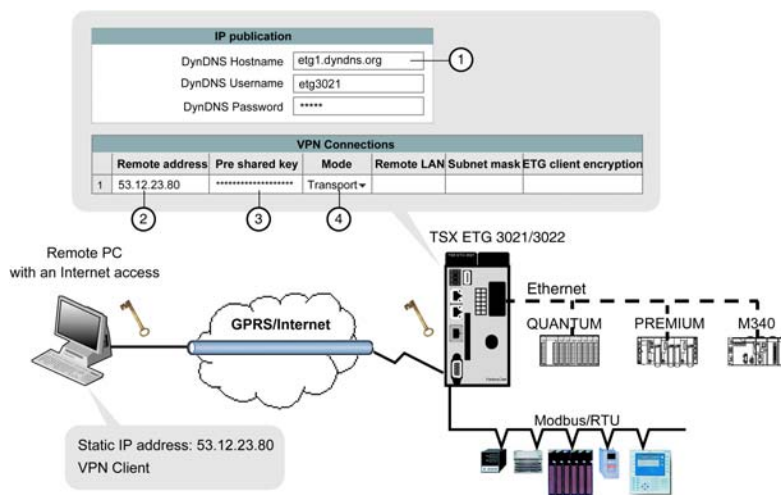
## Example 4

You are configuring remote access to TSX ETG 302• to devices with:

- SIM card with a GPRS dynamic IP address
- remote PC with a static public IP address: 53.12.23.80
- VPN in Transport mode

The following parameters must be the same in 2 or more places, if not, the connection may not work.

Legend	Parameter	Setting
1	ETG dynamic address	etg1.dyndns.org (DynDNS address)
2	PC static address	53.12.23.80
3	PC and ETG VPN authentication	same Preshared key on both side
4	PC and ETG VPN mode	Transport



Access the TSX ETG 302• and devices remotely:

You can access the TSX ETG 302• website by typing the URL of the module (etg1.dyndns.org) in the address bar of a Web browser.

Modbus devices can be accessed remotely from any PLC programming software or SCADA using their Modbus slave ID.

**NOTE:** In transport mode, remote access to the Ethernet devices is not allowed.

There are two ways you can configure the VPN client:

- Using the `.bat` file and Windows XP:

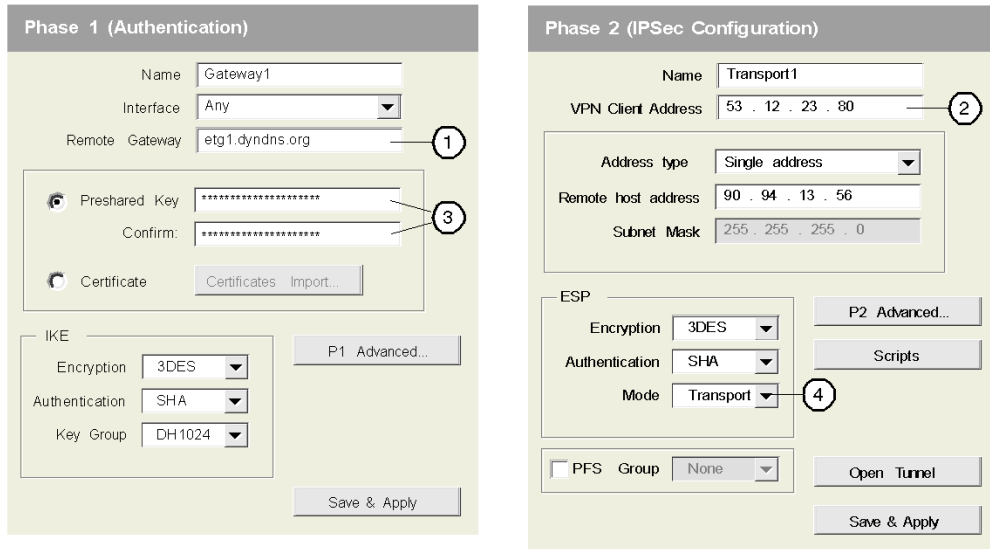
```

ipseccmd -ls 3DES-SHA-2 -n AH[MD5] -f 0=etg1.dyndns.org -a
PRESHARE:"etg1presharedkey" -p "ETGTransport" -r
"PCToTarget" -lk 3600s -w reg -x

ipseccmd -ls 3DES-SHA-2 -n AH[MD5] -f etg1.dyndns.org=0 -a
PRESHARE:"etg1presharedkey" -p "ETGTransport" -r
"TargetToPC" -lk 3600s -w reg -x
    
```

**NOTE:** The two lines above have been word wrapped but each appear on one line.

- Using the TheGreenBow software, you must configure the client as follows:



**NOTE:**

- You can use the email service of the module to retrieve the public IP address of the module (see page 277) via email. Use both (DynDNS and email) to have an alternative method if one service does not work.
- You can also ping the URL to retrieve its IP address if the client application does not support URL addressing.
- If the public IP address of the remote PC is dynamic, use the DynDNS Updater for the PC.
- Use the tunnel mode to access the module and the Ethernet network (see page 200).



## Intersite Communications with VPN

### Presentation

This page describes how to create intersite communications using two gateways and a VPN connection.

### VPN Limitations

VPN connection between two TSX ETG 302•:

- If you have a static IP address on GPRS public network then you can configure a device service under Web Designer. You need to specify this IP address in properties page of device service.
- If you have a dynamic IP address on GPRS public network then you cannot communicate between two TSX ETG 302• because you need to use a domain name and DNS is not supported by device service under TSX ETG 302•.
- VPN is supported between two TSX ETG 302• but IP address allowed to establish VPN connection are only IP addresses range from LAN.

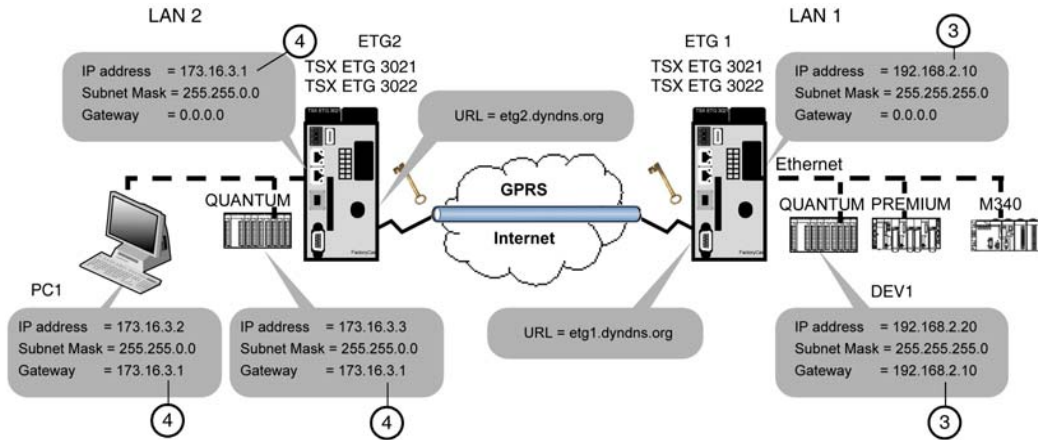
### Example 1: Module with Dynamic Public IP Address

Two TSX ETG 302• modules connected to the Internet using:

- SIM card with a dynamic public IP address
- VPN in Tunnel mode. The tunnel mode is mandatory to access Ethernet devices (VPN ports not blocked by the service provider).

The following parameters must be the same in 2 or more places, if not, the connection may not work.

Legend	Parameter	Setting
1	ETG1 GPRS IP address	etg1.dyndns.org (use DynDNS address)
2	ETG2 GPRS IP address	etg2.dyndns.org (use DynDNS address)
3	Ethernet address ETG1	192.168.2.10
4	Ethernet address ETG2	173.16.3.1
5	ETG1 & ETG2 VPN authentication	Preshared key
6	ETG1 & 2 VPN mode	Tunnel
7	Remote LAN (ETG1 side)	173.16.*.* or 173.16.0.0 + subnet mask
8	Remote LAN (ETG2 side)	192.168.2.* or 192.168.2.0 + subnet mask



VPN configuration (ETG1 side):

VPN Connections						
	Remote address	Pre shared key	Mode	Remote LAN	Subnet mask	ETG client encryption
1	etg2.dyndns.org	*****	Tunnel	173.16.0.0	255.255.0.0	

Annotations: 2 (Remote address), 5 (Pre shared key), 6 (Mode), 7 (Remote LAN)

VPN configuration (ETG2 side):

VPN Connections						
	Remote address	Pre shared key	Mode	Remote LAN	Subnet mask	ETG client encryption
1	etg1.dyndns.org	*****	Tunnel	192.168.2.0	255.255.255.0	Medium

Annotations: 1 (Remote address), 5 (Pre shared key), 6 (Mode), 8 (Remote LAN)

DynDNS configuration:

ETG 1		ETG 2	
IP publication		IP publication	
DynDNS Hostname	etg1.dyndns.org	DynDNS Hostname	etg2.dyndns.org
DynDNS Username	etg1	DynDNS Username	etg2
DynDNS Password	*****	DynDNS Password	*****

Annotations: 1 (ETG 1), 2 (ETG 2)

## Accessing a Device

Once the VPN connection has been established, all devices communicate using their Ethernet IP addresses as if they are in the same LAN. The public IP address is only used for the VPN connection.

For instance, PC1 communicates with DEV1 using the Ethernet IP address: 192.168.2.20.

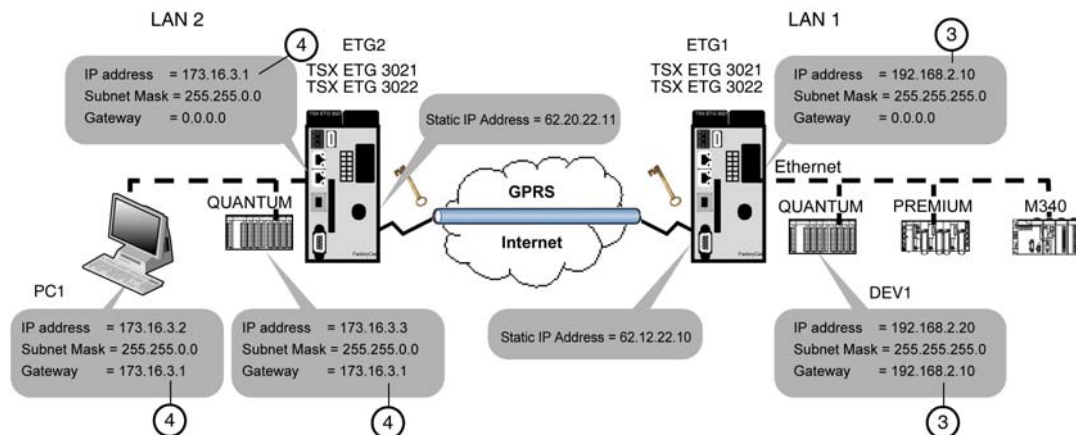
### Example 2: Module with Static IP Address

Two TSX ETG 302\* modules connected to the Internet using:

- module with a static IP address
- VPN in Tunnel mode. The tunnel mode is mandatory to access Ethernet devices (VPN ports not blocked by the service provider).

The following parameters must be the same in 2 or more places, if not, the connection may not work.

Legend	Parameter	Setting
1	ETG1 static IP address	62.12.22.10
2	ETG2 static IP address	62.20.22.11
3	Ethernet address ETG1	192.168.2.10
4	Ethernet address ETG2	173.16.3.1
5	ETG1 & ETG2 VPN authentication	Preshared key
6	ETG1 & 2 VPN mode	Tunnel
7	Remote LAN (ETG1 side)	173.16.*.* or 173.16.0.0 + subnet mask
8	Remote LAN (ETG2 side)	192.168.2.* or 192.168.2.0 + subnet mask



VPN configuration (ETG1 side):

VPN Connections						
	Remote address	Pre shared key	Mode	Remote LAN	Subnet mask	ETG client encryption
1	62.20.22.11	*****	Tunnel	173.16.0.0	255.255.0.0	

VPN configuration (ETG2 side):

VPN Connections						
	Remote address	Pre shared key	Mode	Remote LAN	Subnet mask	ETG client encryption
1	62.12.22.10	*****	Tunnel	192.168.2.0	255.255.255.0	Medium

## PPP Security: VPN Control Configuration Page

### Introduction

TSX ETG 302• modules enable access to the module over the Internet.

Therefore, for increased security of remote access, the TSX ETG 302• module implements the following security services:

- VPN (Virtual Private Network) provides a higher level of security by encrypting the data transfer between two hosts.
- IP control (*see page 174*) provides a basic level of security by preventing devices with unknown IP addresses to connect to the module.

### VPN Setup

To access the VPN configuration page, click **Modem/PPP Security** in the **Setup** menu of the website.

### View of the PPP Security Configuration Page

The choice between IP Control and VPN is exclusive. Only one type of security can be selected at the same time. The table not selected is grayed out.

IP control enable

IP Range authorized			
	From IP address	To IP address	
1	62.44.4.0	62.44.1.255	▲
2	80.10.23.100	80.10.23.100	■
3	62.117.0.0	62.117.255.255	▼
4			▼

VPN enable

VPN Connections						
	Remote add...	Pre shared key	Mode	Remote LAN	Subnet mask	ETG client...
1	ETG1.dyndns.	*****	Transport			▼ ▲
2	62.12.123.100	*****	Tunnel ▼	192.168.2.0	255.255.255.0	High ▼ ■
3			▼			▼
4			▼			▼

Apply      Undo

## VPN Parameters

Parameter	Description
<b>VPN enable</b>	Select this box to activate the VPN service. The module or network you want to communicate with must be listed in the <b>VPN Connections</b> table.
<b>Remote address</b>	Remote IP address of the remote host (module, PC or remote network). You can use a dot notation (x.x.x.x) or a symbolic address name.
<b>Preshared key</b>	Shared secret key used between the module and the remote for authentication. The sending and receiving devices must share the same key, up to 8 characters is recommended.
<b>Mode</b>	<ul style="list-style-type: none"> <li>● <b>Transport</b> mode encrypts the data portion of each IP packet but leaves the header un-encrypted. Only the TSX ETG 302• module is accessible.</li> <li>● <b>Tunnel</b> mode encrypts both the header and the data of the IP packet providing more communication privacy. The complete TSX ETG 302• local network is accessible.</li> </ul>
<b>Remote LAN</b>	In the <b>Tunnel</b> mode, this is the virtual IP address of the client or IP network. If the field is empty, there is no virtual IP address and only the host with the address in the <b>Remote address</b> field can access the module.
<b>Subnet mask</b>	Mask for the network if you communicate with an IP network. If the field is empty, the subnet mask is 255.255.255.255 and only the host with the address in the <b>Remote LAN</b> field can access the module
<b>ETG client encryption</b>	If the TSX ETG 30•• module acts as a client (case ETG -> ETG), you can specify the encryption protocol used for the connection: <ul style="list-style-type: none"> <li>● <b>None:</b> AH</li> <li>● <b>Medium:</b> ESP (DES)</li> <li>● <b>High:</b> ESP (3DES)</li> </ul> <p><b>Note:</b> If you keep this parameter empty, the device acting as a server adapts to the encryption protocol of the client.</p>

**NOTE:** If **VPN** is checked and the table is empty the module does not allow GPRS connections.

## IP Control Parameters

For more information on the IP Control Parameters, refer to PPP Security: IP Control Configuration Page (*see page 174*).

---

## 4.11 Common Modem Features

---

### Overview

This section describes the features common to all modems.

### What's in this Section?

This section contains the following topics:

Topic	Page
Phone List Configuration	216
Callback on Ring	218

## Phone List Configuration

### Access

You can access the phone list services configuration page via the Setup menu of the website.

### View of the Phone Configuration Page

Client: Local PPP Address

Served     
  Fixed     
  TCP/IP address

Phone Number				
	Phone number	Destination	User	Password
1	90493653068	10.10.0.11	USER	****
2	0681936569	10.10.0.12	USER	****
3	2601	10.10.0.13	USER	****
4	0493653726	10.10.0.14	USER	****
5	0492382668	10.10.0.15	USER	****
6				
7				
8				
9				
10				

### Local PPP Address Parameters

Parameters	Action
Served	The gateway sets the PPP address while creating the connection on the remote device.
Fixed	The default address (10.10.0.3) is used.
Use TCP/IP address	The PPP address of the modem interface is the same than the one of the Ethernet interface.

### Remote Device Connection Parameters

Parameters	Values
Input fields	
Phone number	Specify the phone number of the remote device.



---

Parameters	Values
Destination	Specify the IP address or the hostname (if DNS is configured) of the remote device.
User	Specify the Username of the remote device.
Password	Specify the Password of the remote device.

**Edit Button Parameters**

Parameters	Action
Apply	Confirm the input.
Undo	Cancel the current changes.

## Callback on Ring

### Presentation

The callback on ring function enables you to launch a modem connection by calling the module. Then, the module calls a telephone number automatically on a ringing event.

This avoids paying the connection by the caller and uses the module GPRS account.

### Principle

After a delay and a number of rings configured by the user, the module calls back a configured number. The delay allows time for the caller to hang-up.

#### **NOTE:**

- If the callback function is enabled, the module does not answer a call.
- Callback is possible for the TSX ETG 3010/3021/3022 modules if you use their internal modem.
- Depending on the technology and the telephone structure, the number of rings may not be precise.

---

## 4.12 Ending a Connection

---

### Overview

This section describes how to close a TCP/IP or a modem with the TSX ETG 30\*\*.

### What's in this Section?

This section contains the following topics:

Topic	Page
Closing a Connection	220
Managing a Broken Connection	221

## Closing a Connection

A TCP/IP connection can be closed by either:

- the remote station. It sends a TCP/IP connection cutoff to end communication, or
- the TSX ETG 30••. If the maximum number of open connections has been reached, the module closes the oldest connection.

The module sends a report (message rejected) when it closes a connection.

In the case of a modem connection, the module drops the connection if:

- the remote station ends communication and breaks the modem connection,
- the remote device is not authorized to connect,
- the time between two frames defined during configuration elapses,
- the connection time exceeds the time defined during configuration (*see page 157*),
- a module acting as a remote station server attempts to establish a connection to another remote station in client mode via modem,
- an email connection is terminated and a command (override on module command register) is sent by the device,
- it receives a write request from the internal register command of the gateway.
- the Hang-up after email function has been selected.

## Managing a Broken Connection

### Introduction

A connection breakdown may occur in 2 ways:

- disconnection of a network cable (disconnected or damaged cable, etc.),
- disappearance of the remote device (device not operating correctly, power outage, etc.).

The loss of connection is detected after two hours by the Keep Alive request.

If the connection is re-established during this time, communication resumption depends on the way that the connection was broken.

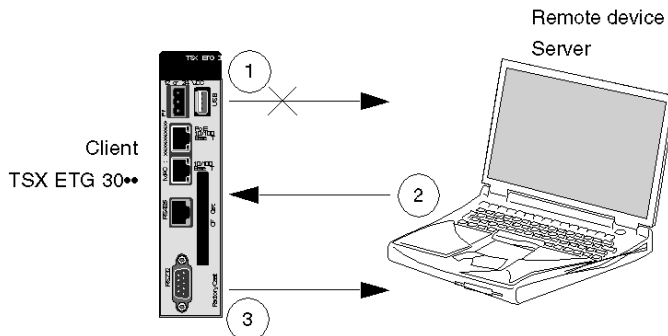
### Cable Reconnection

In this case, the connection was broken due to a network cable, but the two stations are still operational.

When the cable is reconnected, communication between the TSX ETG 30•• module and the remote device is resumed.

### Remote Device as Server

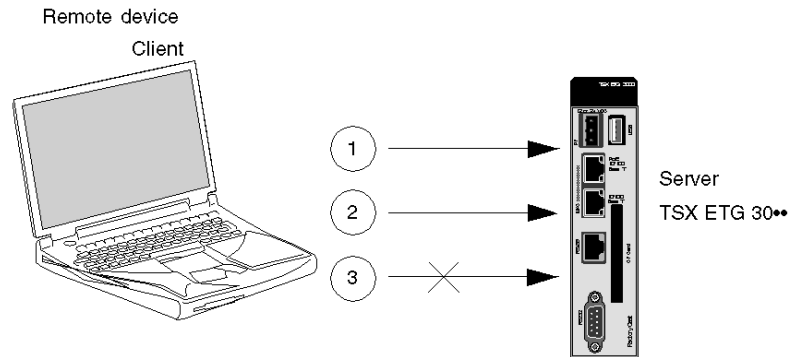
The remote device that disappeared was acting as server.



- 1 The TSX ETG 30•• module (client) still transmits on the old connection (which remains partly open).
- 2 The server receiving data with no associated connection sends a Reset command and closes the old connection.
- 3 The TSX ETG 30•• module (client) opens a new connection.

## Remote Device as Client

The remote device that disappeared was acting as client.



- 1 The client opens a new connection.
- 2 The TSX ETG 30 module (server) receives the request to open a new connection.
- 3 The TSX ETG 30 module (server) closes the old connection (if there is no activity in progress) and authorizes the new connection.

---

## 4.13 Remote Connection Availability: Modem Stand-by Function

---

### Overview

This section describes the remote connection availability feature with the TSX ETG 302• module.

### What's in this Section?

This section contains the following topics:

Topic	Page
Remote Connection Availability: Modem <b>Stand-by</b> Function	224
<b>Stand-by</b> Configuration Page	226

## Remote Connection Availability: Modem Stand-by Function

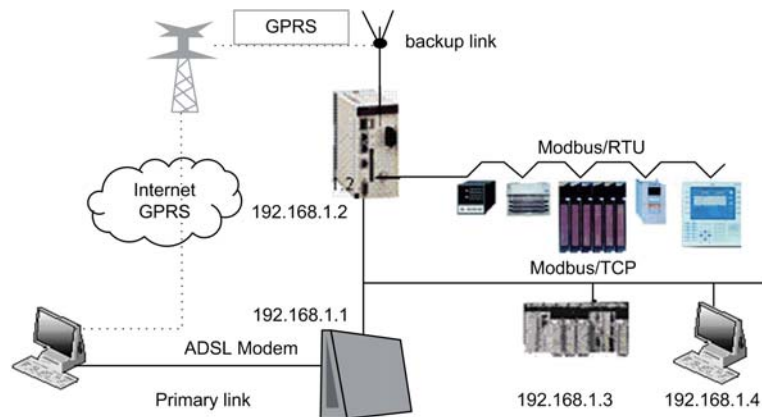
### Overview

This function (available with the TSX ETG 302• module) increases the remote connection availability by providing a “**Stand-by** GPRS connection path” when a communication on the primary DSL link is interrupted. An automatic change-over is performed between the primary DSL link (Ethernet port) and the back-up GPRS modem. The function is configured via the **Setup** page of the embedded website or using the Web Designer software.

### Principles

The service periodically checks the availability of the primary link and connects the TSX ETG 302• to the secondary link in case of loss of communication from the first link. As soon as the primary link is operational, the secondary link is automatically deactivated. This functionality is referred to as automatic primary/backup changeover.

The following diagram shows the overall remote connection availability system architecture:



### Applications Examples

Three typical use cases of the remote connection availability:

- **Continuity of access:**

The PC connected on the Internet uses the primary DSL link to connect to the remote devices. If the primary link is not operational, the TSX ETG 302• automatically starts the GPRS connection and the PC uses this secondary link.

**NOTE:** An automatic changeover from primary to secondary link involves a change of IP address.



- **Continuity of communication:**  
The TSX ETG 302• or a local Ethernet device accessing an equipment on the Internet via the primary link, uses the secondary link when the primary link is not available.
- **Alert:**  
The TSX ETG 302• sends the email using the secondary link when the primary is not operational.

### **Advantages and Limitations**

The remote connection availability presents the following advantages:

- It provides a continuity of access.
- it provides a continuity of communications.

The remote connection availability presents the following limitations:

- The primary link can only be an Ethernet connection.
- The secondary link can only be the GRPS connection from the TSX ETG 302•.

## Stand-by Configuration Page

### Introduction

The **Stand-by** connection function (or remote connection availability) is configured via the **Setup** page from the embedded website or via the Web Designer software.

**NOTE:** Write access is controlled by a security password with the default value **USER**.

### Accessing the Stand-by Configuration Page

Step	Action
1	Click the <b>Setup</b> link on the <b>Home</b> page.
2	Click the <b>Stand-by</b> link on the <b>Setup</b> page.

### View of the Stand-by Configuration Page

Enable

**Connections**

Primary ETHERNET Backup INTERNAL MODEM

**Ethernet Link Monitoring**

Monitoring period 3mn  Switch back to Ethernet

Nb of Retry 0

**Ethernet Link Checking**

IP address ping 92.123.45.10

Http Get address

Ethernet port 1  Ethernet port 2

**Modem Link Monitoring**

GPRS attachment check Monitoring period 1h

### Stand-by Parameters

General parameters

Parameter	Description
<b>Enable</b>	Click this checkbox to activate the remote connection availability service.

## Connections parameters

Parameter	Description
<b>Primary</b>	Name of the primary Ethernet link.
<b>Backup</b>	Name of the secondary GPRS connection link.

## Ethernet link monitoring parameters

Parameter	Description
<b>Monitoring Period</b>	Choose a time period to monitor the availability of the primary Ethernet link.
<b>Number of Retry</b>	Define the number of consecutive times that the primary Ethernet link can lose its connection before switching to the secondary GPRS link.
<b>Switch Back to Ethernet</b>	Click the checkbox to allow the primary Ethernet link to take over the secondary GPRS link as soon as the primary connection is restored. <b>NOTE:</b> The checkbox is unchecked by default. In this case, the secondary link is activated until a manual action is performed to return to the primary Ethernet link.

## Ethernet link checking parameters

Parameter	Description
<b>IP address ping</b>	Click the checkbox to ping the IP address entered in the associated field (dot or symbolic notation possible).
<b>Http get address</b>	Click this checkbox to test an IP address by http (dot or symbolic notation possible).
<b>Ethernet port x</b>	Click this checkbox to test the cable connection on the Ethernet port number 1 or 2.

**NOTE:** All checkboxes from the Ethernet link checking parameters can be combined together. In this case, the Ethernet link is considered as connected only if all checked conditions are working correctly.

## Modem link monitoring parameters

Parameter	Description
<b>GPRS attachment check</b>	Click this checkbox to enable periodic check of the secondary link connection. This is for diagnostic purposes only.
<b>Monitoring period</b>	Choose a time period to check the secondary link connection.

## 4.14 DHCP (FDR) Service

---

### Overview

This section describes the DHCP (FDR) service.

### What's in this Section?

This section contains the following topics:

Topic	Page
DHCP(FDR) Service Presentation	229
TSX ETG 30•• as DHCP Client	230
TSX ETG 30•• as DHCP(FDR) Client	231

---

## DHCP(FDR) Service Presentation

### Presentation

You can configure the Ethernet link parameters of the TSX ETG 30\*\* module either manually or by using automatic configuration protocol called DHCP.

DHCP (*Dynamic Host Configuration Protocol*) is a protocol for booting diskless terminals or stations using centralized management of network parameters.

The main purpose is to provide an IP address or a configuration for a station booting on the network.

The module acts a DHCP client.

The DHCP server can be a Premium fitted with a TSX ETY module or a Quantum fitted with an NOE module.

**NOTE:** The DHCP service only configures the Ethernet parameters. This service is not available for a modem connection (RS232 link).

## TSX ETG 30\*\* as DHCP Client

### Principle

The principle used is as follows:

- The TSX ETG 30\*\* module requests an IP configuration (IP address, subnet mask, gateway) from a DHCP server by means of its MAC address.
- The DHCP server uses a MAC address/IP Configuration correspondence table to return the IP configuration to the TSX ETG 30\*\*.

**NOTE:** In order to use the DHCP service, you must configure your DHCP server.

### Initial Startup

Behavior of the TSX ETG 30\*\* module on initial startup:

The module sends a configuration request to the server:

- If the module is not recognized, it starts up with its default IP configuration (factory-set).
- If the DHCP server sends an IP configuration, the TSX ETG 30\*\* uses it but **without storing it in Flash memory**.

### Subsequent Startups

Behavior of the module on subsequent startups:

The TSX ETG 30\*\* module sends a configuration request to the server:

- If the DHCP server sends a configuration, the module uses it.
- If the DHCP server does not respond within about 5 minutes, the module switches to downgraded operating mode and uses the IP configuration stored in the Flash memory (the factory-set default configuration or one saved in the Flash memory).

## TSX ETG 30•• as DHCP(FDR) Client

### Presentation

This service allows the automatic retrieval of configurations by a TSX ETG 30•• module connected to an Ethernet segment.

The FDR service can provide the following parameters:

- Modbus/RTU configuration
- IP filtering configuration
- modem configuration
- phone list configuration
- DNS configuration
- NTP configuration
- SNMP configuration

**NOTE:** The FDR server does not backup security passwords, Services configuration, DataTables pages, GraphicScreens pages and custom web pages. As the FDR service does not store passwords, the module uses the default passwords.

The FDR function uses a combination of the DHCP and FTP protocols.

The TSX ETG 30•• uses a name (Device Name) to obtain its configuration from the server. The **Device Name** is a string of characters (maximum of 15) associated with the module that must be **unique** within the architecture.

The TSX ETG 30•• is therefore able to configure itself automatically using a parameters file previously saved in the DHCP server, for example a Premium TSX ETY 5103.

**NOTE:** In order to use the FDR service, you must identify the client device by its Device Name. When configured as FDR server, the TSX ETY 4103/5103 can manage a maximum of 16 TSX ETG 30•• clients.

### Operation

The operating principle of the FDR service is as follows:

1	A TSX ETG 30•• connects to the network with a configured name (Device Name).
2	The TSX ETG 30•• sends a DHCP request, indicating its associated Device Name.
3	If the Device Name is included in the DHCP server's configuration table, the server sends the following parameters to the module: <ul style="list-style-type: none"> <li>● the IP address that must be used</li> <li>● the IP address of the FTP server</li> <li>● the location of the configuration file in the FTP server</li> </ul>
4	The TSX ETG 30•• then accesses the FTP server to upload or download the configuration file to or from the FTP server. The configuration file is identified by a name composed of the Device Name with a <i>.prm</i> extension.

## Initial Startup

Behavior of the TSX ETG 30\*\* module on initial startup:

The TSX ETG 30\*\* module sends a configuration request to the server:

- if the module is not recognized, it starts up with the default configuration (factory-set) after approximately 1 minutes.
- if the module is recognized, the TSX ETG 30\*\* starts up with the configuration provided by the FDR service and stores it in its Flash memory (excepting the IP configuration).

## Subsequent Startups

Behavior of the module on subsequent startups:

The TSX ETG 30\*\* module sends a configuration request to the server:

- if the module is recognized, the TSX ETG 30\*\* starts up with the configuration provided by the FDR service and stores it in its Flash memory (excepting the IP configuration).
- if the module is not recognized, it starts up after approximately 5 minutes with the default configuration saved in the Flash memory.



---

## 4.15 DNS Service

---

### Overview

This section describes the DNS service of the TSX ETG 30•• module.

### What's in this Section?

This section contains the following topics:

Topic	Page
DNS Service Presentation	234
DNS Service Configuration	236

## DNS Service Presentation

### Definition

The DNS (*Domain Name System*) is a distributed database that stores and associates many types of information with domain names, but most importantly, it translates domain names (computer hostnames) to IP addresses.

### Presentation

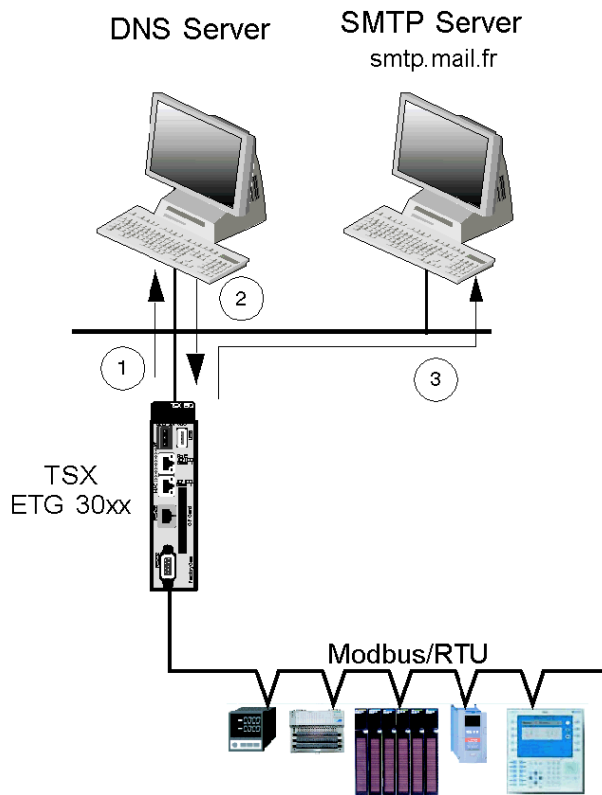
DNS is mandatory for sending email. You must configure the DNS service in order to reach a device over the Internet by its name. DNS uses a client/server architecture. The TSX ETG 30•• acts as a client (*see page 236*).

**NOTE:** The DNS service can be configured by using either Web Designer for TSX ETG 30•• or the embedded website.

### DNS Architecture

The DNS server is located on the same Ethernet segment than the TSX ETG 30••. The DNS service is used to translate the name of the SMTP server ("smtp.mail.fr"), the name of the NTP server and the name of the SNMP server.

The following figure shows a connection to the DNS server:



The following table details the connection process for sending an email:

Step	Description
1	The module requests the IP address of the SMTP server from the DNS server by sending the name of the SMTP server ('smtp.mail.fr').
2	The DNS server translates the name of the SMTP server into its corresponding IP address. It hands the result back to the module.
3	The module sends its email request to the SMTP server using the IP address of the SMTP server.

## DNS Service Configuration

### Presentation

You can access DNS configuration page via the Setup menu of the website.

### View of the DNS Configuration Page

**DNS Server**

Domain Name Server  .  .  .

**Domain Name**

Domain Name

### Parameters

Parameters	Values
<b>Input fields</b>	
Domain Name Server	Set the IP address of the DNS server.
Domain Name	Specify the domain name of your network for the module.
<b>Edit buttons</b>	
Apply	Confirm the input.
Undo	Cancel the current changes.

---

## 4.16 NTP Service

---

### Overview

This section describes the NTP service provided by the TSX ETG 30•• module.

### What's in this Section?

This section contains the following topics:

Topic	Page
NTP Service	238
NTP Service Configuration	239

## NTP Service

### Synchronizing System Clocks

The *NTP (Network Time Protocol)* is a protocol used for synchronizing the clocks of computer systems.

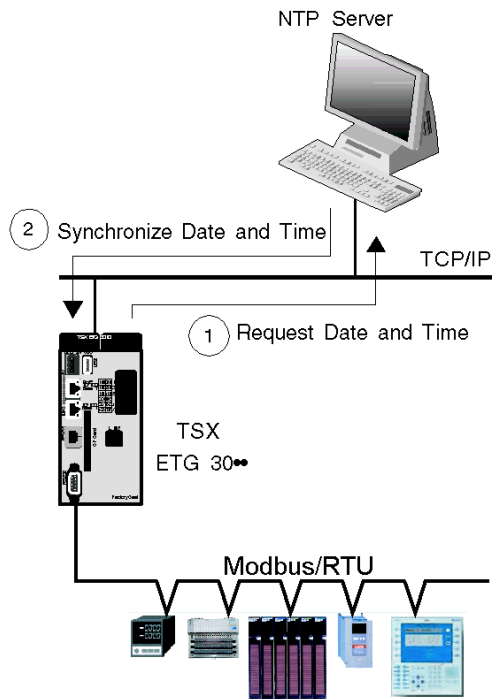
### Client/Server Architecture

NTP uses a client/server architecture. The TSX ETG 30\*\* acts as a client. Once configured, the NTP server synchronizes the clock of the TSX ETG 30\*\* with its own clock (over TCP/IP).

If you choose to use the NTP service, real time clock is no longer available. At the first connection to the NTP server, the local date and time of the TSX ETG 30\*\* module is set to undefined.

**NOTE:** The NTP service can be configured by using either Web Designer for TSX ETG 30\*\* or the embedded website.

The following illustration shows a connection to a NTP server:



## NTP Service Configuration

### Access

You can access the NTP configuration page via the Setup menu of the website.

### View of the NTP Configuration Page

The screenshot displays the NTP configuration interface. It is divided into two main sections: 'NTP Server Parameters' and 'Time Zone'. The 'NTP Server Parameters' section includes three input fields: 'Primary NTP server', 'Secondary NTP server', and 'Polling period (s)'. The 'Time Zone' section features a dropdown menu for selecting a time zone and a checked checkbox labeled 'Automatically adjust clock for daylight saving change'. At the bottom of the form are two buttons: 'Apply' and 'Undo'.

**NOTE:** The module must be initialized (rebooted) in order for new configuration parameters to take effect.

### Configuration Parameters

Parameters	Action
Primary NTP server	Set the IP address of the primary NTP server.
Secondary NTP server	Set the IP address of the secondary NTP server.
Polling period	Specify the polling period for updating time.
Time zone	Specify the local time zone (GMT, GMT+1, ...).
Automatically adjust clock for daylight saving change	Define if daylight saving time is managed.
<b>Edit buttons</b>	
Apply	Confirm the input.
Undo	Cancel the current changes.

## 4.17 SNMP server

---

### Overview

This section describes the SNMP service of the TSX ETG 30\*\*.

### What's in this Section?

This section contains the following topics:

Topic	Page
SNMP Communication in UDP/IP	241
SNMP Service Configuration	244



## SNMP Communication in UDP/IP

### Elements of the SNMP Architecture

The SNMP defines network management solutions in terms of protocol and supervised data exchange.

The SNMP architecture is based on the following key elements:

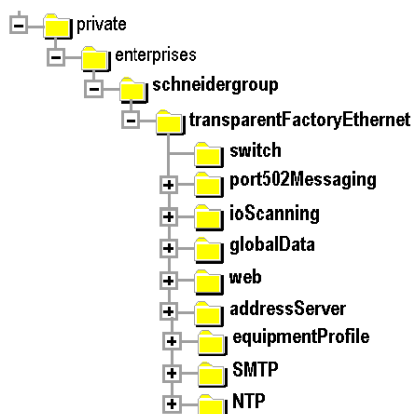
- the **Manager** is used to supervise all or part of the network,
- one or more **Agents**. Each device being supervised has a software module called an **Agent** used by the SNMP protocol,
- an MIB is a database or collection of objects updated by the agents.

The SNMP agent service is implemented on the TSX ETG 30\*\* module. The SNMP protocol enables a Manager to access standard MIB objects in the TSX ETG 30\*\* module.

The MIB-II is used to manage TCP/IP communication layers.

The MIB Ethernet Transparent Factory allows a manager to access data on the messaging service on port 502.

View of the Ethernet Transparent Factory MIB tree via a Manager:



The source file of the Ethernet Transparent Factory MIB is available on the TSX ETG 30\*\* module. It can be downloaded with a Web browser by clicking the MIB Upload link on the Diagnostics home page (*see page 363*). The TSX ETG 30\*\* module complies with MIB version 1.4. This file can be compiled with most commercial SNMP Managers.

## The SNMP Protocol

The SNMP protocol defines 5 types of message between agent and manager: these messages are stored in UDP datagrams.

Messages from the manager to an agent:

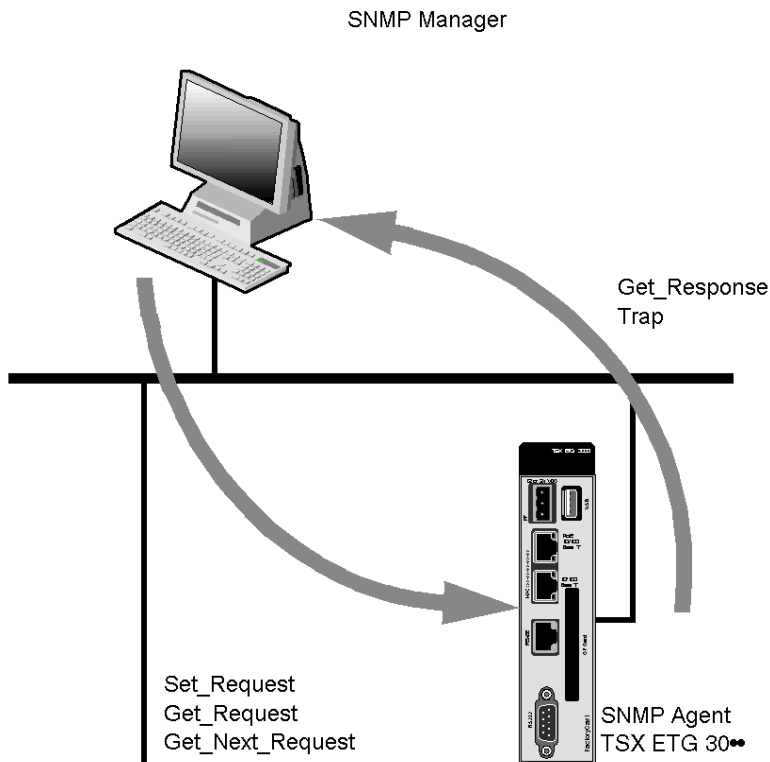
- **Get\_Request**: message used to obtain the value of one or more variables
- **Get\_Next\_Request**: used to obtain the value of subsequent variables
- **Set\_Request**: used to position the value of a variable

Messages from an agent to the manager:

- **Get\_Response**: used by the agent to return the variable value requested
- **Trap**: used by the agent to signal an event to the manager (unauthorized access attempt or rebooting of the device)

## SNMP Services

The SNMP manager sends write or read requests (Set\_Request, Get\_Request, Get\_Next\_Request, etc.) for objects defined in the SNMP MIB-II, and the SNMP agent for the TSX ETG 30\*\* module responds.



The module's SNMP agent sends events (Traps) to the Manager. The following System Traps are managed:

- Coldstart Trap: the event is sent when the module is powered up
- Authentication Failure Trap: event sent after an authentication anomaly. The **Community Name** field in the message received is different from that configured on the module. This trap can be validated when the TSX ETG 30\*\* module is configured.

## SNMP Service Configuration

### Access

In order to use the module as an SNMP agent, you must set the configuration parameters.

You can access the SNMP configuration page via the **Setup** menu of the website.

**NOTE:** Only ASCII 7-bit characters can be used in the character string input fields.

### View of the SNMP Page

IP address managers	
Manager 1	<input type="text"/>
Manager 2	<input type="text"/>
Agent	
System location	<input type="text"/>
System contact	<input type="text"/>
Community names	
Read-Write	<input type="text"/>
Read-Only	<input type="text"/>
Trap	<input type="text"/>
Enable authentication failure trap	<input checked="" type="checkbox"/>
<input type="button" value="Apply"/> <input type="button" value="Undo"/>	

### Configuration Parameters

Parameters	Values
<b>Input fields</b>	
<b>Enable authentication failure trap</b>	This allows the agent to notify the administrator that the request has been refused due to an inoperative authentication (community name configured in the administrator differs from that configured in the agent).
<b>Edit buttons</b>	
<b>Apply</b>	Confirm the input.
<b>Undo</b>	Cancel the current changes.

---

### IP Address Managers Parameters

Use the **IP address managers** area to enter the IP addresses of the SNMP administrators. The modules allow a maximum of two administrators.

These addresses are used for transmitting events (TRAP).

### Agent Parameters

Use the **Agent** area to locate and identify an agent, from the SNMP administrator. It comprises two fields:

- The **System Location** field: indicates the physical location of the device (string of 32 characters maximum).
- The **System Contact** field: indicates the person to contact for management of the device and the way to contact them (string of 32 characters maximum).

### Community Names Parameters

Use the **Community Names** area to define a community name for the Set, Get and Trap service families. It comprises three fields:

- The **read/write (Set)** field: defines the name for the Set service (string of 16 characters maximum). The default value is **Public**.
- The **read only (Get)** field: defines the community name for the Get service (string of 16 characters maximum). The default value is **Public**.
- The **Trap** field: defines the community name for the Trap service (string of 16 characters maximum). The default value is **Public**.

## 4.18 Device Explorer

---

### Overview

The Device Explorer feature enables you to display status and diagnostic information of Ethernet and Modbus devices connected to the TSX ETG 30\*\*.

### What's in this Section?

This section contains the following topics:

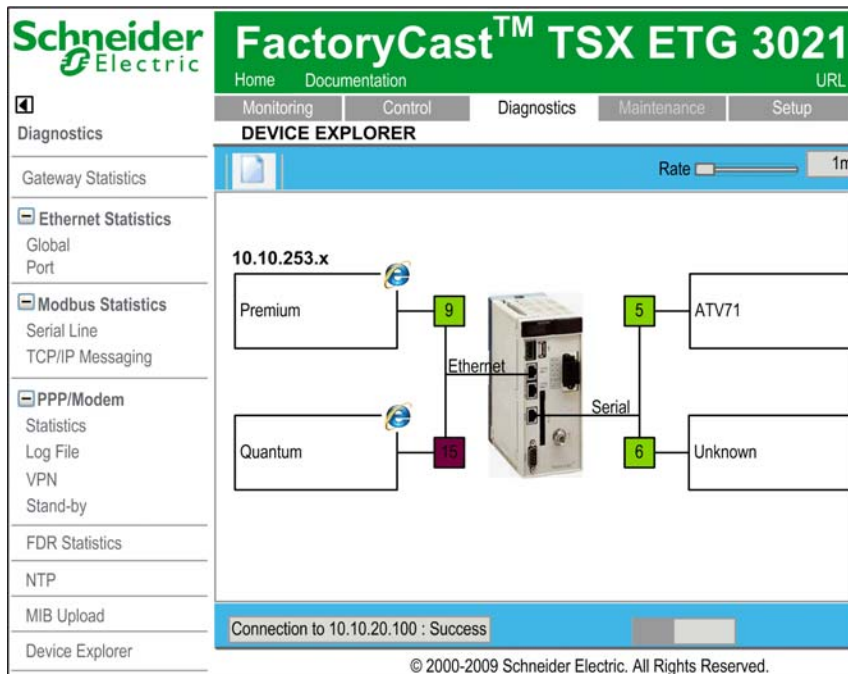
Topic	Page
Device Explorer Presentation	247
<b>Device Explorer</b> Configuration	249

## Device Explorer Presentation

### Introduction

The TSX ETG 30\*\* integrates a proxy function. This feature enables you to access status and diagnostic information of devices connected to the module.

### Device Explorer Page



In animation mode, the TSX ETG 30\*\* sends an identification request (43/14 Modbus request) to each device of the network (cyclic check).

The device appears as not operating correctly (red instead of green) in the animation page, if:

- it does not answer to the identification request,
- the response to the animation request is negative.

**NOTE:** Modbus devices that are not present on the link generate Modbus time-outs. It decreases the performance of the variables polling especially if you are using the **Automatic Discovery** features.

## Accessing the Device

By double clicking the Internet Explorer icon on the rectangles symbolizing the devices, the website of the device appears in a new tab of the Web browser.

**NOTE:**

- Your Web browser should manage tabs.
- The device must operate properly.

If the device is connected to the Ethernet network, the module launches a new page with the IP address of the device as URL.




## Device Explorer Configuration

### Presentation

In order to use the **Device Explorer** feature, you must first define your network configuration.

### Accessing the Device Explorer Configuration Page

Step	Action
1	Click the <b>Diagnostics</b> link on the <b>Home</b> page.
2	Click the <b>Device Explorer</b> link on the <b>Diagnostics</b> page.
3	Click  to access the configuration.

### View of the Device Explorer Configuration Page













The screenshot displays the Schneider Electric FactoryCast™ TSX ETG 3021 web interface. The top navigation bar includes links for Home, Documentation, and URL. Below this, there are tabs for Monitoring, Control, Diagnostics, Maintenance, and Setup. The main content area is titled "DEVICE EXPLORER" and features a toolbar with various icons. A central image shows a physical device. At the bottom of the interface, a status bar indicates "Animation Started". The footer contains the copyright notice: © 2000-2009 Schneider Electric. All Rights Reserved.

## Description of the Toolbar



The toolbar enables you to access the main functions of the configuration page directly by clicking its icons.

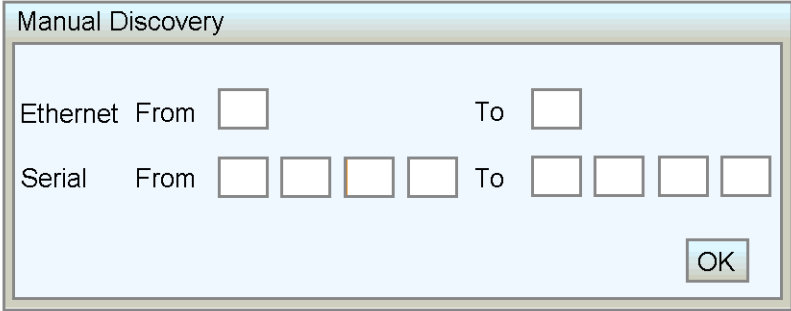


Button	Function
	Access the edition mode.
	Save the current network configuration.
	Automatic discovery of the whole network.
	Manual discovery of the network. You must specify the range of addresses that will be scanned.
	Delete the selected device.
	Add a new device on the Ethernet link.
	Add a new device on the serial link.
	Copy the selected object to the clipboard.
	Paste the clipboard.
	Launch the animation mode.

## Discovery of the Devices

The Device Explorer applet enables you to define how to scan the network for discovery of the connected devices:

- By clicking , the whole network will be scanned. The scan of your Ethernet network starts at IP address = x.x.x.1 and it stops at IP address = x.x.x.254. It starts at Modbus address = 1 and it stops at Modbus address = 31 for your Modbus network.
- By clicking , the scan only concerns a specified range of addresses. The **Manual Discovery** window appears in which you can define the range of addresses to be scanned.



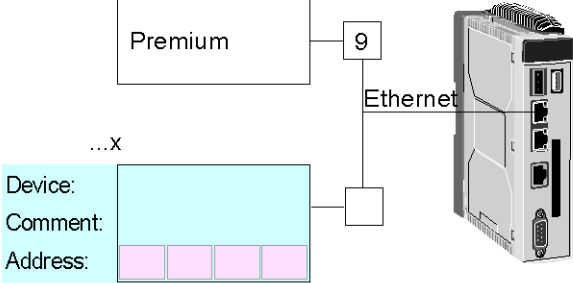


The image shows a dialog box titled "Manual Discovery". It contains two rows of input fields. The first row is labeled "Ethernet" and has "From" followed by a single text box and "To" followed by another single text box. The second row is labeled "Serial" and has "From" followed by four text boxes and "To" followed by four text boxes. An "OK" button is located in the bottom right corner of the dialog box.

### NOTE:


- The automatic discovery is a process that can take time: around 1 s for an Ethernet device and 3 s for a Modbus device.
- The box in the bottom corner informs you on the status of the scan. It also displays the address that is currently scanned.

## Manual Edition of the Devices

Step	Action
1	<ul style="list-style-type: none"> <li>● Click  to add a device on the Ethernet link, or</li> <li>● Click  to add a device on the Modbus link.</li> </ul> <p><b>Result:</b> an empty box representing the device appears.</p> <p>10.10.253.x</p> 
2	You can add a comment on the device in the <b>Comment</b> area.
3	Specify the address of device in the <b>Address</b> area (mandatory).
4	Click somewhere else in the window to validate your change.

### Modifying and Removing a Device

To modify the properties of a device, simply double click on it.

To remove a device, select it by clicking on it and click  .

---

## 4.19 FTP Service

---

### FTP

#### Introduction

The FTP service offers the ability to connect a computer to a TSX ETG 30\*\* or to connect two TSX ETG 30\*\* modules together over the Internet. The user can send and/or access files between either a computer and an TSX ETG 30\*\* or between two modules. File commands can also be performed via the FTP service.

#### Characteristics

The FTP service included with the TSX ETG 30\*\* modules offers both server and client capabilities. For example, the CSV files generated by the datalogging service (*see page 302*) can be sent or accessed with the FTP service. A FTP client can log on to the module with a user name and password.

**NOTE:** The default user name is **wsupgrade**, and the default password is **wsupgrade**. Both the user name and password are case sensitive.

The FTP Server allows the customer to upload or download files, including:

- Datalogging files (.csv),
- Services files (.jar),
- Files of the Data Editor, Graphic Editor and Graphic Viewer (.zip),
- Custom HTML Web pages, applets.

The FTP server is compatible with Internet Explorer 6 (minimum) and other freeware FTP client.

## 4.20 Power Over Ethernet

---

### Power Over Ethernet

#### Presentation

*Power over Ethernet (PoE)* technology describes a system to transmit electrical power, along with data, to remote devices over standard twisted-pair cable in an Ethernet network.

**NOTE:** Power over Ethernet is only available with the TSX ETG 3000 module.

#### **DANGER**

##### **HAZARD OF ELECTRIC SHOCK, EXPLOSION OR ARC FLASH**

- Remove the Ethernet cable that provides PoE (only for TSX ETG 3000 modules) to disconnect all power before installing or removing any hardware and cables.
- Always use a properly rated voltage sensing device to confirm power is off.
- Always use suitable insulation equipment when inserting or removing all or part of this equipment.

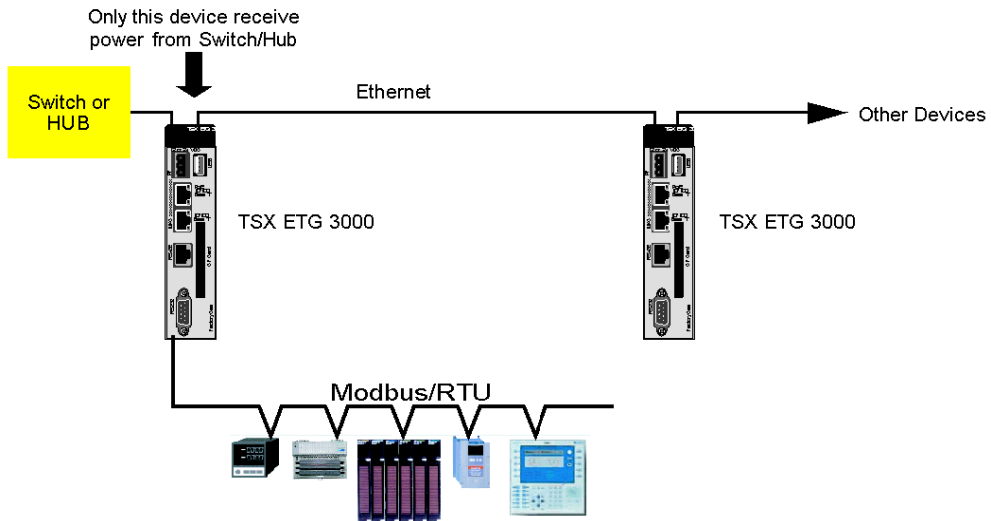
**Failure to follow these instructions will result in death or serious injury.**

#### PoE Topology

You can power the TSX ETG 3000 module using PoE. You can connect it to a powered hub or switch, in this case, you do not need to connect the module to an external power supply. The powered hub or switch must comply to the 802.3af standard.

The Eth1 port of the TSX ETG 3000 module (the highest one) supports Power over Ethernet 802.3af standard. The reference of the PoE injector is TCSEAV0100.

The figure below shows an example of PoE topology:



## 4.21 Daisy Chain

### Daisy Chain

#### Presentation

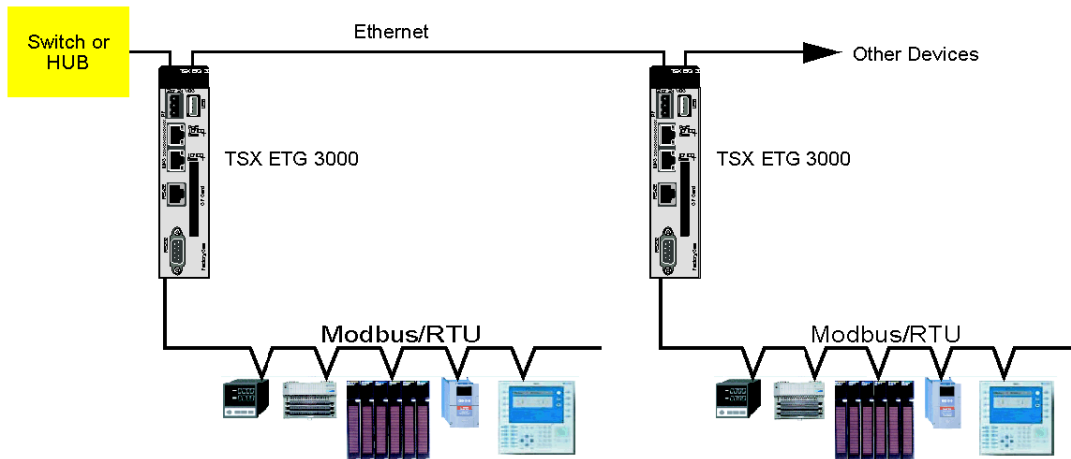
A daisy chain is a wiring scheme in which, for example, device A is wired to device B, device B is wired to device C, device C is wired to device D, et cetera. Connections do not form webs (in the preceding example, device C cannot be directly connected to device A), nor do they loop back from the last device to the first. In case of loop back, the Ethernet network does not work.

#### Daisy Chain Topology

The 2 Ethernet ports of the TSX ETG 30\*\* enables the daisy chain feature. You can connect as many devices as you want in a chain.

**NOTE:** The daisy chain feature does not propagate PoE (*see page 254*).

The illustration below shows you an example of daisy chain topology:





---

# Embedded HMI Services

# 5

---

## Overview

This chapter describes the embedded HMI services offered by the TSX ETG 30\*\* module.

## What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
5.1	Overview of the Embedded HMI Services	258
5.2	Device Service	266
5.3	Email and SMS Service	276
5.4	Calculation Service	285
5.5	Database Service	293
5.6	Datalogging Service	301
5.7	Active Pages Service	315
5.8	Recipe Service	322
5.9	Recipe Database Service	330

## 5.1 Overview of the Embedded HMI Services

---

### Overview

This section provides a general presentation of the embedded HMI services offered by the TSX ETG 30\*\* module.

### What's in this Section?

This section contains the following topics:

Topic	Page
Device and HMI Services	259
Service Status Overview	261

---

## Device and HMI Services

### Presentation

The TSX ETG 30•• modules can perform internal standalone processing such as HMI functions. HMI functions also called "services" are executed inside the module itself independently of any other processing done at SCADA level.

The following HMI functions are available:

- **Device Services:** for configuring devices data acquisition (device variables polling).
- **Calculation:** perform calculations scripts for custom data processing.
- **Email:** for alarm or report notification via email or SMS.
- **Datalogging:** automatic archiving of application information such as measures, events, alarms, device status, etc, to CSV log files into TSX ETG 30•• module memory (Flash memory, CF card, USB memory key).
- **Database connection:** automatic archiving of application information into an external relational database (SQL Server, Oracle, MySql).
- **Active pages:** creation of custom dynamic HTML reports.
- **Recipe:** automatic management of recipe data files to initialize a group of variables set at predetermined values when a specified event occurs.

Web Designer for TSX ETG 30•• software is used to configure the HMI services. You only need to configure the parameters for each service; no programming is required.

The software provides a simulation mode to simplify test and debug of the application without being connected to an TSX ETG 30•• module.

### Flexible Web HMI Solution

These embedded HMI services provide a robust and reliable solution that can be useful for the following use cases:

- Data concentrator front-end with Web and HMI portal features.
- A standalone Web based HMI running inside the TSX ETG 30•• module when a conventional HMI or SCADA solution is not required. In this case, it provides a flexible cost effective multi user HMI solution using a simple Web browser interface for remote monitoring.
- In combination with a conventional HMI or SCADA solution, they make easier remote monitoring and control.
- A backup solution for SCADA applications. It enables backup of some critical SCADA functions such as data acquisition, datalogging, alarming, reporting, email, etc.

** WARNING**

**UNINTENDED EQUIPMENT OPERATION**

- Restrict access to the embedded server by configuring passwords.
- Carefully select the symbols and direct addresses you authorize to be modified online.
- Do not authorize online modification of variables of critical nature concerning human and material integrity.

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

**HMI Services Limitations**

Device and HMI services have the following limitations:

- Device service: 31 devices over serial line, 5 devices over Ethernet, 1 for the TSX ETG 30 server itself.
- Other HMI services: 6 services in total.
- Variables:
  - a maximum of 1000 variables per application.
  - a maximum of 100 internal variables per application.

## Service Status Overview

### Introduction

The main purpose of this feature is to provide status information on each service for diagnostic/debug purposes.

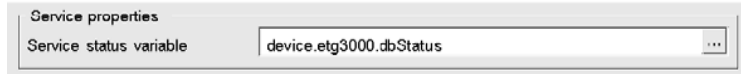
By default, each service has an internal status variable named as follows: "service type.service name.sstatus" (e.g. calculation.device1.sstatus). The status values of each service (device, email, datalogging, etc.) are copied to the internal system or event of the TSX ETG 30\*\* module. This status variable can be used to send warning information via email, to trigger archiving to a database or for any other action useful for improving runtime operating modes.


Example:

To send an email in the event of an anomaly with the database service, you must:

- declare a persistent variable with Web Designer for TSX ETG 30\*\* and assign it to a device address (integer, for example), then import this variable into the Device service,
- define the persistent variable to trigger an email in the email service.

The **Service properties** configuration screens includes a **Service status variable** field as shown below:



Simply click  to select a status variable which can be configured for the service (INT, UINT, SHORT and USHORT type).

The status values are:

- 0 for a service that is stopped,
- 1 for a service that is running,
- $\geq 2$  for a service on which an anomaly has occurred.

The detected error codes are assigned in ascending order of severity.

If the status value gives a detected error code and then a more important event occurs, the status value of the service will change to inform you that the new event (which in this case is more important than the first) has occurred.

When a new action is operational (log, creation of a table, etc.), the service is assigned the value 1 to inform you that the service is operating correctly.

Each service has its own detected error codes.

### Device Service Status Values

Value	Comment
0	The service is stopped.
1	The service is running.
2	The PLC application's signature is different from the one contained in the FactoryCast HMI application for the device service.

**NOTE:** Value 2 will only be accessible from the service's internal status variable. In this case, it is indeed not possible to write in a variable.

### Email and SMS Service Values

Value	Comment
0	The service is in stop mode.
1	The service is in run mode.
10	e-mail refused by SMTP server.
11	e-mail not sent.
12	No more connections or queue saturated.

The highest values have the priority. The service value 12 has the priority but becomes 11 subsequently. The service value 11 disappears once the email has been sent correctly.

### Database Service Error Codes

Value	Comment
0	The service is in stop mode.
1	The service is in run mode.
20	Log in table delayed.
21	No access to table.
22	Table no longer exists.
23	Database connection interruption.
50	The database buffer is full, and new events are no longer stored. <b>NOTE:</b> Reconnecting the database or Ethernet cable is required to send values to the database and store new incoming values.

Value	Comment
100	Incorrect database configuration.
200	Database undefined.
1000	Detected fatal error with database.
1000	Detected critical error with database.

**NOTE:**

The database service error code 20 should be considered as a notification, not an alarm. It may occur when:

- the service is started (value not ready),
- the PLC has not had the time to send the data as part of the request,
- the signature of the PLC application is different to that of the application.

The highest values have the priority. The database service error code 23 has priority over the database service error code 21. When the database service error code 23 has been corrected, the value 21 can then be sent to the variable declared for the status of the service.

**Datalogging Service Codes**

Value	Comment
0	OK
2	The current file '.csv' is renamed '.cs0'.
10	The module cannot reach a variable that should be logged because: <ul style="list-style-type: none"> <li>● the variable does not appear in the namespace.</li> <li>● the variable is not write enabled.</li> <li>● the preceding value has not been updated yet.</li> </ul>
11	FTP transfer interruption.
12	The URL specified for FTP cannot be accessed.
13	The internal flash is full.
14	The internal RAM is full.
15	The module cannot write on the media specified.
16	The module cannot access the namespace.
17	The maximum number of tables has been reached (10 maximum) in your <i>.xml</i> file.
18	The maximum number of variables has been reached in a table.
19	The service is empty, no table defined.

**NOTE:**

- The datalogging service code 2 should be considered as a status.
- The highest values have the priority. The datalogging service code 17 has priority over the datalogging service code 12. When the datalogging service code 17 has been corrected, the datalogging service code 12 can then be sent to the variable declared for the status of the database service
- The values of the table status variable are identical to the values of the datalogging service status variable.

**Recipe Service Codes**

Value	Comment
0	The service is in stop mode.
1	The service is in run mode.
60	Anomaly in recipe configuration.
61	Anomaly when writing the value.
62	Communication interruption.
63	Anomaly in recipe file.

**RecipeDB Service Codes**

Value	Comment
0	The service is in stop mode.
1	The service is in run mode.
20	Log in table delayed.
21	No access to table.
22	Table no longer exists.
23	Database connection anomaly.
24	Incorrect service configuration.
25	Fatal detected error.

**NOTE:**

The value 20 should be considered to be a system message, not an alarm. It may occur when:

- The service is started (value not ready).
- The PLC has not had the time to send the data as part of the request.
- The signature of the PLC application is different to that of the FactoryCast HMI application.



---

The high values take priority. The code 23 has priority over code 21. When code 23 has been corrected, the value 21 can then be sent to the variable declared for the status of the service.

### Codes for Other Services

The other services (calculation, active pages) do not have any specific codes. The code only specifies whether the service is stopped (0) or running (1).

<b>Value</b>	<b>Code</b>	<b>Comment</b>
0	<b>Stop</b>	The service is in stop mode.
1	<b>Run</b>	The service is in run mode.

## 5.2 Device Service

---

### Overview

This section describes the Device service.

### What's in this Section?

This section contains the following topics:

Topic	Page
Device Service Presentation	267
Naming Variables	269
Device Service Configuration	270

## Device Service Presentation

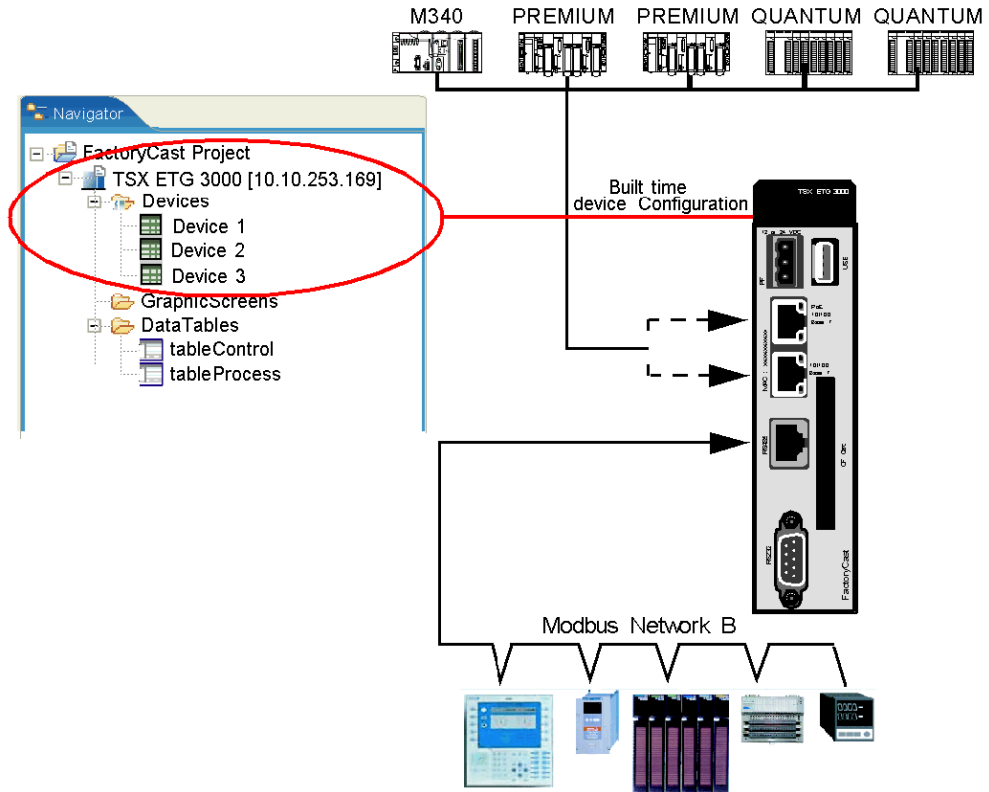
### Presentation

The Device services enables you to configure data acquisition from devices connected to the gateway either over serial line or Ethernet network. The TSX ETG 30\*\* module will perform data acquisition in runtime. Only 1 device service can be configured per connected device. It consists of a list of related register variables to be exchanged (in read or write) with the devices.

Configuration is done using Web Designer for TSX ETG 30\*\*. It generates a "Namespace file" representing the complete list of device variables resulting from the overall device services. Those variables can be used within the Web Designer application in any of the HMI services.

**NOTE:** The Device service is configured using Web Designer for TSX ETG 30\*\*. Before creating a new service, you must create a Web Designer for TSX ETG 30\*\* project (refer to the *Web Designer for TSX ETG 30\*\* User Manual*).

The following figure describes the Device service process:



### Limitation

The number of variables stored in the module is limited to 1,000 for all device services.

The maximum number of device service is:

- 31 devices over serial line
- 5 devices over Ethernet
- 1 for the TSX ETG 3000 server itself

## Naming Variables

### Introduction

Application variables come either from the connected devices or from the calculation service. They can be accessed by any other HMI services. Rules for naming variables must be applied in order to have variable names uniqueness within the application.

### Naming Rule

Variable names are composed of 3 parts separated by ‘.’:

- The type of service (Device or Calculation).
- The name of the service.
- The name of the variable.
- Device.device1.speed: variable coming from the device service named speed and coming from the service called device.
- Calculation.move.valveopened: variable coming from the calculation service named valveopened and coming from the service called move.

### Variables Coming From Internal Registers

It is also possible to access variables from internal registers of the TSX ETG 30\*\*. This can be achieved using symbolic names. The internal registers tables (*see page 102*) details all internal variables and their symbolic names.

## Device Service Configuration

### General

The Device service enables you to create symbols associated with variables coming from devices connected to the module over serial line, Ethernet or internal registers. These variables are associated with symbols that will be used by other services (email, calculation, database, ...).

**NOTE:** Select the **Persistent** mode for variables that you want to use with other services.

Unauthorized or incorrect modifications made to symbols and direct addresses may have undesirable or even change the behavior of your application in ways that may be undesirable or even hazardous.

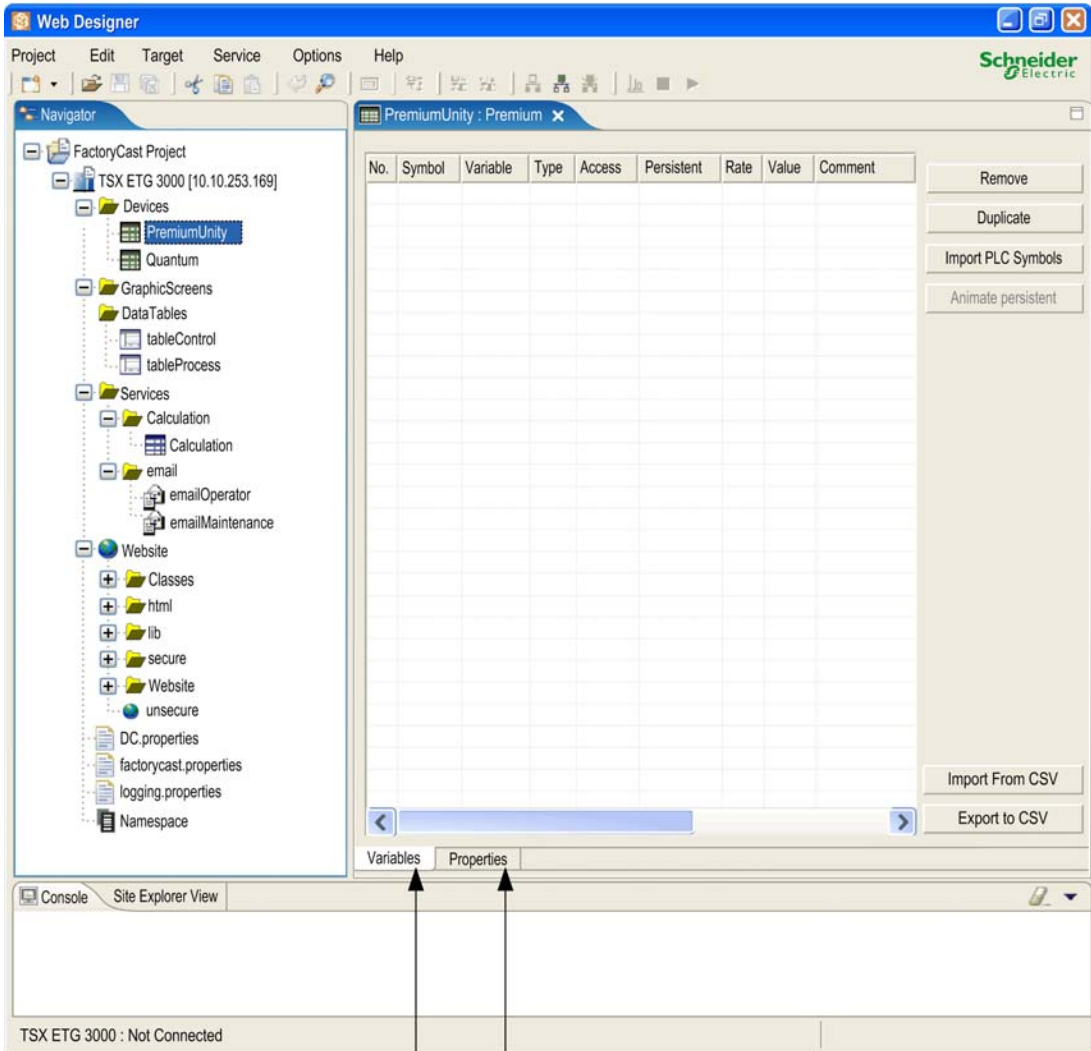
### **WARNING**

#### **UNINTENDED EQUIPMENT OPERATION**

- Carefully select the symbols and direct addresses you authorize to be modified online.
- Do not authorize online modification of variables of critical nature concerning human and material integrity.

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

Properties configuration screen for the device service:



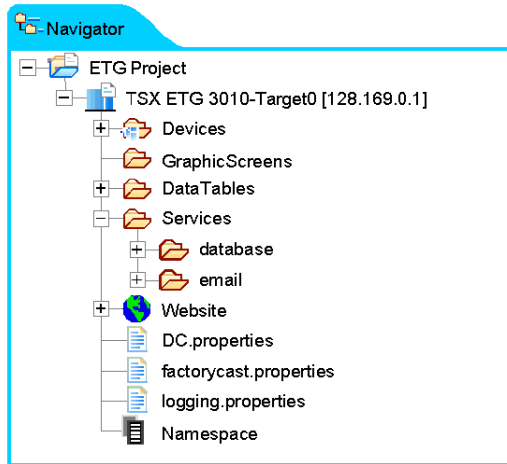
- 1 Use this button to configure the service variables.
- 2 Use this button to configure the service properties.

### Creating a Device Service

To create a new device service:

- double click the name of your device in the browser, or
- right click the name of your device in the browser, then click **Edit**.

View of the Web Designer for TSX ETG 30\*\* browser:



### Configuring the Properties

Properties configuration screen for the device service:

Address	<input type="text"/>
Protocol	Modbus
Driver	BUSX_ASYNC
Base Rate	1000
Comment	<input type="text"/>
Service status variable	<input type="text"/>

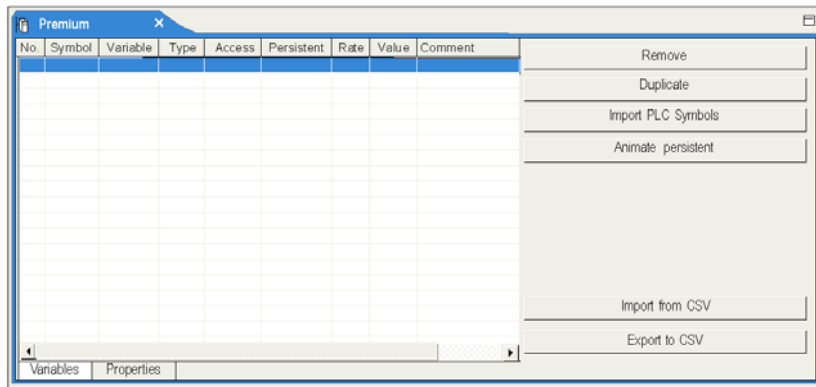


This table describes the various fields that make up the properties configuration screen:

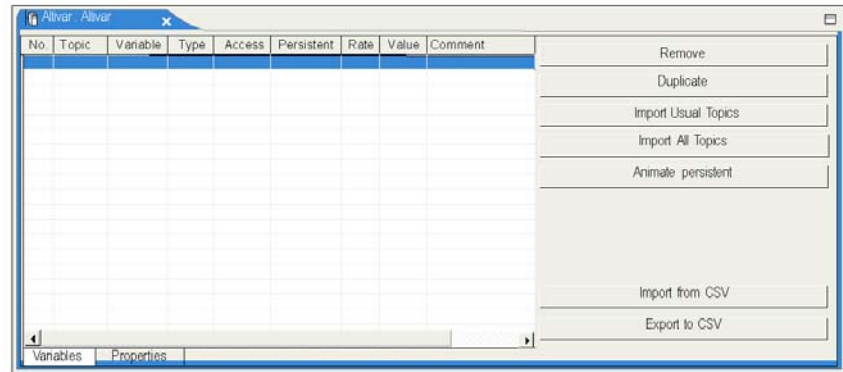
Fields	Function
<b>Address</b>	Variables stored in the module may be associated with remote devices. The Address field is used to define the Modbus or IP address of the remote device.
<b>Protocol</b>	The protocol is determined by the project type (shaded), i.e. Modbus (for Quantum PLCs), or UNITE (for Premium PL7s), or UMAS (for Premium/Quantum Unity).
<b>Driver</b>	The driver is determined by the project type (shaded), i.e. either BUSX_ASYNC for Premium modules, or BUSX for Quantum modules.
<b>Base rate</b>	Used to define the default frequency at which variables are read.
<b>Comment</b>	Used to add a comment to the device service properties. This comment is only visible in this screen.
<b>Service status variable</b>	Used to determine the status of the device service.

### Configuring Variables

This table enables you to import variables coming from a PLC:



This table enables you to import variables coming from a non-PLC device:



This table describes the various fields that make up the variables configuration screen:

Fields	Function
<b>Input fields</b>	
<b>No</b>	ID of the variable, this number is generated automatically.
<b>Symbol</b>	Name of the variable stored in the module.
<b>Variable</b>	Physical address of the variable.
<b>Type</b>	Used to define the type of variable. The following types may be used: <ul style="list-style-type: none"> <li>● BOOL,</li> <li>● EBOOL,</li> <li>● SHORT,</li> <li>● USHORT,</li> <li>● INT,</li> <li>● UINT,</li> <li>● DINT,</li> <li>● REAL,</li> <li>● STR.</li> </ul>
<b>Access</b>	Set the access right of the variable: <ul style="list-style-type: none"> <li>● R (Read only),</li> <li>● R/W (Read/Write enabled).</li> </ul>
<b>Persistent</b>	Select this box if you want to use the variables with services.
<b>Rate</b>	Variable scan frequency (in ms).
<b>Value</b>	Display the value of the variable. You can change the value of variables that are write enabled.
<b>Comment</b>	Used to add a comment to the device service properties. This comment is only visible in this screen.

Fields	Function
<b>Buttons</b>	
<b>Remove</b>	Delete the selected variable from the table.
<b>Duplicate</b>	Duplicate the selected variable.
<b>Import PLC symbols</b>	Use this button to import PLC symbols from a FEF (PL7), STU (Unity Pro), XVM (Unity Pro) or PRJ (Concept) file. <b>NOTE:</b> Concept projects (.prj) can only be imported if the <b>ExportVariables</b> parameter in <i>concept.ini</i> is set to 1. You must modify this parameter before saving the project under Concept (re-start Concept after modifying the parameter setting).
<b>Import Usual Topics</b>	Use this button to import variables from a non-PLC device.
<b>Animate persistent</b>	Use this button to animate variables when the project is in online mode, with connections either to the simulator or to the module.
<b>Import from CSV</b>	Use this button to import symbols from a .csv file. This file may be created using Excel or a text editor (in this case, change the .txt file extension to .csv). The following syntax must be used: <i>symbol,variable,type,access,persistence,rate,comment</i> access = 0 for a read only symbol and 1 for a write enabled symbol. persistence = 0 for a non persistent symbol and 1 for a persistent.
<b>Export to CSV</b>	Use this button to export the variable table for the current service to a .csv file.

**NOTE:** For more information about variable management, refer to *Web Designer for TSX ETG 30•• User Manual*.

## 5.3 Email and SMS Service

---

### Overview

This section describes the Email and SMS service.

### What's in this Section?

This section contains the following topics:

Topic	Page
Email and SMS Service Presentation	277
Email and SMS Services Configuration	279

---

## Email and SMS Service Presentation

### Presentation

The TSX ETG 30•• module can automatically and dynamically send email or SMS to alert specified users to:

- production reports,
- maintenance reminders,
- plant status updates,
- other plant information.

The email service enables you to create various email notifications (including recipient's names, email addresses, message subject, email body and attached files).

Email bodies can include fixed text message, hyperlinks, file attachment and real-time application values dynamically integrated in the email body at the moment the email is sent by the module. File can be attached to the email (for example Data logging file generated by the Data logging service).

SMS messages may also be sent to mobile phones if you are using a TSX ETG 3021 or if the client's email server has the capability. SMS are a dedicated configuration of the email service. Email or SMS are sent when predefined application or process event is triggered.

### Service Requirements

The email service of the TSX ETG 30•• module only provides an SMTP client interface. The email service client connects to a local or remote SMTP server to distribute the mail to its recipients. Local SMTP server has to be installed in the site where the TSX ETG 30•• module is installed. Remote SMTP server can be available from your email provider. TSX ETG 30•• modules support DNS and authentication functions in order to connect to the SMTP server of the provider. The TSX ETG 3021 module has an embedded GSM modem. They can send directly SMS to destination mobile phone without installation of specific devices on the network.

### Service Operation

Email service acts as an SMTP client. When the preconfigured event trigger occurs, the TSX ETG 30•• module uses SMTP (over TCP port number 25) to send the email notification to the SMTP server. That server is connected to the plant's network or to the Internet, thereby allowing the message to reach the destination recipients. The TSX ETG 3021 module has an embedded GSM modem. They are able to send directly SMS without other devices on the network.

**NOTE:** Even though notifications are sent automatically after an event is triggered, there may be a significant delay before the recipient gets the message. A notification sent to a mobile phone is received only when the phone is on and within the coverage area. Therefore, this service should only be used for noncritical notifications, such as maintenance reminders or production reports

## Security

Email service supports SMTP server authentications. SMTP server must allow PLAIN or LOGIN authentication. Other authentication protocols are not supported. This optional login password, which is authenticated by the SMTP mail server, verify if the client is authorized to connect to SMTP server. An optional setting allows to change the TCP port number from the default setting of 25 to adapt to specific SMTP server configuration

## Limitation

The number of messages (email or SMS) you can configure in the project is restricted to 100.

The maximum number of Email or SMS services is 2.

**NOTE:** The SMTP server must allow PLAIN or LOGIN authentication. Other authentication protocols are not supported.

## Email and SMS Services Configuration

### Presentation

This section describes how to configure the Email / SMS service

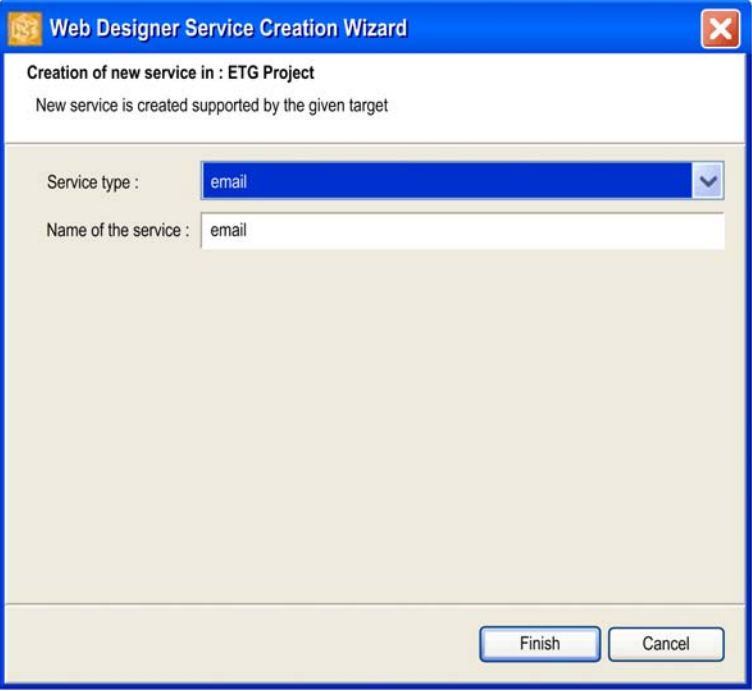
**NOTE:**

- The Email and SMS services are configured using Web Designer for TSX ETG 30\*\*.
- The SMS service is only available for the TSX ETG 3021 module.
- Schneider Electric does not guarantee:
  - integrity of data sent by the module with the email or SMS service,
  - the exactness of the time at which the message has been sent.
- If an anomaly occurs, the status variable of the email service (*see page 262*) is set to **error**.





## Creating an Email / SMS Service

Step	Action
1	Click the name of your project in the browser.
2	<p>To add the new service you can:</p> <ul style="list-style-type: none"> <li>● click <b>Project</b> → <b>New</b> → <b>Service</b> or,</li> <li>● right-click <b>Services</b> in the browser, then click <b>New Service</b> .</li> </ul> <p><b>Result:</b> the following window appears:</p> 
3	Select <b>Email</b> in the <b>Service Type</b> list box.
4	Enter the service name, or click <b>Finish</b> to keep the name by default.
5	The Email service has been created.

## Configuring the Properties

The screenshot shows a configuration window for SMTP server properties. It is divided into several sections:

- SMTP server:**
  - SMTP server address: [Text Input]
  - SMTP server port: [25]
  - Secure authentication: [ ]
  - Login: [Text Input]
  - Password: [Text Input]
- Sender:**
  - Sender: [Text Input]
  - Reply address: [Text Input]
- Module:**
  - Maximum size of send queue: [100]
  - Time before retry to send (in seconds): [5]
- Service:**
  - Service status variable: [plc.plc.dbStatus]

At the bottom of the window, there are two tabs: 'E-mails' and 'Properties'.

The fields in the properties configuration screen are:

Fields	Function
<b>SMTP server</b>	
<b>SMTP server address</b>	SMTP server name or IP address. If the SMTP server is located on the PPP link, the server name must appear in the Phone list ( <i>see page 216</i> ).
<b>SMTP server port</b>	TCP port used by the SMTP server (generally port 25).
<b>Secure Authentication</b>	Check this button if authentication is needed to access the SMTP server. The server must allow PLAIN or LOGIN authentication.
<b>Login</b>	Login to access the SMTP server.
<b>Password</b>	Password to access the SMTP server.
<b>Sender</b>	
<b>Sender</b>	Email address for the device sending the email. This address will identify the device when the user receives the Email.
<b>Reply address</b>	Email address to which a reply will be sent if the user clicks <b>Reply</b> .



The fields in the Email and SMS configuration screen are:

Fields	Function	
<b>Send SMS</b>	Select this box to send a SMS instead of an email.	
<b>Identifier</b>	Name for the Email.	
<b>Trigger</b>	Event that will trigger the Email. <b>Note:</b> You can only use persistent variables as trigger.	
<b>Type</b>	<b>NY:</b> Notify	Triggered by bit status change or word value change.
	<b>RE:</b> Rising Edge	Triggered by a bit rising edge or by an increasing word value.
	<b>FE:</b> Falling Edge	Triggered by a bit falling edge or by a decreasing word value.
	<b>BQ:</b> Bad Quality	Triggered if the trigger status is Bad quality.
<b>Destination</b>	Email addresses or SMS numbers of the recipients (separator ',').	
<b>Subject</b>	Subject of the notification.	
<b>Content</b>	Content of the notification.	
<b>Source</b>	Select the media from which the attached file comes from.	
<b>Path</b>	Specify the name and the path of the file to be attached to the email. Typical attached files are Datalogging CSV files. Default path is /USERDATA for all Media	

**NOTE:**

- You can include dynamic application data inside the message body:
  - double click in the body of the message at the place where you want to display the value of a variable and select the variable within the list which is displayed.
  - type the variable name between brackets. If you want to print the value of the value1 variable created in the device service, type (device.plc1.value1)
- The following fields are mandatory to record and save an Email: **Identifier**, **Trigger** and **Destination**.

---

## 5.4 Calculation Service

---

### Overview

This section describes the Calculation service.

### What's in this Section?

This section contains the following topics:

Topic	Page
Calculation Service Presentation	286
Calculation Service Configuration	288

## Calculation Service Presentation

### Presentation

The TSX ETG 30\*\* module can perform various arithmetic and logical operations on a combination of application variables. The calculation service also enables you to read/write variables of the device service as well as inputs/outputs of the embedded I/O board. Some of these calculations include scaling, formatting, and logic processing for event triggering.

The calculation service comprises spreadsheet where formulas are defined in cells. The spreadsheet is interpreted and processed by the server. The result of each formula is associated with a new internal variable. Formulas are executed periodically, according to the rate configured for each spreadsheet.

### List of Available Operators

The following list provides details of numeric binary, boolean and unary operators. The result of the operation is displayed in the Result type column:

Type	Operator	Description	Example
Binary numeric	+	addition	1+2 displays the result "3"
	-	subtraction	5-2 displays the result "3"
	*	multiplication	5*2 displays the result "10"
	/	division	15/3 displays the result "5"
	<	less than	1<2 displays the result "true"
	>	greater than	1>2 displays the result "false"
	<=	less than or equal to	1<=2 displays the result "true"
	>=	greater than or equal to	2>=2 displays the result "true"
	==	equal to	3==2 displays the result "false"
!=	not equal to	1!=2 displays the result "true"	
Boolean numeric	&	and	1&1 displays the result "1"
		or	1   0 displays the result "1"
Unary	!	inverse	!1 displays the result "0"
	-	sign inversion	-1 displays the result -1
	.<. ? . : .		a<b ? c:d returns c if a<b otherwise the result is d

#### NOTE:

- The calculation service enables you to mix variables of different types
- For exceptional mathematical cases such as division by 0, the result of the calculation is *nan* (not a number).

**Execution of Formula**

Formulas are executed periodically, according to the rate configured in the **Properties** screen. The formula cells are interpreted then executed one by one from top to bottom. The result variable configured in the variable field is updated with the new value.

**Limitation**

The number of calculations you can configure in the project is restricted to 100.  
The maximum number of calculation services is 2.

## Calculation Service Configuration

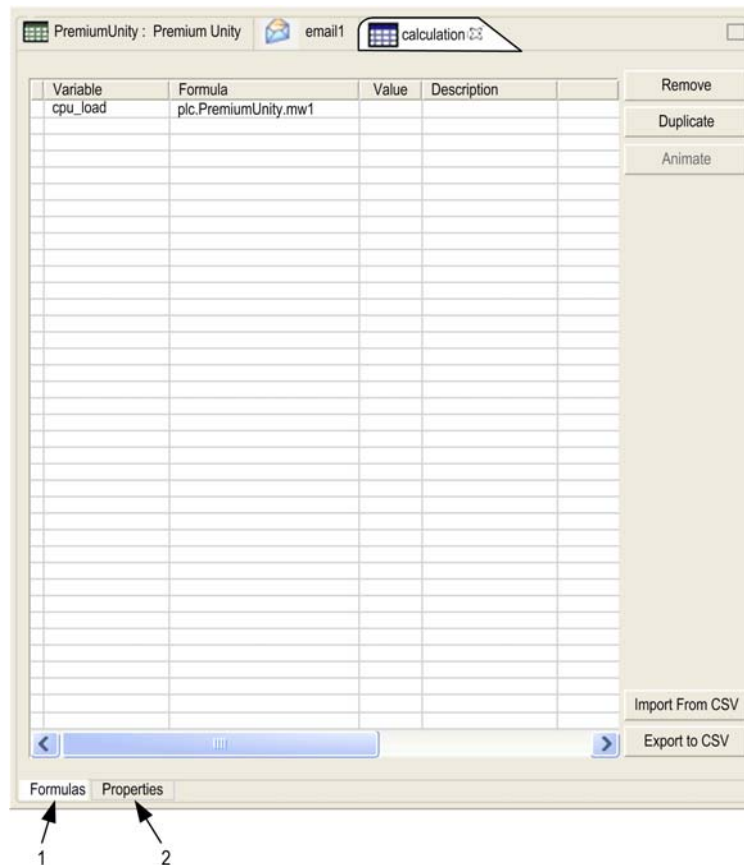
### General

This section describes how to configure the Calculation service.

**NOTE:**

- The Calculation service is configured using Web Designer for TSX ETG 30••.
- If an anomaly occurs, the status variable of the Calculation service (see page 265) is set to **error**.

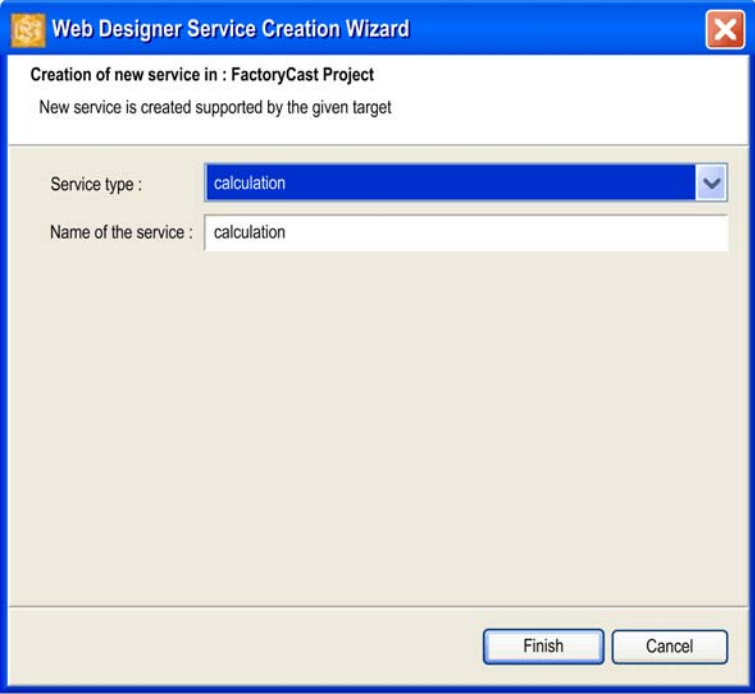
The properties configuration screen for the Calculation service is shown below:



- 1 Use the Formulas tab to configure formula.
- 2 Use the Properties tab to configure properties common to all formula.



## How to Create a Calculation Service

Step	Action
1	Click the name of your project in the browser.
2	<p>To add the new service you can:</p> <ul style="list-style-type: none"> <li>● click <b>Project</b> → <b>New</b> → <b>Service</b> or,</li> <li>● right-click <b>Services</b> in the browser, then click <b>New Service</b>.</li> </ul> <p><b>Result:</b> the following window appears.</p> 
3	Select <b>Calculation</b> in the <b>Service Type</b> list box.
4	Enter the service name, or click <b>Finish</b> to keep the name by default,
5	The Calculation service has been created.

### Configuration of the Table Properties

Calculation Rate (in ms)

Service status variable

Formulas Properties

This table describes the fields in the Calculation configuration screen:

Fields	Function
<b>Calculation rate (in ms)</b>	Listbox in which you can set the frequency of execution of formula.
<b>Service status variable</b>	Used to determine the status of the calculation service.



Fields	Function
<b>Import from CSV</b>	Use this button to import symbols from a .csv file. This file may be created using Excel or a text editor (in this case, change the .txt file extension to .csv). The following syntax must be used: <i>variable,formula,description</i> .
<b>Export to CSV</b>	Use this button to export the variable table for the current service to a .csv file.

---

## 5.5 Database Service

---

### Overview

This section describes the Database service.

### What's in this Section?

This section contains the following topics:

Topic	Page
Database Service Presentation	294
Database Service Configuration	295

## Database Service Presentation

### Presentation

The Database service enables the TSX ETG 30\*\* to perform automatic archiving of application data (process events, status, measures, etc.) into a remote relational database server. Data can be archived (written) periodically or on event. Capability to connect directly TSX ETG 30\*\* modules to relational databases provide easy integration of control level system into company information management system.

The TSX ETG 30\*\* module can be connected directly to the following remote relational databases:

- Oracle,
- SQL Server,
- MySQL.

Various database tables can be created per database server. Database service allows data archiving into 3 different database servers for redundancy purpose.

### Database Characteristics

- type of database supported:
  - Oracle: from version 8i to 10g or above,
  - SQL Server: from version 6.5 to 2008 R2 or above,
  - MySQL: from version 4.1 to 5.5 or above.
- automatic table creation: the TSX ETG 30\*\* automatically creates a table in the database

### Limitation

The maximum number of database services that can be configured is 3.

The maximum number of database tables is 10 per service.

The maximum number of columns/variables is 50 per table.

## Database Service Configuration

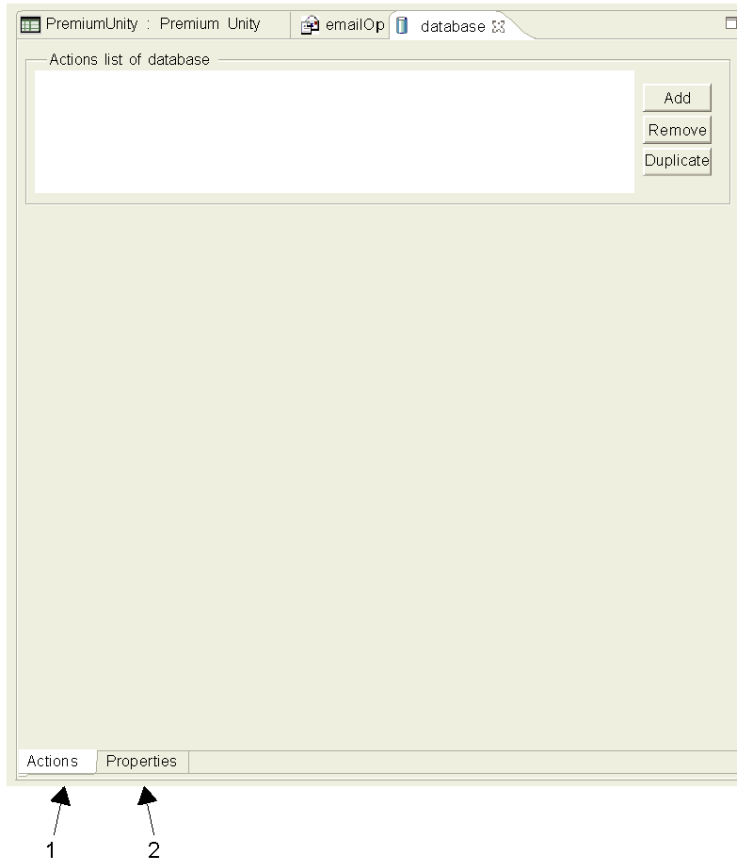
### General

This section describes how to configure the Database service.

**NOTE:**

- The Database Connection service is configured using Web Designer for TSX ETG 30\*\*.
- if an anomaly occurs, the status variable of the Database Connection service (*see page 262*) is set to **error**.
- Schneider Electric does not guarantee:
  - integrity of data sent by the module with the database service,
  - the exactness of the time at which data are logged.

The configuration screen for the Database connection service is shown below:

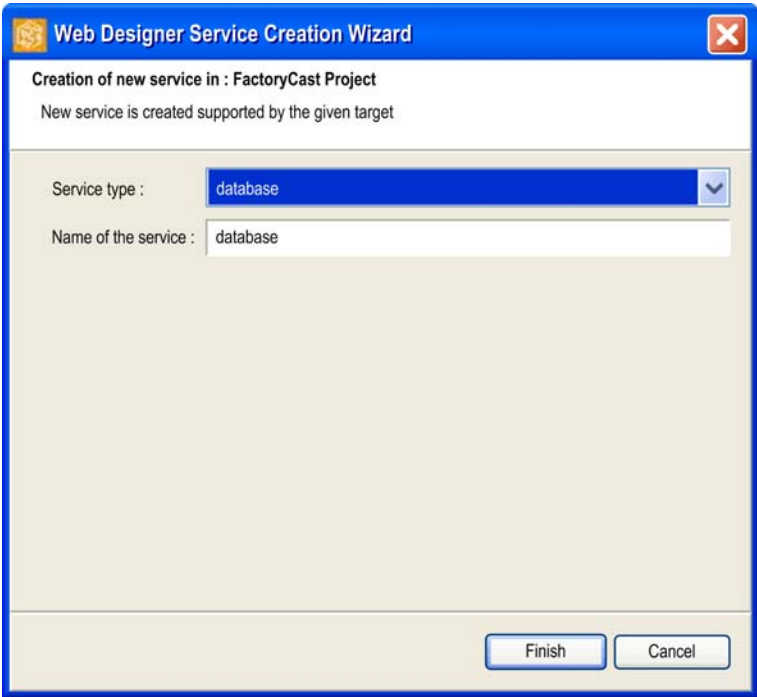


- 1 Use this button to configure the actions list (Database tables).
- 2 Use this button to configure the general properties of the database.



## How to Create a Database Service

The following table describes how to create a Database service:

Step	Action
1	Click on the name of your project in the browser.
2	<p>To add the new service you can:</p> <ul style="list-style-type: none"> <li>• click <b>Project</b> → <b>New</b> → <b>Service</b> or,</li> <li>• right-click <b>Services</b> in the browser, then click <b>New Service</b>.</li> </ul> <p><b>Result:</b> the following window appears.</p> 
3	Select <b>database</b> in the <b>Service Type</b> list box.
4	Enter the service name, or click <b>Finish</b> to keep the name by default,
5	The Database connection service has been created.

### Configuration of Properties

The properties configuration window for the Database connection service is shown below:

The following table describes the parameters of the database service:

Fields	Function
Type	<ul style="list-style-type: none"> <li>● <b>Oracle</b>: from version 8i to 10g or above</li> <li>● <b>MySql</b> from version 4.1 to 5.5 or above</li> <li>● <b>SQL Server</b>: from version 6.5 to 2008 R2 or above</li> </ul>
Address	IP address or DNS name of the database server.
Port	Used to validate a default address.
Use Default Port	Port used by the database to access the server.
Database Name	Name of the database that will contain all data configured by the user.
User Name & Password	Protection of database and access by username and password (database connection string).
Timeout	Period of time that will be allowed to elapse before the system kills the current process when an anomaly occurs.
Service status variable	Used to determine the status of the database service.



The following table describes the fields in the database tables configuration screen:

Fields	Function
List of actions for the database	List of all tables configured on the server. Use the <b>Add</b> , <b>Remove</b> and <b>Duplicate</b> buttons to create, delete or copy a table. When you duplicate an existing database, the new name will have the suffix <code>_copy</code> added to its original name.
Actions parameters	Name of the new archiving table. If the table does not exist in the database, it will create it on the first connection. To trigger variable logging, you must specify an event. This will either be a periodic event (use periodic log), in which case you need to set a time base, or it will be an event associated with a variable (use trigger), in which case you need to provide the name of the variable (for example, calculation.calculation1.Pressure1).
SQL Request	Two options are available: <ul style="list-style-type: none"> <li>● create SQL requests in the <b>Free Text SQL Request</b> window,</li> <li>● or use the table provided to specify the name of the columns, and the variables type and name. In this case, the column name syntax must comply with the database syntax. As a general rule, avoid using punctuation marks, upper case letters, and hyphens, (refer to your database documentation).</li> </ul>
Maximum records	Configure the maximum number of records that will be stored in a database table. If that number is reached, the oldest data are erased.
Delete table when start	If the box is checked, the database table will be deleted on restart of the module.
Column name	Name of the column. Its syntax must comply with the database syntax.
Variable	Variable to be logged.
Type	Type of the variable.

### Database Properties

The table below shows the equivalents of Xopen types for other databases:

Xopen type	MySql,	Oracle,	SQL Server.
BOOLEAN	SMALLINT	NUMBER(1)	SMALLINT
TINYINT	SMALLINT	SMALLINT	SMALLINT
SMALLINT	SMALLINT	NUMBER	SMALLINT
INTEGER	INTEGER	NUMBER	INTEGER
BIGINT	DECIMAL(200)	NUMBER	NUMERIC(38,0)
DOUBLE	DECIMAL(50,80)	NUMBER	FLOAT(32)
REAL	REAL	REAL	REAL
TIMESTAMP	TIMESTAMP	TIMESTAMP	TIMESTAMP
DATE	DATETIME	DATE	DATETIME
VARCHAR	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)
All other types	VARCHAR(255)	VARCHAR(255)	VARCHAR(255)

---

## 5.6 Datalogging Service

---

### Overview

This section describes the Datalogging service.

### What's in this Section?

This section contains the following topics:

Topic	Page
Datalogging Service Presentation	302
Datalogging Service Configuration	306

## Datalogging Service Presentation

### Description

The datalogging service allows application data archiving (events, alarms, process data, devices status, measures, etc.) in the internal memory of the TSX ETG 30\*\* module. This service allows you to log data into local CSV files in ASCII format. The CSV files are stored in the memory of the TSX ETG 30\*\* module (internal flash memory or saved memory of the module, Compact Flash (CF) card or USB memory key).

The datalogging can be performed either periodically, or when a specific event occurs (configured by the user). CSV files are directly usable by a MS Excel spreadsheet or a database management system (DBMS). The CSV files can also be sent via FTP, or attached to an email (using the email service) and sent automatically to specified users. The CSV files can also be accessed by FTP client.

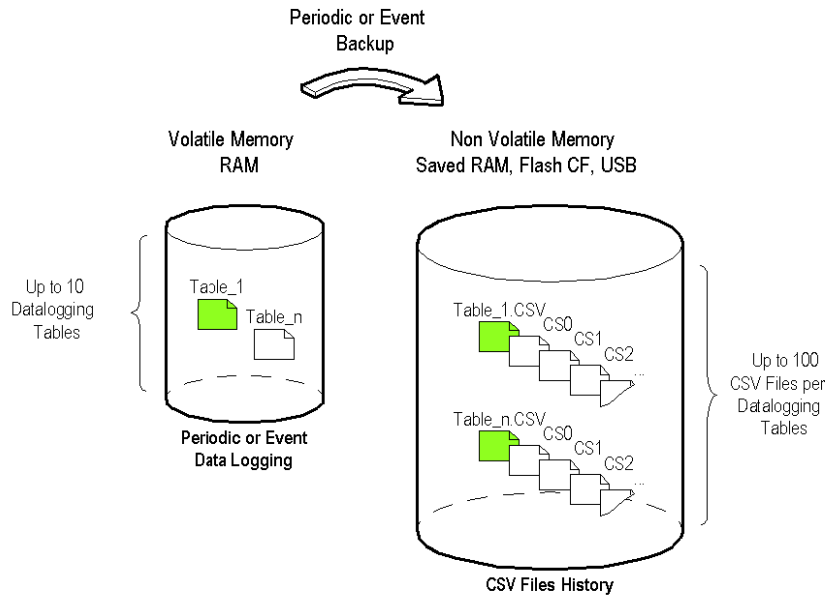
### Principle

Datalogging is performed as described below:

- Datalogging is performed basically in the RAM memory of the module (in order to not have issues with the life duration of the flash memories due to the limited number of write to such memories - typically 100,000 maximum write access by default, but it may be changed by the user for each table).
- Datalogging service can manage up to 10 groups of datalogging files (tables). This allows to archive several different tables of data each with a different logging period.
- Datalogging files can be backup, periodically or on event, from the RAM memory to non volatile memory of the module (saved RAM, internal flash, Compact flash card or USB memory key) into history files in CSV format. Table\_n.csv represents the last backup file, history files are renamed as Table\_n.cs0, Table\_n.cs1, etc.
- CSV files can be purged from Flash memory (internal, Compact flash or USB key) on event trigger.

**NOTE:** In case of loss of power during datalogging, the file that is being backed up is lost.

Principle overview:



## File Format

The file format is fixed and cannot be modified by the user. The file is encoded in pure ASCII format as a text file with a .csv extension. Microsoft Excel is able to open these files.

### Example of log file:

```
2007-10-01;02:44:55;plc.plc1.height;150;plc.plc1.length;200;plc.plc1.width;50;
2007-10-01 03:48:08; plc.plc1.height;140;plc.plc1.length;150;plc.plc1.width;30;
2007-10-01 04:55:10; plc.plc1.height;220;plc.plc1.length;280;plc.plc1.width;80;
2007-10-01 06:01:05; plc.plc1.height;170;plc.plc1.length;220;plc.plc1.width;60;
```

### Example of optimized log file:

```
Date;plc.plc1.height;plc.plc1.length;plc.plc1.width;
2007-10-01 02:44:55;150;200;50;
2007-10-01 03:48:08;140;150;30;
2007-10-01 04:55:10;220;280;80;
2007-10-01 06:01:05;170;220;60;
```

### Size of the Log File

The following table shows you an estimation of the log file size in bytes depending on the number of variables logged and the number of logs:

Number of logs	Number of variables						
	1	2	5	10	20	50	100
1	65	110	245	470	920	2270	4520
2	130	220	490	940	1840	4540	9040
5	325	550	1225	2350	4600	11350	22600
10	650	1100	2450	4700	9200	22700	45200
20	1300	2200	4900	9400	18400	45400	90400
50	3250	5500	12250	23500	46000	113500	226000
100	6500	11000	24500	47000	92000	227000	452000

**NOTE:** Schneider Electric recommends you to do not backup datalogging files at a frequency lower than 30 minutes in order to save internal flash life.

### Backup time

You could find hereafter an estimation of backup time depending of the media used:

- Internal Flash: 1 Kb backup = 40 ms rename = 50 ms
- CF Card: 1 Kb backup = 250 ms rename = 100 ms
- USB 2.0: 1 Kb backup = 500 ms rename = 200 ms

(For example: 50 variables x 500 records = 1135 K. Backup with 10 saving files on USB = 1135x500 ms + 10x200 ms = 569,5 s).

**NOTE:** When designing your application, allow enough time between backups.

To avoid overlaps, the time gap between consecutive backups must be greater than the estimated time.

### Limitations

The Datalogging service has the following limitations:

- The maximum number of Datalogging services is 1.
- The maximum number of tables is 10 in total.
- The maximum number of variables is 100 per table.
- The maximum number of records per table is 10000.
- The maximum number of backup copies is 2 by default (up to 100 backup CSV files per table), it may be changed for each table.
- The Datalogging Service Web Designer application must be designed to require less than 1 MB of memory.



The memory space allocated to the datalogging service is limited to:

- 500 kB on the saved RAM memory,
- 15 MB on the internal flash memory,
- 1 GB on the Compact Flash card,
- 2 GB on the USB memory key.

## Datalogging Service Configuration

### Presentation

This section describes how to configure the datalogging service.

You can use the log file to animate a Datalogging history object (*see page 456*) in the Graphic Editor.

#### NOTE:

- The datalogging service is configured using Web Designer for TSX ETG 30••.
- Schneider Electric recommends that you do not backup datalogging files at a frequency lower than 30 minutes in order to save flash life (internal, internal saved RAM, CF card or USB memory key).
- If an anomaly occurs, the status variable of the datalogging service (*see page 263*) is set to **error**.
- USB memory keys do not have to comply to the same rigorous industrial standards and environmental constraints that the TSX ETG 30•• modules must meet.
- Schneider Electric recommends that you do not use USB memory keys for permanent operations. Instead, use the CF card feature, which offers greater stability.
- Schneider Electric does not guarantee the exactness of the time at which data have been logged.

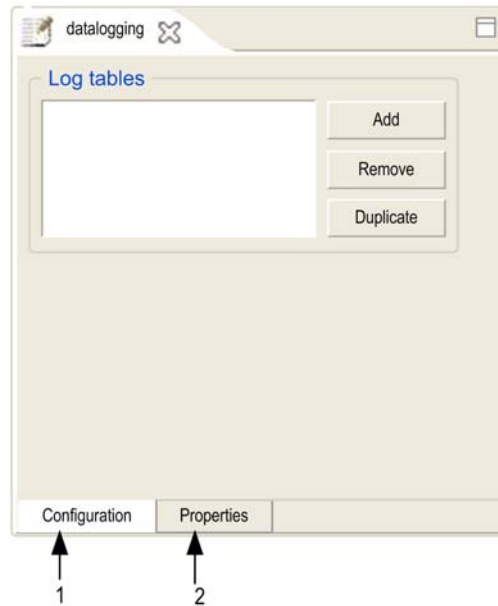
### Accessing the Datalogging Configuration and Properties Screen

To access the datalogging configuration screen, double click on Datalogging in the project navigation window.

The configuration properties screen contains:

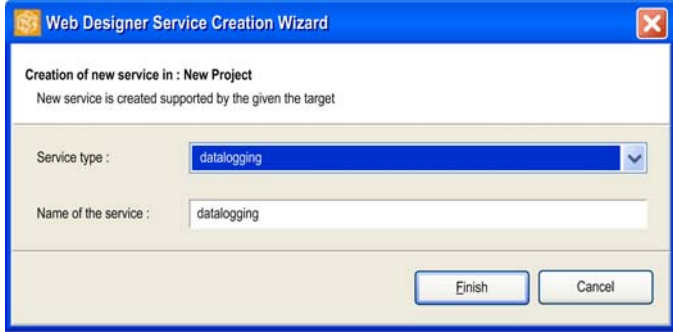
- a Configuration tab
- a Properties tab

The **configuration** properties screen for the datalogging service is shown below:



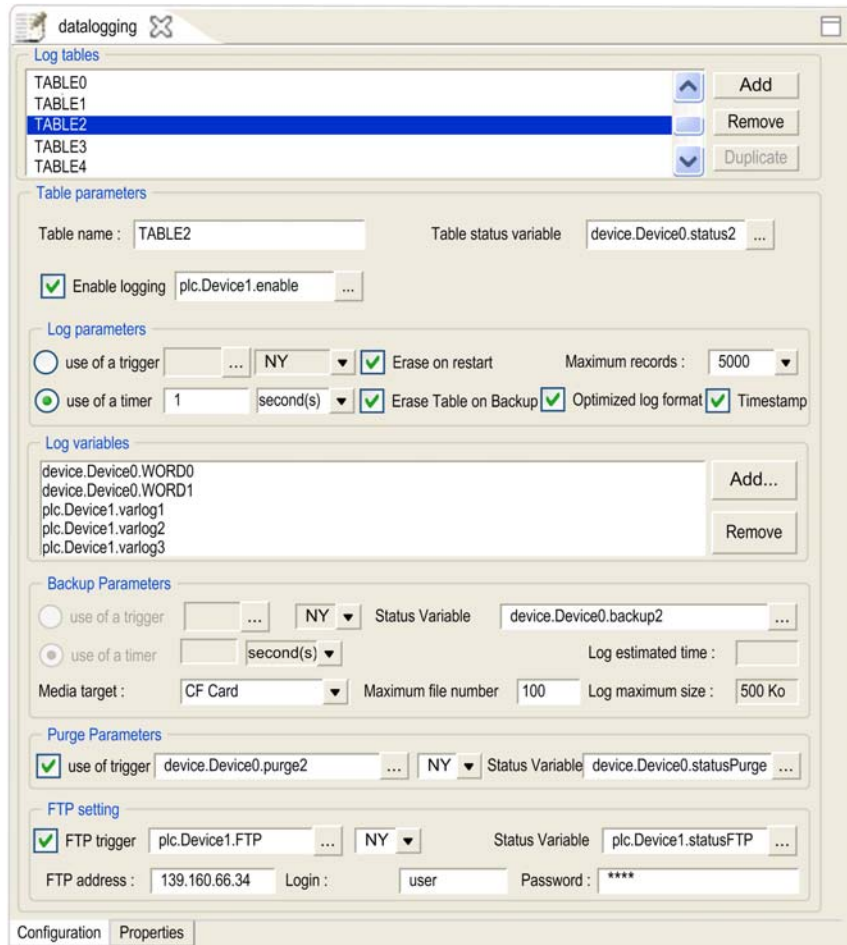
- 1 Use this tab to customize the Datalogging service.
- 2 Use this tab to configure the general properties of the datalogging service.

## How to Create a Datalogging Service

Step	Action
1	Click the name of your project in the browser.
2	<p>To add the new service you can:</p> <ul style="list-style-type: none"> <li>● click <b>Project</b> → <b>New</b> → <b>Service</b> or,</li> <li>● right-click <b>Services</b> in the browser, then click <b>New Service</b>.</li> </ul> <p><b>Result:</b> the following window appears.</p> 
3	Select <b>datalogging</b> in the <b>Service Type</b> list.
4	Enter the service name, or click <b>Finish</b> to keep the name by default.
5	The datalogging service has been created.

**NOTE:** You can create only one datalogging service.

## View of the Datalogging Configuration Screen



## Datalogging Configuration Screen Parameters

Log Tables Parameters:

Fields	Function
<b>Log Tables</b>	<p>List of the current log tables stored in the module. It is possible to:</p> <ul style="list-style-type: none"> <li>● Create a new log file using the <b>Add</b> button,</li> <li>● Remove a log file using the <b>Remove</b> button,</li> <li>● Duplicate a log file using the <b>Duplicate</b> button.</li> </ul>

Tables Parameters:

Fields	Function
<b>Table name</b>	Name of the log file that contains all data configured by the user.
<b>Table status variable</b>	Determine the status of the table.
<b>Enable logging</b>	By selecting this check box and defining an associated variable in the <b>Enable logging</b> variable, the table can only perform actions such as: log data, backup, purge or FTP when the associated variable is set to a value other than zero and when the trigger is activated. <b>NOTE:</b> Select the type of trigger in the drop down menu. Refer to Meaning of the Trigger Types ( <i>see page 314</i> ).

Log Parameters:

Fields	Function
<b>Use of a trigger / timer</b>	To trigger logging of variables, you must configure an event in the <b>Use of a trigger / timer</b> variable. This event must be either: <ul style="list-style-type: none"> <li>• a periodic event (<b>use of a timer</b>), in which case you need to set a time base</li> <li>• an event associated with a variable (<b>use of a trigger</b>), in which case you need to provide the name of this variable (for example, calculation.calculation1.Pressure1).</li> </ul> <b>NOTE:</b> Select the type of trigger in the drop down menu. Refer to Meaning of the Trigger Types ( <i>see page 314</i> ).
<b>Erase on restart</b>	Selecting this check box deletes the table log files on restart of the module.
<b>Erase Table on Backup</b>	Selecting this check box removes the previous logs in the table after backup, so the table restarts empty.
<b>Timestamp</b>	Selecting this box records in the log file Timestamp (hour and date) for each event. <b>Note:</b> Timestamps are mandatory if you want to use the log file for a datalogging history.
<b>Optimized log format</b>	Selecting this box for optimizes the log file format ( <i>see page 303</i> ) of the .csv file. In this case, the variable name does not appear in each record.
<b>Maximum record number</b>	Configures the maximum of records that can be stored in a log file. If that number is reached, new records overwrite old records.

Log Variables Parameters:

Fields	Function
<b>Log variables</b>	<p>The <b>variable name</b> part lists the variables (PLC/devices or Calculation variables) that are stored in the log file.</p> <p>It is possible to:</p> <ul style="list-style-type: none"> <li>● Create a new log variable using the <b>Add</b> button</li> <li>● Remove a log variable using the <b>Remove</b> button</li> </ul>

Backup Parameters:

Fields	Function
<b>Use of a trigger / timer</b>	<p>To trigger logging of variables, you must configure an event in the <b>Use of a trigger / timer</b> variable. This event must be either:</p> <ul style="list-style-type: none"> <li>● a periodic event (<b>use of a timer</b>), in which case you need to set a time base</li> <li>● an event associated with a variable (<b>use of a trigger</b>), in which case you need to provide the name of this variable (for example, calculation.calculation1.Pressure1).</li> </ul> <p>The <b>use of a trigger</b> and <b>use of a timer</b> fields are greyed-out depending whether the <b>Global backup</b> check box from the datalogging properties window (<i>see page 312</i>) is checked.</p> <p><b>NOTE:</b> Select the type of trigger in the drop down menu. Refer to Meaning of the Trigger Types (<i>see page 314</i>).</p>
<b>Media target</b>	Use to define the media target to use.
<b>Maximum file number</b>	Defines the maximum number of CSV files to use for each table. By default it is set to 2. The maximum authorized value is 100. The last file is the .csv file, the previous is the .0 file.
<b>Status variable</b>	Determine the status of the Backup action. The status is set to 0 when the service starts, to 1 when the backup action begins, and to 2 when the backup action completes.
<b>Log estimated time</b>	Provides information on the time length of the log based on the maximum file number, the logging and backup period. It is only available when using a timer.
<b>Log maximum size</b>	Provides the maximum log size. This field can be changed via the backup parameters in the Datalogging Properties Screen ( <i>see page 312</i> ).

Purge Parameters:

Fields	Function
<b>Use of trigger</b>	Specifies the event that triggers the purge.
<b>Status Variable</b>	Determine the status of the Purge action. The status is set to 0 when the service starts, to 1 when the purge action begins, and to 2 when the purge action completes.

FTP Settings:

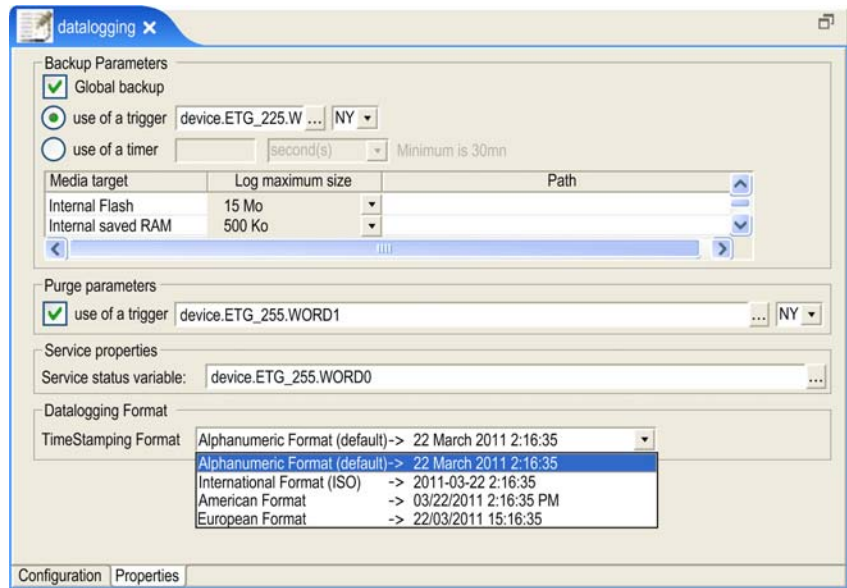
Fields	Function
<b>FTP trigger</b>	Specifies the event that triggers sending the selected table log files (CSV file) via FTP.
<b>FTP address</b>	The address of the remote FTP server.
<b>Status Variable</b>	Determine the status of the FTP action. The status is set to 0 when the service starts, to 1 when the FTP action begins, and to 2 when the FTP action completes.
<b>Login and Password</b>	Login parameters for the remote FTP server access.

**FTP Access to Datalogging CSV Files**

Datalogging CSV files stored in the module memory can be accessed from any FTP client utility using the following path in each of the memory media of the module:

- /NAND/FLASH1/USERDATA for the internal Flash.
- /RAMDISK/USERDATA on the internal saved RAM.
- /CFA00/USERDATA on the CF card.
- /USBHD/00/USERDATA on the USB key.

**View of the Datalogging Properties Screen**





## Datalogging Properties Parameters

### Backup Parameters:

Fields	Function
<b>Global backup</b>	When checked, all created tables use the same event to trigger a backup. When not checked, each created table has its own event to trigger a back up. <b>NOTE:</b> When a <b>Global backup</b> check box is ticked, the fields <b>use of a trigger</b> and <b>use of a timer</b> become available.
<b>use of a trigger</b>	Provide the name of a variable to trigger variable logging on an event associated to this variable. <b>NOTE:</b> Select the type of your trigger in the drop down menu available on the left. The meaning of those types are available at Meaning of the Trigger Types ( <i>see page 314</i> ).
<b>use of timer</b>	Provides a periodic time base to trigger variable logging on an event.
<b>Media target</b>	<b>Internal saved RAM</b> to store the information on the RAM memory of the module <b>Internal Flash</b> to store the information on the Internal Flash memory of the module <b>CF card</b> to store the information on the Compact Flash card of the module <b>USB</b> to store the information on a USB memory key
<b>Log maximum size</b>	Specify the maximum size of memory allocated to the backup files. The maximum log file size is defined for each media via the drop down menu, but the value can be changed. For more information, refer to Datalogging Limitations ( <i>see page 304</i> ).
<b>Path</b>	If empty, the default path is \USERDATA for each of the media in Media target is as follow: <ul style="list-style-type: none"> <li>● /NAND/FLASH1/USERDATA for the internal flash</li> <li>● /RAMDISK/USERDATA for the internal saved RAM</li> <li>● /CFA00/USERDATA for the CF card</li> <li>● /USBHD/00/USERDATA for the USB key</li> </ul> <p>You can specify different relative paths if necessary, for example:</p> <ul style="list-style-type: none"> <li>● <b>MYDIR</b> in Internal Flash Path field: the files will be created in /NAND/FLASH1/USERDATA/MYDIR</li> <li>● <b>MYDIR/MYSUBDIR</b> in CF Card Path field: the files will be created in /CFA00/USERDATA/MYDIR</li> </ul>

### Purge Parameters:

Fields	Function
<b>Use of a trigger</b>	If checked, this event triggers a purge of the current backup files on all media currently in use. <b>NOTE:</b> Select the type of your trigger in the drop down. Refer to Meaning of the Trigger Types ( <i>see page 314</i> ).

Service Properties Parameters:

Fields	Function
<b>Service status variable</b>	Selects the variable with the associated event that is to be used as a trigger to check the status of the Datalogging service.

Datalogging Format Parameters:

Fields	Function
<b>TimeStamping Format</b>	Selects the format for the timestamping (Alphanumeric,International, European,American).

**NOTE:** With Microsoft Excel, in International and European formats, by default, the seconds are not displayed.

To see the seconds:

- Modify and add second in the cell format for the column date (jj/mm/aaaa hh:mm:ss) before saving the file.

### Meaning of the Trigger Types

You can select the type of your trigger. The types available are:

Type	Meaning	Function
<b>NY</b>	Notify	Triggered by a change from a bit status or word value.
<b>RE</b>	Rising Edge	Triggered by a bit rising edge or by an increasing word value.
<b>FE</b>	Falling Edge	Triggered by a bit falling edge or by a decreasing word value.
<b>BQ</b>	Bad Quality	Triggered if the trigger status has bad quality.

---

## 5.7 Active Pages Service

---

### Overview

This section describes the Active Pages service.

### What's in this Section?

This section contains the following topics:

Topic	Page
Active Pages Service Presentation	316
Active Pages Service Configuration	318

## Active Pages Service Presentation

### General

The active pages service allows you to create custom dynamic HTML pages that can be used for real-time HTML reporting. Active pages service can incorporate real-time values of application variables into the body of the HTML pages with a behavior similar to an active server page. Real-time values are refreshed by the sever side in run-time.

This technology has 2 benefits:

- Creation of animated Web pages without using Java applets, so that HTML page size is kept to a minimum. Very useful in case of modem connection or in a Pocket PC environment.
- The only protocol used for communication with the browser is HTTP, so this technology can cross firewalls.

### Accessing your Active Pages

To access the active pages, use the following syntax:

- `http://IP address of module: Active Pages port`, or
- `http://domain name of module: Active Pages port`

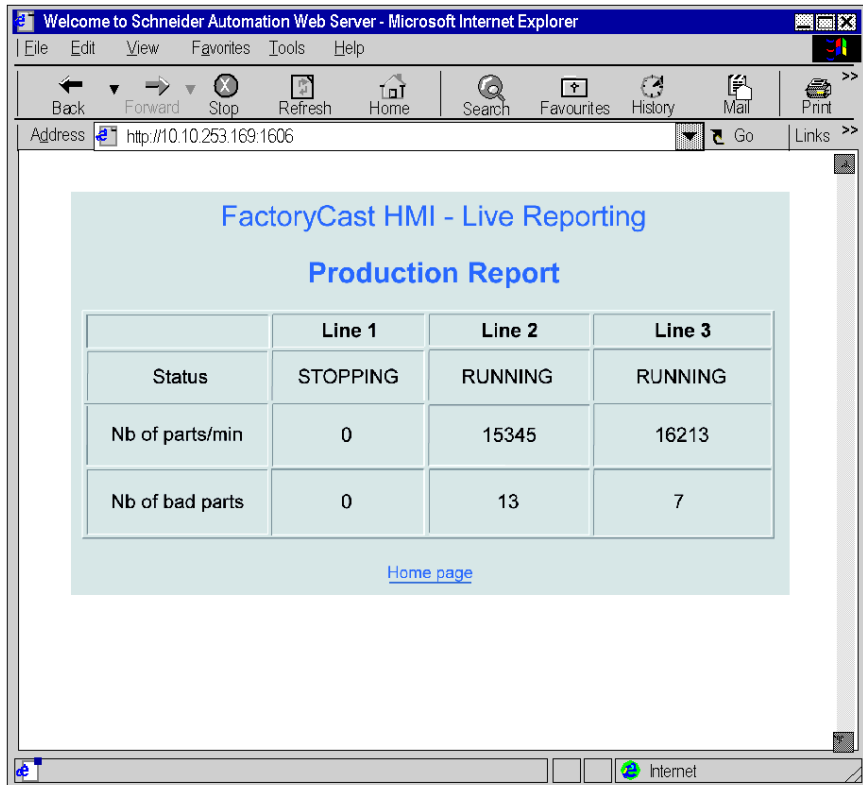
**Example:** `http://10.10.253.169:1606` or `http://my_website:1606`

To access a specific page, you must use the following syntax:

- `http://IP address of module: Active Pages port / name of page`, or
- `http://domain name of module: Active Pages port / name of page`

**Example:** `http://10.10.253.169:1606/page0` or `http://my_website:1606/page0`

The following diagram illustrates an example of an Active Page report:



**NOTE:** The page called up is not refreshed automatically but you can include a refresh command in the HTML code of the page.

Example of HTML code to be used in a page to be automatically refreshed:

```
<meta http-equiv="refresh" content="1";url="page1">
```

where content=refresh time in seconds:url=name of page to be reloaded

### Limitation

The maximum number of active pages service is 2 per project.

The active pages service may include up to a maximum of 64 pages in the project.

1,000 variables can be declared for the entire set of project pages.

## Active Pages Service Configuration

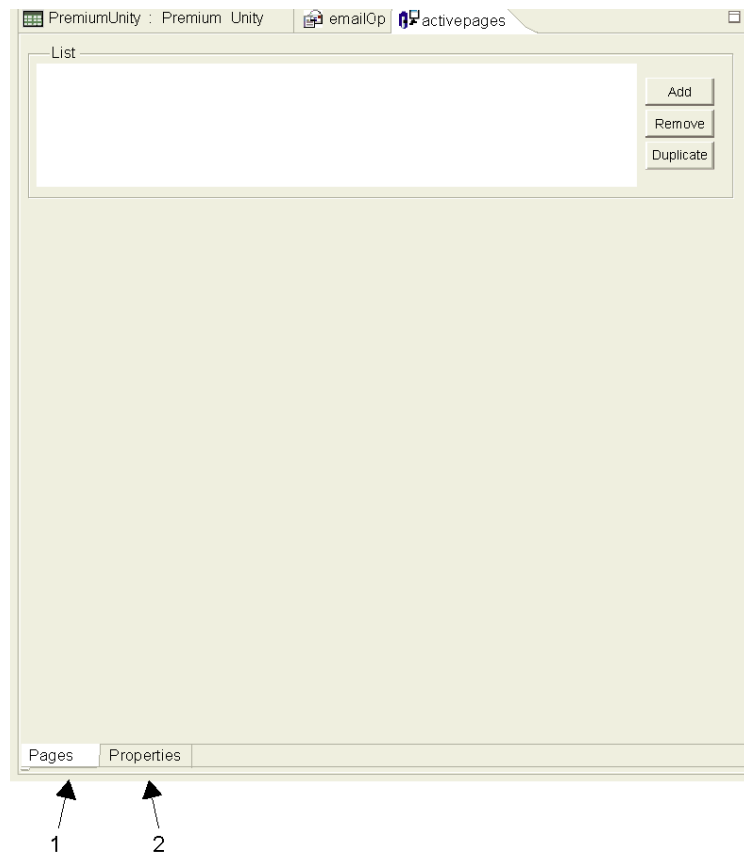
### Presentation

This section describes how to configure the Active Pages service.

**NOTE:**

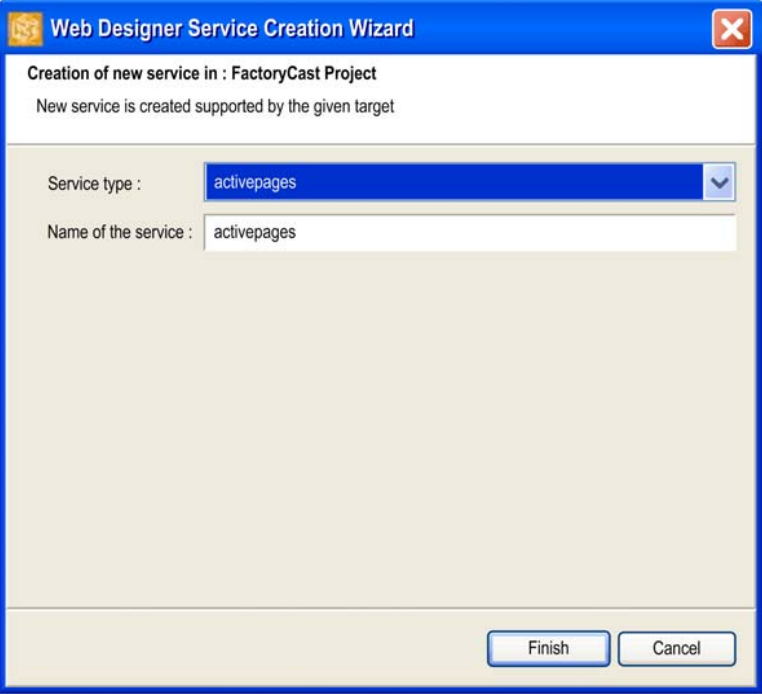
- The Active Pages service is configured using Web Designer for TSX ETG 30••.
- If an anomaly occurs, the status variable of the active pages service (see page 265) is set to **error**.

The properties configuration screen for the Active Pages service is shown below:



- 1 Use this button to configure the Active Pages service.
- 2 Use this button to configure properties of the pages.

## How to Create an Active Pages Service

Step	Action
1	Click on the name of your project in the browser.
2	<p>To add the new service you can:</p> <ul style="list-style-type: none"> <li>● click <b>Project</b> → <b>New</b> → <b>Service</b> or,</li> <li>● right-click <b>Services</b> in the browser, then click <b>New Service</b>.</li> </ul> <p><b>Result:</b> the following window appears.</p> 
3	Select <b>active pages</b> in the <b>Service Type</b> list box.
4	Enter the service name, or click <b>Finish</b> to keep the name by default.
5	The Active Pages service has been created.

## Configuration of Properties

The screenshot shows a configuration window with two sections. The top section, titled 'Page server properties', contains a 'Port' label followed by a text input field containing the value '1606'. The bottom section, titled 'Service properties', contains a 'Service status variable' label followed by an empty text input field and a small square button with three dots (a browse button).

The following table describes the fields in the properties configuration screen:

Fields	Function
Port	Active pages port number. The port numbers are between 1024 and 65535. By default, the first port declared will be assigned number 1606.
Service status variabe	Used to determine the status of the Active Pages service.

**NOTE:** It is possible to create two Actives Pages services. In this case, the two services must be configured with different port numbers.



## Configuring Pages

The pages configuration screen for the Active Pages service is shown below :

The screenshot shows the configuration interface for the Active Pages service. It is divided into two main sections: 'List' and 'Content'.

- List:** A table with one entry 'page0'. To the right of the table are three buttons: 'Add', 'Remote', and 'Duplicate'.
- Content:** A section for editing the page content. It includes a 'Name' field with 'page0' and an 'Import' button. Below this is a large text area for the HTML code. The code includes a meta tag for refresh, a header for 'Production Report', and a table with three rows of data. The table data is populated with variables from the PLC service, such as `{device.ETC_Server.SSTATUS}`, `{plc.Twido.SSTATUS}`, and `{plc.P57_WMY.SSTATUS}`.

At the bottom of the window, there are two tabs: 'Pages' and 'Properties'.

The following table describes the fields in the page configuration screen:

Fields	Function
List	List of all the active HTML pages created in the application on the server. Use the <b>Add</b> , <b>Remove</b> and <b>Duplicate</b> buttons to create, delete or copy a page. When an existing page is duplicated, the new name will have the suffix <code>_copy</code> added to its original name.
Import	Use this button to import complete HTML code from an <code>.htm</code> or <code>.txt</code> file. You can create the file using Frontpage or a text editor. It overwrites the code already written to the page.
Name	Name of the page currently being created. The page can be renamed in this field.
Page	Use this field to write the HTML code for the current page or to view code which has been imported. Double-clicking in the field enables you to display the list of variables of the PLC service. When one of these variables is selected, it is directly incorporated into the code. When one of these variables is selected, so that in runtime, realtime value of PLC or devices variables will be refreshed in the HTML report.

## 5.8 Recipe Service

---

### Overview

This section describes the Recipe service.

### What's in this Section?

This section contains the following topics:

Topic	Page
Recipe Service Presentation	323
Recipe Service Configuration	324

## Recipe Service Presentation

### Presentation

The Recipe service is used to initialize a group of variables set at predetermined values (also called recipe) when a specified event occurs. The predetermined values are saved in a recipe file in XML format. Recipe files can be stored locally in the TSX ETG 30\*\* memory or in a remote PC. The recipe execution is initialized automatically by the TSX ETG 30\*\* module on an event trigger (process event, operator action, etc). When the event occurs, the recipe service accesses the recipe file (in XML format) stored locally or remotely through an HTTP server, reads the values that correspond to the variables, then updates the PLC variables.

### Example of an XML File

```
<?xml version="1.0" encoding="utf-8" ?> -  
<table>-  
  <data>  
    <name>plc.localRTDB.VarRecette403</name>  
    <value>4</value>  
    <ingredient>dounwn part black</ingredient>  
  </data>- <data>  
    <name>plc.localRTDB.VarRecette402</name>  
    <value>3</value>  
    <ingredient>up part blue</ingredient>  
  </data>- <data>  
    <name>plc.localRTDB.VarRecette401</name>  
    <value>2</value>  
    <ingredient>left part red</ingredient>  
  </data>- <data>  
    <name>plc.localRTDB.VarRecette400</name>  
    <value>1</value>  
  </data>  
</table>
```

### Limitations

- Recipe files must be accessible by an HTTP server.
- You can create only 1 recipe service.

## Recipe Service Configuration

### Presentation

This section describes how to configure the Recipe service.

Executing a wrong recipe can change system behavior.

### **WARNING**

#### **UNINTENDED EQUIPMENT OPERATION**

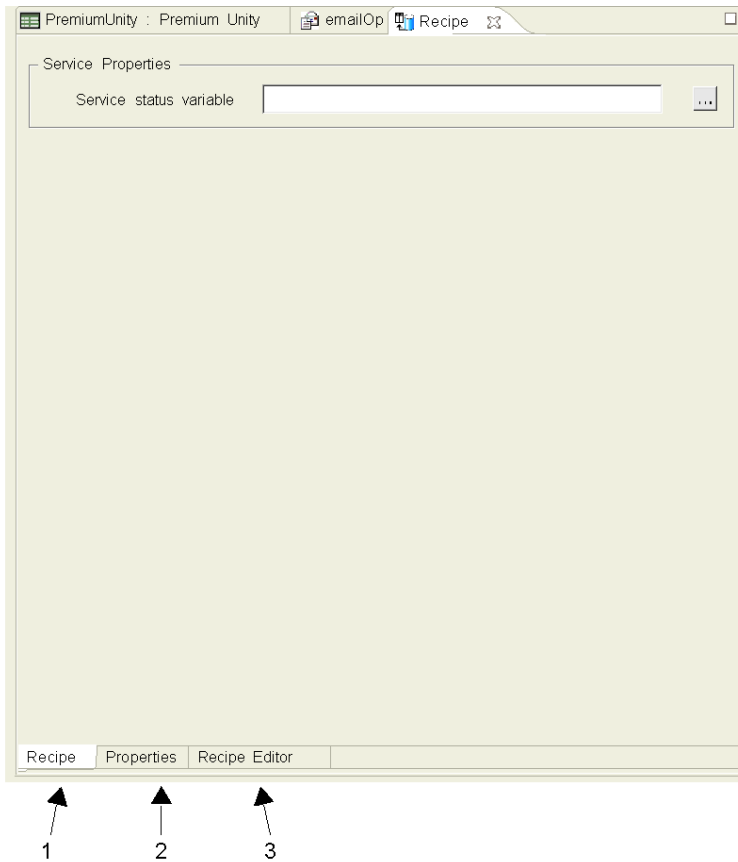
- Carefully select the symbols and direct addresses you authorize to be modified by the recipe.
- Do not authorize modification by the recipe of variables of critical nature concerning human and material safety.

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

#### **NOTE:**

- The recipe service is configured using Web Designer for TSX ETG 30\*\*.
- If an anomaly occurs, the status variable of the recipe service (*see page 264*) is set to **error**.

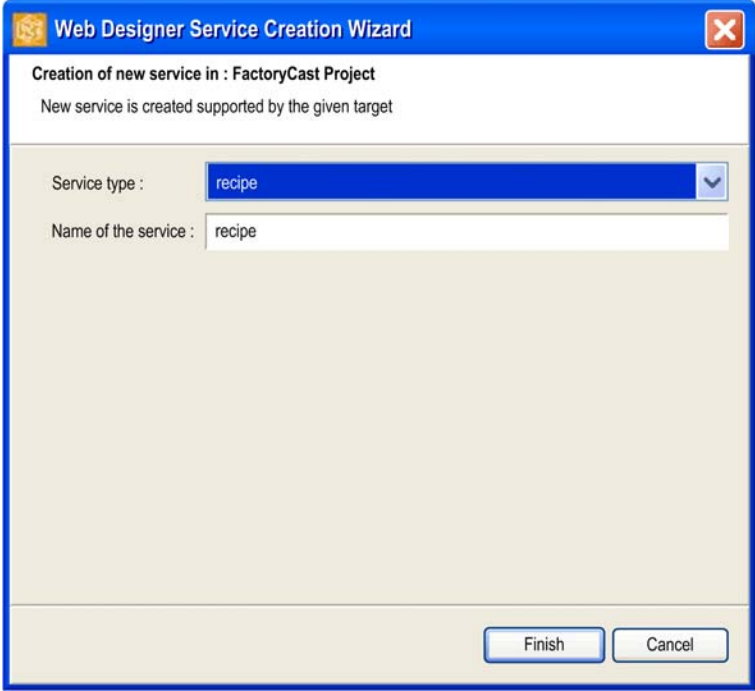
The Recipe service configuration screen is shown below:



The numbered references are as follows:

Reference	Function
1	View the "Recipe" service's status with this tab.
2	Configure recipes' properties with this tab.
3	Configure a recipe's values with this tab.

## How to Create a Recipe Service

Step	Action
1	Click on the name of your project in the browser.
2	<p>To add the new service you can:</p> <ul style="list-style-type: none"> <li>● click <b>Project</b> → <b>New</b> → <b>Service</b> or,</li> <li>● right-click <b>Services</b> in the browser, then click <b>New Service</b>.</li> </ul> <p><b>Result:</b> the following window appears.</p> 
3	Select <b>recipe</b> in the <b>Service Type</b> list box.
4	Enter the service name or click <b>Finish</b> to keep the name by default.
5	The Recipe service has been created.

## Configuring the Properties

Recipe Url	Trigger	Type	Status Variable	Remove
http://localhost/Recette.xml	plc.localRTDB.TrigRec1	NY	plc.localRTDB.Status	
http://localhost/Recette500.xml	plc.localRTDB.TrigRec2	RE	plc.localRTDB.Status	
http://139.160.67.44/Test.xml	plc.localRTDB.TrigRec3	FE	plc.localRTDB.Status	
http://localhost/Recette4.xml	plc.localRTDB.TrigRec4	BQ	plc.localRTDB.Status	

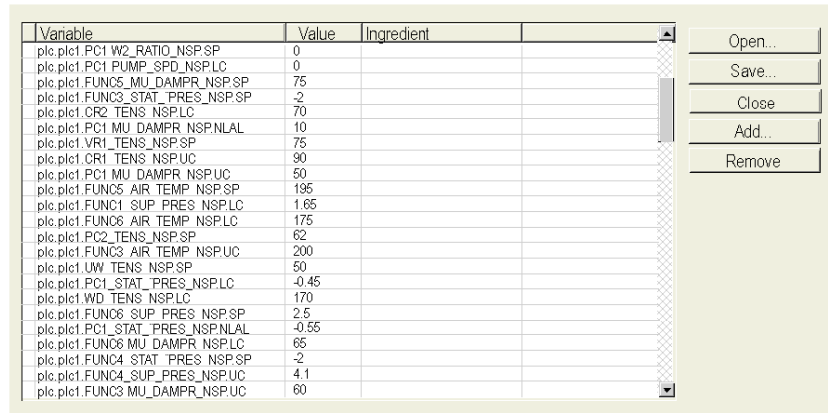
This table describes the various fields that make up the configuration screen:

Field	Function
Recipe URL	Location of the recipe file, with the extension <i>.xml</i> . If the file is stored in the module, the URL must be in the format <i>http://localhost/directory/filename.xml</i> , or <i>http://IP address of HTTP server/directory/filename.xml</i> .
Username	Username for connecting to the HTTP server.
Password	Password for connecting to the HTTP server.
Status Variable	Variable that describes the recipe's status: <ul style="list-style-type: none"> <li>● 0: no action</li> <li>● 1: recipe writing in progress</li> <li>● 60: detected error in recipe configuration</li> <li>● 61: detected error when writing the value</li> <li>● 62: communication detected error</li> <li>● 63: detected error in recipe file</li> </ul>
Trigger	Variable from which the event is triggered.

Field	Function	
Type	NY: Notify	Triggered on a bit or word value state change.
	RE: Rising Edge	Triggered on a rising edge for a bit, or a value increase for a word.
	FE: Falling Edge	Triggered on a falling edge for a bit, or a value decrease for a word.
	BQ: Bad Quality	Triggered if the trigger status is "Bad quality".
Remove	To remove a recipe from the list.	

### Recipe Editor

The Recipe Editor screen is used to change the recipe values. The resulting recipe file will be saved in XML format:



This table describes the "Recipe Editor":

Field	Function
Variable	Name of the variable to which a value is to be given.
Value	Value to attribute.
Ingredient	Optional comment to describe the variable's role.
Open	To open an XML-format recipe file.
Save	To save a recipe as an XML file.
Close	To close the Recipe Editor.
Add...	To add a variable.
Remove	To remove a variable.



## Adding Variables

Step	Action
1	Click on the Add button. <b>Result:</b> the variable selection box is displayed.
2	Select the variables to insert into the recipe file. Multiple variables may be selected by holding down the Ctrl or Shift keys while left-clicking.
3	Close the window with the Confirm button. <b>Result:</b> the selected variables are added to the recipe file.

## Saving a Recipe File

Step	Action
1	Click on the <b>Save</b> button. <b>Result:</b> the <b>Save as</b> dialog box is displayed.
2	The file must be saved on disk in order to be transferred to an HTTP server later. If you would like the file to remain local to the module, you must save it in a directory located in the website root of the project currently being configured. For example: <i>C:\...\workspace\project_name\services\recipes\my recipe.xml</i>
3	Close the window with the <b>Save</b> button. <b>Result:</b> the recipe file is saved

## Transferring a recipe file to a module

Step	Action
1	In the navigator, select the recipe file to transfer in the directory located under the website root ( <i>wwwroot</i> ).
2	Right-click on <b>Partial transfer -&gt; PC -&gt; Module</b> .

**NOTE:** For a recipe file that is not local to the module, you must first transfer it to the target PC server without using Web Designer for TSX ETG 30••.

## 5.9 Recipe Database Service

---

### Recipe Database Service

#### Presentation

The Recipe database service enables you to set the value of process variables to predetermined values stored in a database tables. In other words, you can create recipes or initialize application variables of your process with values stored in a database (SQLServer, MySql and Oracle).

You can also save the state of your process by storing the value of your application variables into database tables.

This can be achieved periodically, or when a specific event occurs (configured by the user).

Allowing write access can change system behavior.

The maximum number of Recipe database service is 1 per project.

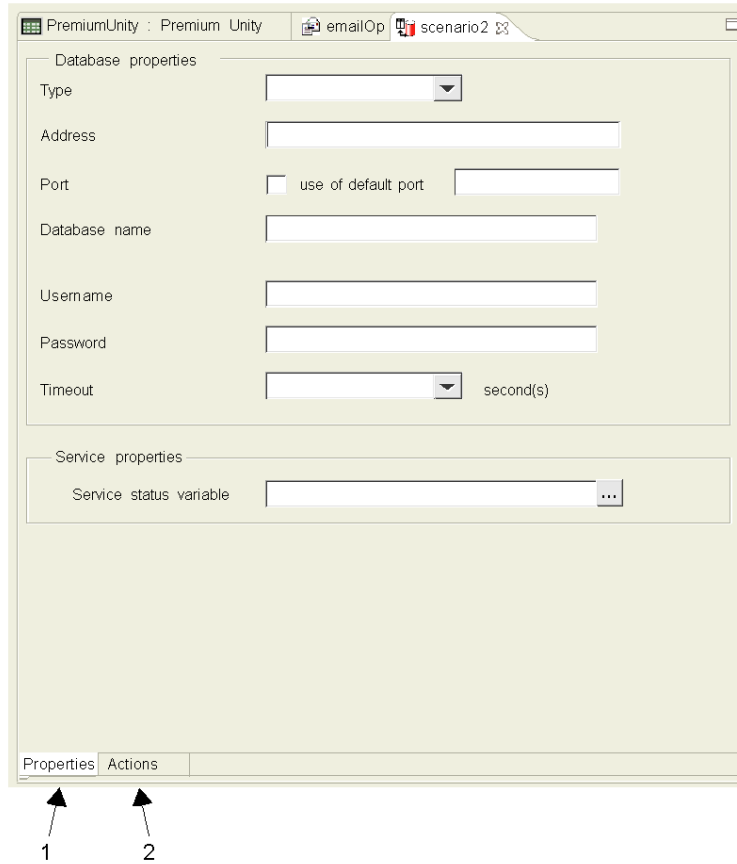
### WARNING

#### UNINTENDED EQUIPMENT OPERATION

- Carefully select the symbols and direct addresses you authorize to be modified by the recipe.
- Do not authorize modification of variables of critical nature concerning human and material safety by the recipe.

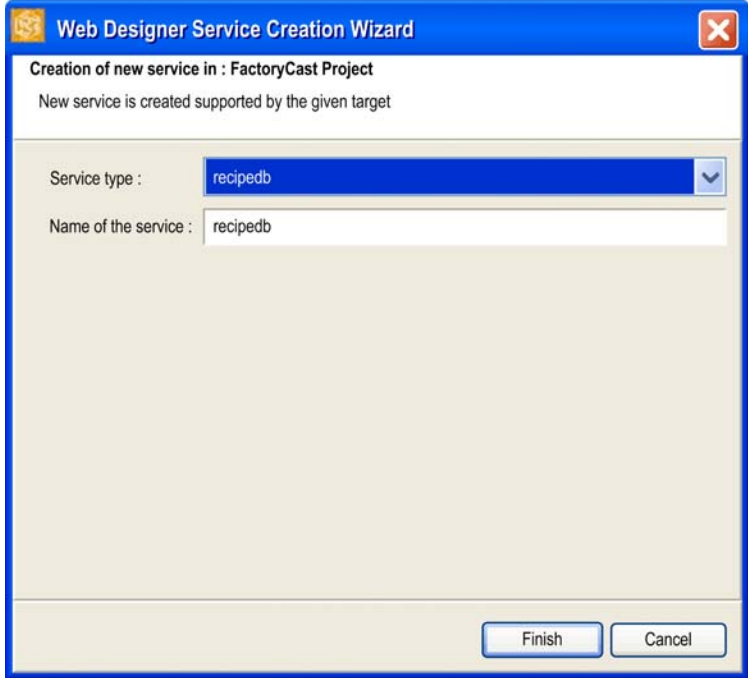
**Failure to follow these instructions can result in death, serious injury, or equipment damage.**

The Recipe database service configuration screen is shown below:



- 1 Configure the properties of the recipe database system with this tab.
- 2 Configure the actions to be undertaken with this tab.

## How to Create a Recipe Database Service

Step	Action
1	Click the name of your project in the browser.
2	<p>To add the new service you can:</p> <ul style="list-style-type: none"> <li>● click <b>Project</b> → <b>New</b> → <b>Service</b> or,</li> <li>● right-click <b>Services</b> in the browser, then click <b>New Service</b>.</li> </ul> <p><b>Result:</b> the following window appears.</p> 
3	Select <b>recipedb</b> in the <b>Service Type</b> dropdown list.
4	Enter the service name or click <b>Finish</b> to keep the name by default.
5	The recipe database service has been created.

## Configuration of Properties

The properties configuration window of the Database connection service is shown below:

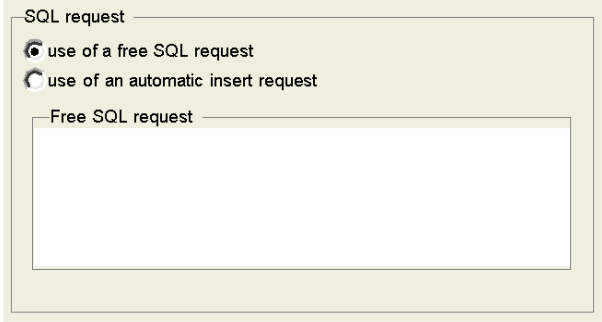
The following table describes the fields of the database server configuration screen:

Fields	Function
<b>Type</b>	<ul style="list-style-type: none"> <li>● <b>Oracle:</b> from version 8i to 10g or above</li> <li>● <b>MySql</b> from version 4.1 to 5.5 or above</li> <li>● <b>SQL Server:</b> from version 6.5 to 2008 R2 or above</li> </ul>
<b>Address</b>	IP address of the database server.
<b>Port</b>	Used to validate a default address.
<b>use of default port</b>	Port used by the database to access the server.
<b>Database name</b>	Name of the database that will contain all data configured by the user.
<b>User name and password</b>	Username and password (database connection string) to access the database server.
<b>Timeout</b>	Period of time that will be allowed to elapse before the system kills the current process when it improperly operates.
<b>Service status variable</b>	Used to determine the status of the database service.

## Configuring the Actions

This table describes the fields that make up the action screen:

Field	Function
<b>Name of the status action variable</b>	List the actions already configured.
<b>Action parameters</b>	
<b>Table name</b>	Type a name to identify the action.
<b>Action status variable</b>	Type the variable name that will represent the action status.
<b>Action type</b>	
<b>Database -&gt; PLC</b>	Initialization of your system variable with the value of the database.
<b>PLC -&gt; Database</b>	Save the current state of the system in the database.

Field	Function	
<b>Trigger criterion</b>		
<b>Trigger use</b>	Specify the variable that will trigger the action.	
	<b>NY:</b> Notify	Triggered on a bit or word value state change.
	<b>RE:</b> Rising Edge	Triggered on a rising edge for a bit, or a value increase for a word.
	<b>FE:</b> Falling Edge	Triggered on a falling edge for a bit, or a value decrease for a word.
	<b>BQ:</b> Bad Quality	Triggered if the trigger status is "Bad quality".
<b>Timer use</b>	Specify the period at which the action will be triggered.	
<b>SQL request</b>		
<b>Use of a free SQL request</b>	<p>Type your SQL request using the SQL syntax. In this case the screen looks like this:</p> 	
<b>Use of an automatic request</b>	<p>This function enables you to update tables without using a SQL syntax. Refer to the following paragraph for more information. Use the table provided for this purpose specifying the name of the column and the type of the variable. In this case, the column name syntax must comply with the database syntax. As a general rule, avoid using punctuation marks, upper case letters and hyphens (refer to your database documentation).</p> <p><b>Note:</b> The name of the action and the name of the table are identical if you use Automatic Request therefore you cannot read and write to the same table in that case. Use the free SQL request to precise the name of the table.</p>	
<b>Associated request for a free SQL request</b>	This part displays the SQL request syntax corresponding to the automatic request you have entered.	

### Automatic Request Example

In this example, you will update the block1 table in the database with PLC values when the state of the device.plc.control1 variable changes.

The following figure shows you the window that corresponds to the example:

**List of the database actions**

ACTION1	Add
	Remove
	Duplicate

**Actions parameters**

Table name: NEW TABLE0

Name of the action status variable: ACTION1

**Action type**

Database -> PLC     PLC -> Database

**Trigger criterion**

Trigger use    [ ]    NY

Timer use    20    second(s)

**SQL request**

use of a free SQL request



use of an automatic request

Column name	Variable	Add	Name of the index column
		Remove	MY_INDEX
			device.plc.index

Associated request for a free SQL request

Actions | Properties



Step	Action
1	Create a <b>recipe database</b> service.
2	Click the <b>Actions</b> tab.
3	Type the name of the table you want to update (i.e. block1) in the <b>Table name</b> field.
4	Specify a variable that will represent the action status (i.e. ACTION1) in the <b>Name of the action status variable</b> field.
5	Select <b>Database -&gt; PLC</b> in the <b>Action type</b> area.
6	Select use of a trigger in the <b>Trigger criterion</b> area and select the variable that will trigger the action by clicking  . You also have to specify the type of event that will trigger the action (i.e. NY).
7	Select <b>use of an automatic insert request</b> .
8	Click <b>Add</b> and type tankPressure for the column name. The column name syntax must comply with the database syntax.
9	Select the corresponding variable (i.e. device.plc.pres1) by clicking  on the left side of the <b>Variable</b> field.
10	Redo step 8 and 9 to add the values of device.plc.flow1 in the flow1 column.
11	Specify a name for the index of the table and select a variable to store the index value.

### Limitations

Web Designer for FactoryCast HMI only manages a single recipe database service.  
 The maximum number of tables is 20 per service.  
 The maximum number of columns is 50 per table.



---

# Embedded Website



---

## Scope of this Chapter

This section describes the embedded website of the TSX ETG 30•• module. You can use the website to configure, diagnose, control and monitor the module.

## What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
6.1	Overview of the Embedded Website	340
6.2	Setup Pages	346
6.3	Diagnostic Pages	362
6.4	Control Pages	386
6.5	Monitoring Pages	392

## 6.1 Overview of the Embedded Website

---

### Overview

This section presents the embedded website with factory settings of the TSX ETG 30\*\* module.

### What's in this Section?

This section contains the following topics:

Topic	Page
Embedded HTTP Server	341
<b>Home</b> Page for the HTTP Server	344
TSX ETG 30** <b>Version</b> Page	345

## Embedded HTTP Server

### Presentation

The TSX ETG 30\*\* module provides as standard an embedded HTTP server with a predefined factory built-in website. You can use the predefined pages of the website for module setup and control as well as application diagnostic and monitoring. They are 'ready to use' using a simple Web browser. No configuration or programming are required. The website is composed of standard Web pages in HTML format and Java applets for real-time and graphical animations.

#### NOTE:

- You can display the website either in French or in English. The applets are displayed in English.
- A Java environment plug-in must be installed on your PC in order to visualize and to use applets. The recommended Java environment is Java Runtime Environment (JRE) 1.4.2 or higher from SUN Microsystems (provided in the CD ROM).

Module functions:

Function	TSX ETG 30**
Number of browsers connected	16 maximum
Website embedded as standard	In the Flash memory
Internal memory reserved for creation of custom pages	32 MB

You can specify where to store the website:

- Flash memory (by default). File path: *FLASH1/wwwroot*.
- CF card. File path: *CFA00/wwwroot*.
- USB memory key. File path: *USBHD/00/wwwroot*.

#### NOTE:

- At module startup, websites stored on a CF card or a USB memory key will be launched instead of the default website.
- If your website is located on a USB memory key, the memory key must not be write protected.
- The TSX ETG 30\*\* stores *DataTables*, *GraphicScreens* and *Services* configuration files on the Flash memory of the module.
- You cannot configure *Services* with the website. Use Web Designer for TSX ETG 30\*\* to configure *Services* (see page 257).

## Default Web Server Functions

The following functions are available:

- diagnostic:
  - Ethernet and Modbus network statistics
  - RS232 modem connection statistics and diagnostics via log file
  - NTP statistics
  - FDR statistics
  - MIB upload
- setup:
  - security or password modification
  - TCP/IP and Modbus parameters configuration
  - RS232 modem and telephone parameters configuration
  - DNS, NTP and SNMP parameters configuration
  - FDR client configuration
  - module rebooting
- monitoring:
  - reading and editing Modbus device or module data
  - monitoring custom graphic pages

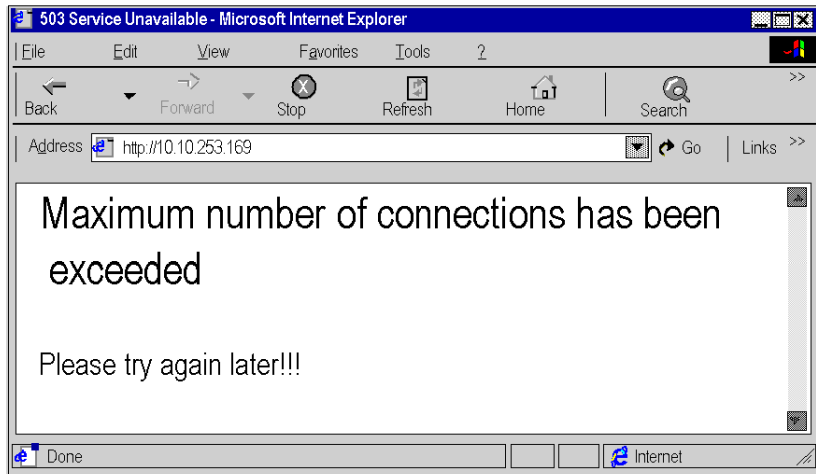
## HTTP Connections

The following connection rules must be observed:

- 1 connected Internet browser can open 2 connections, and the TSX ETG 30\*\* allows a maximum of 32 connections.
- Each HTTP connection closes automatically after one minute of inactivity.
- The connection remains active when passwords are being entered.

This means that up to 16 Internet browsers can be connected to one TSX ETG 30\*\* module.

When the maximum number of HTTP connections is reached, the following screen is displayed:



## Home Page for the HTTP Server

### Presentation

Use this page to access:

- the **Monitoring** page,
- the **Control** page,
- the **Diagnostics** page,
- the **Setup** page.

**NOTE:** On this page you can also choose the language you wish to use to navigate the various service pages and determine the product version.

### Accessing the Home Page

Step	Action
1	Open your usual browser.
2	Enter the IP address of the Ethernet module in the address bar.

### View of the Home Page



**NOTE:** The **Documentation** link provides an hyperlink to the documentation on the Schneider Electric website.



## TSX ETG 30•• Version Page

### Introduction

This page gives the version of the TSX ETG 30•• you are using and displays a view of the module and the firmware version.

### View of the Version Page

The screenshot displays the Schneider Electric FactoryCast™ TSX ETG 3021 web interface. The page features a green header with the Schneider Electric logo and the product name. Below the header is a navigation menu with tabs for Home, Documentation, Monitoring, Control, Diagnostics, Maintenance, and Setup. The main content area is titled "GATEWAY STATISTICS" and displays the following information:

- TSX ETG 3021 V1.50.6
- 12 Jan 2012 09:16:00
- Status indicators: ETH (green), RUN (green), ERR (red)
- Channels: CH0, CH1, CH2, BAT, EXT
- Inputs: IN0, IN1, IN2, IN3, IN4, IN5
- COM: COM, NC
- Outputs: OUT0, OUT1
- Voltage: +24V (red), 0V (red)

At the bottom of the page, the copyright notice reads: © 2000-2012 Schneider Electric. All Rights Reserved.

## 6.2 Setup Pages

---

### Overview

This section describes the pages of the HTTP server used to setup the TSX ETG 30\*\* module.

### What's in this Section?

This section contains the following topics:

<b>Topic</b>	<b>Page</b>
Setup Home Page	347
Security Configuration Page	348
Modbus Connection Configuration Page	349
Ethernet Services Configuration Page	350
<b>IP Filtering</b> Page	351
<b>UnitID Routing</b> Page	353
Modem Link Configuration Page	354
<b>Modem/PPP Security</b> Configuration Page	355
<b>Phone List</b> Configuration Page	356
<b>NAT</b> Configuration Page	357
<b>Stand-by</b> Configuration Page	358
DNS Configuration Page	359
NTP Configuration Page	360
SNMP Function Configuration Page	361

## Setup Home Page

### Introduction

This page lists the various setup services supported by the default Web server of the TSX ETG 30•• module and provides links for accessing the service you require.

### Accessing the Setup Page

Click the **Setup** link on the home page. Click one of the services offered on the **Setup** page.

### View of the Setup Page



To access the service you require, click one of the following links:

- **Security** to configure user names or passwords.
- **Modbus** to configure the Modbus serial connection.
- **IP Configuration** to set up the IP service.
- **IP Filtering** to configure the access rights of the IP service.
- **UnitID Routing** to configure the Modbus UnitID service.
- **Modem** to configure the modem service and its security. Several setup links are available under the Modem link: **Configuration**, **PPP Security**, **Phone list**, **NAT** and **Stand-by**.
- **DNS** to configure the DNS service.
- **NTP** to configure the NTP service.
- **SNMP** to configure the SNMP service.

## Security Configuration Page

### Introduction

Use this page to change the different passwords (*see page 79*) of the TSX ETG 30\*\* module.

**NOTE:** Write access is controlled by a password (**Security**) whose default value is **USER**.

### Accessing the Security Configuration Page

Step	Action
1	Click the <b>Setup</b> link on the Home page.
2	Click the <b>Security</b> link on the setup page.

### View of the Security Configuration page

HTTP Password	<input type="button" value="HTTP"/>
Configuration Password	<input type="button" value="Configuration"/>
Data Password	<input type="button" value="Data"/>

## Modbus Connection Configuration Page

### Introduction

Use this page to configure the Modbus function for the TSX ETG 30\*\* module.

**NOTE:** Write access is controlled by a password (**Security**) whose default value is **USER**.

The contents of this page are described in more detail in *Configuration Parameters for the Modbus Link*, page 99.

### Accessing the Modbus Configuration Page

Step	Action
1	Click the <b>Setup</b> link on the Home page.
2	Click the <b>Modbus</b> link on the setup page.

### View of the Modbus Configuration page

Serial line	
Baud rate	19200 ▼
Transmission mode	MODBUS RTU
Data bits	8
Parity	NONE ▼
Stop bits	1 ▼
Delay before new request (ms)	4 ▲▼
Response timeout (ms)	3000 ▲▼
Enable broadcast	<input type="checkbox"/>

## Ethernet Services Configuration Page

### Introduction

Use this page to configure Ethernet services for the TSX ETG 30•• module.

**NOTE:** Write access is controlled by a password (**Security**) whose default value is **USER**.

The contents of this page are described in more detail in *Configuration Parameters for TCP/IP Services*, page 121.

### Accessing the IP Configuration Page

Step	Action
1	Click the <b>Setup</b> link on the Home page.
2	Click the <b>IP configuration</b> link on the setup page.

### View of the IP Configuration page

Ethernet parameters

Ethernet frame format

Link speed and duplex

IP parameters

Served from MAC address  
 Served from device name  
 Local

IP address  .  .  .

Subnet mask  .  .  .

Default gateway  .  .  .

FDR parameters

Device name

Replication period (mn)

## IP Filtering Page

### Introduction

Use this page to configure the IP filtering service for the TSX ETG 30\*\* module.

Use this page to:

- Configure the number of connections that can be opened by the module.
- Activate an access control service.
- Enter the remote devices which can connect to the module, depending on whether a mono- or multi-connection communication protocol is used.

**NOTE:** Write access is controlled by a password (**Security**) whose default value is **USER**.

The contents of this page are described in more detail in IP Filtering Configuration (*see page 123*).

**NOTE:** This function filters only the TCP Modbus communication on the 502 port (not http, ftp,...).

### Accessing the IP Filtering Page

Step	Action
1	Click the <b>Setup</b> link on the <b>Home</b> page.
2	Click the <b>IP filtering</b> link on the <b>Setup</b> page.

### View of the IP Filtering page

Access control

Connection configuration			
	IP address	Access	Mode
1	10.10.0.1	<input checked="" type="checkbox"/>	MULTI
2	10.10.0.2	<input checked="" type="checkbox"/>	MULTI
3	10.10.0.3	<input type="checkbox"/>	MONO
4	10.10.0.4	<input checked="" type="checkbox"/>	MULTI
5		<input checked="" type="checkbox"/>	MULTI
6		<input checked="" type="checkbox"/>	MULTI
7		<input checked="" type="checkbox"/>	MULTI
8		<input checked="" type="checkbox"/>	MULTI
9		<input checked="" type="checkbox"/>	MULTI
10		<input checked="" type="checkbox"/>	MULTI

Apply      Undo



## UnitID Routing Page

### Introduction

Use this page to configure the Modbus UnitID Routing function for the TSX ETG 30•• module. With the Modbus UnitID Routing function, the computer connected to the Internet can access the remote device.

**NOTE:** Write access is controlled by a password (**Security**) whose default value is **USER**.

The contents of this page are described in more detail in Modbus UnitID Routing Configuration (*see page 194*) page.

### Accessing the UnitID Routing Page

Step	Action
1	Click the <b>Setup</b> link on the <b>Home</b> page.
2	Click the <b>UnitID Routing</b> link on the <b>Setup</b> page.

### View of the UnitID Routing page

UnitID Routing			
	Source UnitID	Destination IP address	Destin. UnitID
1	101	139.168.10.20	255
2	102	139.168.10.23	255
3	103	139.168.10.24	255
4			
5			
6			

## Modem Link Configuration Page

### Introduction

Use this page to configure the modem services for the TSX ETG 30•• module.

**NOTE:** Write access is controlled by a password (**Security**) with the default value **USER**.

The contents of this page are described in more detail in *Configuring the PSTN, GSM, or External Modem, page 157*.

### Accessing the Modem Configuration Page

Step	Action
1	Click the <b>Setup</b> link on the Home page.
2	Click the <b>Modem</b> link on the setup page.

**NOTE:** Type the PIN code of the SIM card carefully: if you supply an incorrect PIN code 3 times, the SIM card is locked. If this happens, call your service provider to unlock it.

## Modem/PPP Security Configuration Page

### Introduction

Use this page to configure the modem PPP Security services for the TSX ETG 30 module. This service enables limited access to the module services to authorized devices.

**NOTE:** Write access is controlled by a password (**Security**) whose default value is **USER**.

The content of this page is described in more detail in Configuring the PSTN, GSM or external modem (*see page 174*).

### Accessing the PPP Security Page

Step	Action
1	Click the <b>Setup</b> link on the <b>Home</b> page.
2	Click the <b>PPP Security</b> link on the <b>Setup</b> page.

### View of the PPP Security Page

IP control enable

IP Range authorized		
	From IP address	To IP address
1	62.44.4.0	62.44.1.255
2	80.10.23.100	80.10.23.100
3	62.117.0.0	62.117.255.255
4		

VPN enable

VPN Connections						
	Remote add...	Pre shared key	Mode	Remote LAN	Subnet mask	ETG client...
1	ETG.dyndns.o.	*****	Transport			
2	62.12.123.100	*****	Tunnel	192.168.2.0	255.255.255.0	High
3						
4						

© 2000-2009 Schneider Electric. All Rights Reserved.

**NOTE:** Only one type of security can be selected at the same time. The unselected table is grayed out.

## Phone List Configuration Page

### Introduction

Use this page to configure the phone services for the TSX ETG 30•• module. This service enable you to limit access to the module services to authorized devices.

**NOTE:** Write access is controlled by a password (**Security**) whose default value is **USER**.

The contents of this page are described in more detail in Phone List Configuration (see page 216).

### Accessing the Phone List Configuration Page

Step	Action
1	Click the <b>Setup</b> link on the <b>Home</b> page.
2	Click the <b>Phone list</b> link on the <b>Setup</b> page.

### View of the Phone List Configuration Page

Client: Local PPP Address

Served     
  Fixed     
  TCP/IP address

Phone Number

	Phone number	Destination	User	Password
1	90493653068	10.10.0.11	USER	****
2	0681936569	10.10.0.12	USER	****
3	2601	10.10.0.13	USER	****
4	0493653726	10.10.0.14	USER	****
5	0492382668	10.10.0.15	USER	****
6				
7				
8				
9				
10				

## NAT Configuration Page

### Introduction

Use this page to configure the NAT service for the TSX ETG 302• module. This service enables limited access to the module services available to authorized devices only.

**NOTE:** Write access is controlled by a password (**Security**) with the default value **USER**.

The contents of this page are described in more detail in NAT on Port Configuration Page (see page 188).

### Accessing the NAT Configuration Page

Step	Action
1	Click the <b>Setup</b> link on the <b>Home</b> page.
2	Click the <b>NAT</b> link on the <b>Setup</b> page.

### View of the NAT Configuration Page

Enable

Network Address Port Translation				
	Protocol	Source Port	Destination Address	Destin. Port
1	TCP ▼	5502	139.158.20.23	502 ▲
2	TCP ▼	5080	139.158.20.23	80
3	TCP ▼	6502	139.158.20.24	502
4	▼			
5	▼			
6	▼			
7	▼			
8	▼			
9	▼			
10	▼			▼

## Stand-by Configuration Page

### Introduction

Use this page to configure the Stand-by service for the TSX ETG 302• module. This service enables limited access to the module services available to authorized devices only.

**NOTE:** Write access is controlled by a password (**Security**) with the default value **USER**.

The content of this page are described in more detail in Stand-by Configuration (see page 226).

### Accessing the Stand-by Configuration Page

Step	Action
1	Click the <b>Setup</b> link on the <b>Home</b> page.
2	Click the <b>Stand-by</b> link on the <b>Setup</b> page.

### View of the Stand-by Configuration Page

Enable

**Connections**

Primary **ETHERNET** Backup **INTERNAL MODEM**

**Ethernet Link Monitoring**

Monitoring period **3mn**  Switch back to Ethernet

Nb of Retry **0**

**Ethernet Link Checking**

IP address ping **92.123.45.10**

Http Get address

Ethernet port 1  Ethernet port 2

**Modem Link Monitoring**

GPRS attachment check Monitoring period **1h**

## DNS Configuration Page

### Introduction

Use this page to configure the DNS service for the TSX ETG 30•• module.

**NOTE:** Write access is controlled by a password (**Security**) whose default value is **USER**.

The contents of this page are described in more detail in *DNS Service Configuration*, page 236.

### Accessing the DNS Configuration Page

Step	Action
1	Click the <b>Setup</b> link on the Home page.
2	Click the <b>DNS</b> link on the setup page.

### View of the DNS Configuration page

**DNS Servers**

Domain Name Server  .  .  .

**Domain Name**

Domain Name

## NTP Configuration Page

### Introduction

Use this page to configure the NTP service for the TSX ETG 30••module. Once activated, the date and time are provided to the TSX ETG 30•• by a date time server (NTP server) over a TCP/IP link.

**NOTE:** Write access is controlled by a password (**Security**) whose default value is **USER**.

The contents of this page are described in more detail in *NTP Service Configuration*, page 239.

### Accessing the NTP Configuration Page

Step	Action
1	Click the <b>Setup</b> link on the Home page.
2	Click the <b>NTP</b> link on the setup page.

### View of the NTP Configuration page

**NTP Server Parameters**

Primary NTP server

Secondary NTP server

Polling period (s)  ▼

**Time Zone**

▼

Automatically adjust clock for daylight saving change



## SNMP Function Configuration Page

### Introduction

Use this page to configure the SNMP function for the TSX ETG 30\*\* module.

**NOTE:** Write access is controlled by a password (**Security**) whose default value is **USER**.

The contents of this page are described in more detail in *SNMP Service Configuration, page 244*.

### Accessing the SNMP Configuration Page

Step	Action
1	Click the <b>Setup</b> link on the Home page.
2	Click the <b>SNMP</b> link on the setup page.

### View of the SNMP Configuration page

IP address managers	
Manager 1	<input type="text" value="10.10.0.20"/>
Manager 2	<input type="text" value="10.10.0.50"/>

Agent	
System location	<input type="text" value="servers room"/>
System contact	<input type="text" value="admin@schneider.com"/>

Community names	
Read-Write	<input type="text" value="public"/>
Read-Only	<input type="text" value="public"/>
Trap	<input type="text" value="public"/>

Enable authentication failure trap

## 6.3 Diagnostic Pages

---

### Overview

This section describes the diagnostic pages of the TSX ETG 30\*\* module.

### What's in this Section?

This section contains the following topics:

Topic	Page
<b>Diagnos</b> tics Home Page	363
<b>Gateway Statistics</b> Page	365
<b>Ethernet Statistics</b> Page	366
Modbus Statistics Page	368
<b>PPP/Modem Statistics</b> Page	370
<b>PPP/Modem Log File</b> Page	374
<b>PPP/Modem VPN</b> Page	376
<b>PPP/Modem Stand-by</b> Page	378
FDR (Device Replacement) Statistics Page	380
NTP Statistics Page	382
MIB Upload Page	384
<b>Device Explorer</b> Page	385

## Diagnostics Home Page

### Home Page

This page lists the various diagnostic services supported by the Web server of the TSX ETG 30 module and provides links for accessing the service you require.

### Accessing the Diagnostics Page

Click the link on the **Home** page. Click one of the services offered on the **Diagnostics** home page.

### View of the Diagnostics Home Page



**Schneider Electric** **FactoryCast™ TSX ETG 3021**

Home Documentation URL

Monitoring Control **Diagnostics** Maintenance Setup

**Diagnostics**

- Gateway Statistics
- Ethernet Statistics**
  - Global
  - Port
- Modbus Statistics**
  - Serial Line
  - TCP/IP Messaging
- PPP Modem**
  - Statistics
  - Log File
  - VPN
  - Stand-by
- FDR Statistics
- NTP
- MIB Upload
- Device Explorer

Web site version : 1.50  
© 2000-2012 Schneider Electric. All Rights Reserved.

To access the service you require, click one of the following links:

- **Gateway Statistics:** To find out about managing the diagnostics counters for gateway communications.
- **Ethernet Statistics:** To find out about managing the diagnostics counters for Ethernet communications.
- **Modbus Statistics:** To find out about managing timeout and CRC Modbus message diagnostics counters.

- **PPP Modem:** To provide information about PPP communication. Several information are available under the **PPP Modem** link: **Statistics, Log File, VPN** and **Stand-by**.
- **FDR Statistics:** To find out about managing the diagnostic parameters and counters for the FDR service.
- **NTP:** To find out about managing the diagnostics counters for the NTP communication.
- **MIB Upload:** To load the MIB from the module to the PC.
- **Device Explorer:** To provide status and diagnostic information about the devices connected to the module.

## Gateway Statistics Page

### Introduction

The **Gateway Statistics** page displays a view of the led indicators as well as the current date and time of the module. It provides a first level diagnostic.

### Accessing the Gateway Statistics Page

Step	Action
1	Click the <b>Diagnostics</b> link on the <b>Home</b> page.
2	Click the <b>Gateway statistics</b> link on the <b>Diagnostics</b> page.

### View of the Gateway Statistics Page

The following table describes the elements of the gateway statistics page:

Element	Description
Date and time	Indicate the current date and time of the module.
Led	Display a view of the module LED indicators ( <i>see page 68</i> ).
I/O board	Display the status of the I/O board (not available for the TSX ETG 3000).

## Ethernet Statistics Page

### Introduction

This page provides statistics about the Ethernet network.

### Accessing the Ethernet Statistics Pages Page

Step	Action
1	Click the <b>Diagnostics</b> link on the <b>Home</b> page.
2	Click the <b>Ethernet Statistics</b> link on the <b>Diagnostics</b> page.
3	<ul style="list-style-type: none"> <li>Click the <b>Global</b> link on the <b>Diagnostics</b> page to see general statistics about the Ethernet link.</li> <li>Click the <b>Port</b> link on the <b>Diagnostics</b> page to see port statistics.</li> </ul>

### View of the Global Ethernet Statistics

<table border="1"> <thead> <tr> <th colspan="2">TCP/IP parameters</th> </tr> </thead> <tbody> <tr> <td>IP address</td> <td>10.10.253.169</td> </tr> <tr> <td>Subnet mask</td> <td>255.0.0.0</td> </tr> <tr> <td>Default gateway</td> <td>0.0.0.0</td> </tr> </tbody> </table>		TCP/IP parameters		IP address	10.10.253.169	Subnet mask	255.0.0.0	Default gateway	0.0.0.0	<table border="1"> <thead> <tr> <th colspan="2">Receive statistics</th> </tr> </thead> <tbody> <tr> <td>Frame received</td> <td>452</td> </tr> </tbody> </table>		Receive statistics		Frame received	452
TCP/IP parameters															
IP address	10.10.253.169														
Subnet mask	255.0.0.0														
Default gateway	0.0.0.0														
Receive statistics															
Frame received	452														
<table border="1"> <thead> <tr> <th colspan="2">Ethernet parameters</th> </tr> </thead> <tbody> <tr> <td>MAC address</td> <td>00-80-F4-01-FD-A9</td> </tr> </tbody> </table>		Ethernet parameters		MAC address	00-80-F4-01-FD-A9	<table border="1"> <thead> <tr> <th colspan="2">Transmit statistics</th> </tr> </thead> <tbody> <tr> <td>Frame transmitted</td> <td>209</td> </tr> </tbody> </table>		Transmit statistics		Frame transmitted	209				
Ethernet parameters															
MAC address	00-80-F4-01-FD-A9														
Transmit statistics															
Frame transmitted	209														
<input type="button" value="Reset counters"/>															

The following table describes the elements of the gateway statistics page:

Element	Description
<b>TCP/IP parameters</b>	Display the IP address, subnet mask and default gateway address of the module.
<b>Ethernet parameters</b>	Display the MAC address of the module.
<b>Receive statistics</b>	Indicate the number of frames correctly received via Ethernet by the TSX ETG 30**.
<b>Transmit statistics</b>	Indicate the number of frames correctly transmitted via Ethernet by the TSX ETG 30**.
<b>Reset counters</b>	Set all the counters to 0.

## View of the Ethernet Port Statistics

<b>Port Number</b> 1	<b>Link</b> Link status link down
<b>Transmit statistics</b> Frames transmitted OK 0	<b>Received statistics</b> Frames received OK 0
<b>Port Number</b> 2	<b>Link</b> Link status 100BaseTX-FD link up
<b>Transmit statistics</b> Frames transmitted OK 239	<b>Received statistics</b> Frames received OK 499
<input type="button" value="Reset counters"/>	

The following table describes the elements of the Ethernet ports statistics page:

Element	Description
<b>Port number</b>	This page shows Ethernet ports statistics for 2 ports.
<b>Link</b>	Display the status of the Ethernet link.
<b>Received statistics</b>	Indicate the number of frame received via Ethernet by the TSX ETG 30•• without error.
<b>Transmit statistics</b>	Indicate the number of frame transmit via Ethernet by the TSX ETG 30•• without error.
<b>Reset counters</b>	Set all the counters to 0.

## Modbus Statistics Page

### Introduction

This page provides statistics about the Modbus network.

### Accessing the Modbus Statistics Pages

Step	Action
1	Click the <b>Diagnostics</b> link on the Home page.
2	Click the <b>Modbus statistics</b> link on the diagnostics page.
3	<ul style="list-style-type: none"> <li>Click the <b>Serial line</b> link on the diagnostics page to see general statistics about the serial link.</li> <li>Click the <b>TCP/IP messaging</b> link on the diagnostics page to see Modbus messaging statistics.</li> </ul>

### View of the Serial Line Statistics Page

Low-level statistics	
Sent frames	556
Received frames	320
CRC errors	6
Application-level statistics	
Timeouts errors	5

The following table describes the elements of the serial line statistics page:

Element	Description
Low-level statistics	
Sent frames	Number of frames sent via the serial line.
Received frames	Number of frames received via the serial line.
CRC detected errors	Number of responses received with CRC detected errors.
Application-level statistics	
Timeout detected errors	Number of Modbus requests received with no response.
Reset	Set all the counters to 0.



### View of the TCP/IP Messaging Statistics Page

TCP Connection		Inbound / Outbound statistics					
Max TCP connections	<input type="text" value="64"/>	Opened TCP connections	<input type="text" value="0"/>	Received Modbus msg	<input type="text" value="0"/>	Sent Modbus msg	<input type="text" value="0"/>

Connection table						
Connection	Remote IP addr	Remote Port	Local Port	Sent Msg	Received Msg	Errors Sent

The following table describes the elements of the TCP/IP messaging statistics page:

Element	Description
TCP connection	Indicate the maximum TCP connections that can be opened at the same time.
Inbound / Outbound statistics	
Opened TCP connections	Number of opened TCP connections.
Received Modbus msg	Number of Modbus messages received by the module.
Sent Modbus msg	Number of Modbus messages sent by the module.
Connection table	This table displays connection information for all the remote devices connected to the module.
Reset counters	Set all the counters to 0.

## PPP/Modem Statistics Page

### Introduction

This page provides information about the **PPP/Modem Statistics** page.

### Accessing the PPP/Modem Statistics Page

Step	Action
1	Click the <b>Diagnostics</b> link on the <b>Home</b> page.
2	Click the <b>PPP/Modem</b> link on the <b>Diagnostics</b> page.
3	Click the <b>Statistics</b> link on the diagnostics page to see general statistics about the PPP modem connection.

### View of the PPP/Modem Statistics Page

If you are using a TSX ETG 3000 or a TSX ETG 3010 module, the **PPP/Modem Statistics** page looks like this:

<table border="1"> <thead> <tr> <th colspan="2">Status</th> </tr> </thead> <tbody> <tr> <td>Modem</td> <td>INTERNAL</td> </tr> <tr> <td>Connection</td> <td>INACTIVE</td> </tr> <tr> <td>Phone number</td> <td></td> </tr> </tbody> </table> <table border="1"> <thead> <tr> <th colspan="2">IP address</th> </tr> </thead> <tbody> <tr> <td>Local PPP address</td> <td></td> </tr> <tr> <td>Remote PPP address</td> <td></td> </tr> </tbody> </table> <table border="1"> <thead> <tr> <th colspan="2">Received</th> </tr> </thead> <tbody> <tr> <td>Frames received</td> <td>0</td> </tr> </tbody> </table>	Status		Modem	INTERNAL	Connection	INACTIVE	Phone number		IP address		Local PPP address		Remote PPP address		Received		Frames received	0	<table border="1"> <thead> <tr> <th colspan="2">Server</th> </tr> </thead> <tbody> <tr> <td>Connections accepted</td> <td>4</td> </tr> </tbody> </table> <table border="1"> <thead> <tr> <th colspan="2">Client</th> </tr> </thead> <tbody> <tr> <td>Connections opened OK</td> <td>0</td> </tr> <tr> <td>Connections opened error</td> <td>0</td> </tr> <tr> <td>Connections closed OK</td> <td>0</td> </tr> <tr> <td>Connections closed error</td> <td>0</td> </tr> </tbody> </table> <table border="1"> <thead> <tr> <th colspan="2">Transmitted</th> </tr> </thead> <tbody> <tr> <td>Frames transmitted</td> <td>0</td> </tr> </tbody> </table>	Server		Connections accepted	4	Client		Connections opened OK	0	Connections opened error	0	Connections closed OK	0	Connections closed error	0	Transmitted		Frames transmitted	0
Status																																					
Modem	INTERNAL																																				
Connection	INACTIVE																																				
Phone number																																					
IP address																																					
Local PPP address																																					
Remote PPP address																																					
Received																																					
Frames received	0																																				
Server																																					
Connections accepted	4																																				
Client																																					
Connections opened OK	0																																				
Connections opened error	0																																				
Connections closed OK	0																																				
Connections closed error	0																																				
Transmitted																																					
Frames transmitted	0																																				
<input type="button" value="Reset counters"/>																																					

If you are using a TSX ETG 302• module with GSM, the PPP/Modem statistics page looks like this:

<table border="1"> <thead> <tr><th>Status</th></tr> </thead> <tbody> <tr><td>Modem</td><td>INTERNAL</td></tr> <tr><td>Connection</td><td>INACTIVE</td></tr> <tr><td>Phone number</td><td></td></tr> </tbody> </table> <table border="1"> <thead> <tr><th>IP address</th></tr> </thead> <tbody> <tr><td>Local PPP address</td><td></td></tr> <tr><td>Remote PPP address</td><td></td></tr> </tbody> </table> <table border="1"> <thead> <tr><th>Received</th></tr> </thead> <tbody> <tr><td>Frames received</td><td>0</td></tr> </tbody> </table> <table border="1"> <thead> <tr><th>GSM</th></tr> </thead> <tbody> <tr><td>SIM card</td><td>INSERTED</td></tr> <tr><td>Pin code</td><td>OK</td></tr> <tr><td>Operator</td><td>Orange F</td></tr> <tr><td>Signal level (0..100%)</td><td><div style="width: 0%;"></div></td></tr> </tbody> </table>	Status	Modem	INTERNAL	Connection	INACTIVE	Phone number		IP address	Local PPP address		Remote PPP address		Received	Frames received	0	GSM	SIM card	INSERTED	Pin code	OK	Operator	Orange F	Signal level (0..100%)	<div style="width: 0%;"></div>	<table border="1"> <thead> <tr><th>Server</th></tr> </thead> <tbody> <tr><td>Connections accepted</td><td>0</td></tr> </tbody> </table> <table border="1"> <thead> <tr><th>Client</th></tr> </thead> <tbody> <tr><td>Connections opened OK</td><td>1</td></tr> <tr><td>Connections opened error</td><td>0</td></tr> <tr><td>Connections closed OK</td><td>1</td></tr> <tr><td>Connections closed error</td><td>0</td></tr> </tbody> </table> <table border="1"> <thead> <tr><th>Transmitted</th></tr> </thead> <tbody> <tr><td>Frames transmitted</td><td>0</td></tr> </tbody> </table>	Server	Connections accepted	0	Client	Connections opened OK	1	Connections opened error	0	Connections closed OK	1	Connections closed error	0	Transmitted	Frames transmitted	0
Status																																								
Modem	INTERNAL																																							
Connection	INACTIVE																																							
Phone number																																								
IP address																																								
Local PPP address																																								
Remote PPP address																																								
Received																																								
Frames received	0																																							
GSM																																								
SIM card	INSERTED																																							
Pin code	OK																																							
Operator	Orange F																																							
Signal level (0..100%)	<div style="width: 0%;"></div>																																							
Server																																								
Connections accepted	0																																							
Client																																								
Connections opened OK	1																																							
Connections opened error	0																																							
Connections closed OK	1																																							
Connections closed error	0																																							
Transmitted																																								
Frames transmitted	0																																							

If you are using a TSX ETG 302• module with GPRS, the PPP/Modem statistics page looks like this:

<table border="1"> <thead> <tr><th>Status</th></tr> </thead> <tbody> <tr><td>Modem</td><td>INTERNAL</td></tr> <tr><td>Connection</td><td>CLIENT</td></tr> <tr><td>Phone number</td><td>GPRS / 3G+</td></tr> </tbody> </table> <table border="1"> <thead> <tr><th>IP address</th></tr> </thead> <tbody> <tr><td>Local PPP address</td><td>10.10.20.100</td></tr> <tr><td>Remote PPP address</td><td>GPRS / 3G+Network</td></tr> </tbody> </table> <table border="1"> <thead> <tr><th>Received</th></tr> </thead> <tbody> <tr><td>Frames received</td><td>244</td></tr> </tbody> </table> <table border="1"> <thead> <tr><th>GSM</th></tr> </thead> <tbody> <tr><td>SIM card</td><td>INSERTED</td></tr> <tr><td>Pin code</td><td>***</td></tr> <tr><td>Operator</td><td>Orange F</td></tr> <tr><td>Signal level (0..100%)</td><td><div style="width: 0%;"></div></td></tr> </tbody> </table>	Status	Modem	INTERNAL	Connection	CLIENT	Phone number	GPRS / 3G+	IP address	Local PPP address	10.10.20.100	Remote PPP address	GPRS / 3G+Network	Received	Frames received	244	GSM	SIM card	INSERTED	Pin code	***	Operator	Orange F	Signal level (0..100%)	<div style="width: 0%;"></div>	<table border="1"> <thead> <tr><th>Server</th></tr> </thead> <tbody> <tr><td>Connection accepted</td><td>2</td></tr> </tbody> </table> <table border="1"> <thead> <tr><th>Client</th></tr> </thead> <tbody> <tr><td>Connections opened OK</td><td>1</td></tr> <tr><td>Connections opened error</td><td>0</td></tr> <tr><td>Connections closed OK</td><td>1</td></tr> <tr><td>Connections closed error</td><td>0</td></tr> </tbody> </table> <table border="1"> <thead> <tr><th>Transmitted</th></tr> </thead> <tbody> <tr><td>Frames transmitted</td><td>211</td></tr> </tbody> </table>	Server	Connection accepted	2	Client	Connections opened OK	1	Connections opened error	0	Connections closed OK	1	Connections closed error	0	Transmitted	Frames transmitted	211
Status																																								
Modem	INTERNAL																																							
Connection	CLIENT																																							
Phone number	GPRS / 3G+																																							
IP address																																								
Local PPP address	10.10.20.100																																							
Remote PPP address	GPRS / 3G+Network																																							
Received																																								
Frames received	244																																							
GSM																																								
SIM card	INSERTED																																							
Pin code	***																																							
Operator	Orange F																																							
Signal level (0..100%)	<div style="width: 0%;"></div>																																							
Server																																								
Connection accepted	2																																							
Client																																								
Connections opened OK	1																																							
Connections opened error	0																																							
Connections closed OK	1																																							
Connections closed error	0																																							
Transmitted																																								
Frames transmitted	211																																							

The following table describes the standard elements of the PPP/Modem statistics page:

Element	Description
<b>Status</b>	
<b>Modem</b>	Indicate the status of the modem connection.
<b>Link</b>	Indicate the status of the PPP link.
<b>Phone number</b>	Indicate the status of the telephone connection (if you are using GPRS, this field is hidden).
<b>IP address</b>	
<b>Local PPP address</b>	PPP IP address of the local connection (module). If the PPP connection is closed IP=blank.
<b>Remote IP address</b>	PPP IP address of the remote connection. If the PPP connection is closed IP=blank. If you are using GPRS, this field is set to <b>GPRS Network</b> .
<b>Received</b>	
<b>Frames received</b>	Number of frames received through the PPP/modem link.
<b>Server</b>	
<b>Connections accepted</b>	Number of connections accepted by the server.
<b>Client</b>	
<b>Open connection OK</b>	Number of PPP connections correctly opened.
<b>Open connection detected error</b>	Number of PPP client connections opened with anomalies (IP address detected error, no response from modem, line busy, etc.).
<b>Close connection OK</b>	Number of PPP connections correctly closed.
<b>Close connection detected error</b>	Number of PPP client connections closed with anomalies (IP address detected error, no response from modem, line busy, etc.).
<b>Transmitted</b>	
<b>Frames transmitted</b>	Number of frames sent through the PPP/Modem connection.

The following table describes the GSM/GPRS specific elements of the PPP/Modem statistics page:

Element	Description
<b>GSM</b>	
<b>SIM card</b>	Status of the SIM card: <ul style="list-style-type: none"> <li>● <b>INSERTED</b>: the SIM card is inserted.</li> <li>● <b>NOT INSERTED</b>: the SIM card is not inserted or not inserted properly.</li> <li>● <b>BLOCKED</b>: the SIM card has been locked due to 3 incorrect PIN code attempts (see note below the table).</li> </ul>
<b>PIN code</b>	Status of the PIN code authentication: <ul style="list-style-type: none"> <li>● <b>NOT ATTEMPTED</b>: the SIM card is not inserted or the SIM card is locked due to authentication anomalies (see note below the table).</li> <li>● <b>FAILED X TIME</b>: the PIN code authentication failed X times (see note below the table).</li> <li>● <b>OK</b>: the PIN code supplied is OK.</li> </ul>
<b>Operator</b>	Name of the telephone operator.
<b>Signal level (0...100%)</b>	Strength of the telephone signal.

**NOTE:** Type the PIN code of SIM card carefully. If you supply an incorrect PIN code 3 times, the SIM card is locked. If this happens, call your GSM service provider to unlock it.

## PPP/Modem Log File Page

### Introduction

This page provides information about the **PPP/Modem Log File** page.

### Accessing the PPP/Modem Log File Page

Step	Action
1	Click the <b>Diagnostics</b> link on the <b>Home</b> page.
2	Click the <b>PPP/Modem</b> link on the <b>Diagnostics</b> page.
3	Click the <b>Log file</b> link to display a text file showing the last Hayes commands of the modem.

### View of the Log File Page

The log file lists the last 500 Hayes commands made by the modem as well as information about the PPP link status.

```
[04 May 2007 20:10:00]modem: command = AT&F
[04 May 2007 20:10:00]modem: response = AT&F
[04 May 2007 20:10:00]modem: response = OK
[04 May 2007 20:10:00]modem: command = ATE1&D2S0=1+CMEE=1
[04 May 2007 20:10:00]modem: response = ATE1&D2S0=1+CMEE=1
[04 May 2007 20:10:01]modem: response = OK
[04 May 2007 20:10:01]modem: command = AT+IFC=0,0
[04 May 2007 20:10:01]modem: response = AT+IFC=0,0
[04 May 2007 20:10:01]modem: response = OK
[04 May 2007 20:10:01]modem: command = AT+CICB=0
[04 May 2007 20:10:02]modem: response = AT+CICB=0
[04 May 2007 20:10:02]modem: response = OK
[04 May 2007 20:10:02]modem: command = AT+CPIN?
[04 May 2007 20:10:02]modem: response = AT+CPIN?
[04 May 2007 20:10:02]modem: response = +CPIN:SIM PIN
[04 May 2007 20:10:03]modem: c*****
[04 May 2007 20:10:03]modem: r*****
```

The following table describes the main messages concerning the status of the PPP connection:

Message	Meaning
Modem absent	The module cannot detect the external modem.
Modem connection configured	The modem is well configured.
Direct cable connection configured	The direct cable connection is well configured.
Connection Server: IP Remote Address @IP	Server type connection followed by the remote IP address.

---

<b>Message</b>	<b>Meaning</b>
Connection Server: IP Local Address @IP	Server type connection followed by the local IP address.
Connection Server: Established	The server connection is ready.
Connection Server: Disconnect	The server is disconnected.
Dial xxxxxx	The modem dials the xxxxxx phone number.
Connection Client: Error	Client connection is inoperative.
Connection Client: IP Remote Address @IP	Client type connection followed by the remote IP address.
Connection Client: IP Local Address @IP	Client type connection followed by the local IP address.
Connection Client: Established	The client connection is ready.
Connection Client: Disconnected	The client is disconnected.
PPP Link Down	The PPP link is down, PPP layer stopped.

## PPP/Modem VPN Page

### Introduction

This page provides information about the **PPP/Modem VPN** Page.

### Accessing the PPP/Modem VPN Page

Step	Action
1	Click the <b>Diagnostics</b> link on the <b>Home</b> page.
2	Click the <b>PPP/Modem</b> link on the <b>Diagnostics</b> page.
3	Click the <b>VPN</b> link to display the current VPN configuration.

### View of the VPN Page

The **PPP/Modem VPN** page looks like this:

VPN Connections						
	Remote address	Mode	ETG client encryption	State	Frames received	Frames sent
1						
2						
3						
4						
5						
6						
7						
8						

The following table describes the standard elements of the **PPP/Modem VPN** page:

Text	Description
<b>VPN Connections</b>	
<b>Remote address</b>	Indicates the remote IP address of the remote host (module, PC or remote network).
<b>Mode</b>	Indicates if transport or tunnel mode is used.
<b>ETG client encryption</b>	Indicates the encryption protocol used for the connection: <ul style="list-style-type: none"> <li>● <b>None:</b> AH</li> <li>● <b>Medium:</b> ESP (DES)</li> <li>● <b>High:</b> ESP (3DES)</li> </ul>
<b>State</b>	Indicate the state of the tunnel. The state of the VPN tunnel can be: <ul style="list-style-type: none"> <li>● <b>Close</b></li> <li>● <b>Open</b></li> </ul>



<b>Text</b>	<b>Description</b>
<b>Frame received</b>	Number of frame received.
<b>Frame sent</b>	Number of frame sent.

## PPP/Modem Stand-by Page

### Introduction

This page provides statistics about the **PPP/Modem Stand-by** page.

### Accessing the PPP/Modem Stand-by Page

Step	Action
1	Click the <b>Diagnostics</b> link on the <b>Home</b> page.
2	Click the <b>PPP/Modem</b> link on the <b>Diagnostics</b> page.
3	Click the <b>Stand-by</b> link to display the current configuration of the remote connection availability service.

### View of the PPP/Modem Stand-by Page

The **PPP/Modem Stand-by** page looks like this:

Normal / Backup status		Active connection	
Status	<input type="text" value="RUNNING"/>	Link	<input type="text" value="INTERNAL MODEM"/>
Switch statistics			
Switch	<input type="text" value="1"/>		
Date time last switch	<input type="text" value="07 Jan 2000 23:24:32"/>		
Secondary link statistics			
Checking OK secondary link	<input type="text" value="0"/>		
Checking error secondary link	<input type="text" value="0"/>		
Date time last checking error	<input type="text"/>		
<input type="button" value="Reset counters"/>			

The following table describes the standard elements of the **PPP/Modem Stand-by Diagnostics** page:

Text	Description
<b>Normal/Backup status</b>	
<b>Status</b>	Indicates the connection state: <b>RUNNING</b> or <b>DISABLE</b>
<b>Active connection</b>	
<b>Link</b>	Indicates which communication link is currently used: <b>ETHERNET</b> or <b>MODEM</b> .

---

<b>Text</b>	<b>Description</b>
<b>Switch statistics</b>	
<b>Switch</b>	Indicates the number of time the communication is switched between primary and secondary link.
<b>Date time last switch</b>	Indicates when the last communication switch occurred.
<b>Secondary link statistics</b>	
<b>Checking OK secondary link</b>	Indicates the number of time the secondary link is checked and available.
<b>Checking error secondary link</b>	Indicates the number of time the secondary link is checked and not available.
<b>Date time last checking error</b>	Indicates the time when the last secondary link unavailability occurred.

## FDR (Device Replacement) Statistics Page

### Introduction

Use this page to perform diagnostics on the FDR function.

### Accessing the FDR Statistics Page

Step	Action
1	Click the <b>Diagnostics</b> link on the <b>Home</b> page.
2	Click the <b>FDR statistics</b> link on the <b>Diagnostics</b> page.

### View of the FDR Statistics Page

The screenshot displays the FDR Statistics page with the following sections and data:

- Global diagnostics:** Device name (input field), Status (UNCONFIGURED), DHCP attempt (0), Device parameter saved (input field).
- Backup diagnostics:** Total (0), Automatic (0), User (0).
- Delete diagnostics:** Total (0).
- FTP diagnostics:** FTP connections error (0), FTP backup restore error (0).
- Restore diagnostics:** Total (0), Automatic (0), User (0).

A "Reset counters" button is located at the bottom center of the page.

The following table describes the elements of the FDR statistics page:

Text	Description
<b>Global diagnostics</b>	
<b>Device name</b>	Name associated to the TSX ETG 30**.
<b>Status</b>	Indicate the status of the FDR function: <ul style="list-style-type: none"> <li>Starting, Running, Stopped, Error.</li> </ul>
<b>DHCP attempt</b>	Total number of DHCP tries.
<b>Device parameters saved</b>	Parameters saved on server: <ul style="list-style-type: none"> <li>Yes, No.</li> </ul> <p>This information is only relevant if the FDR function is enabled.</p>

Text	Description
<b>Backup diagnostics</b>	
<b>Total</b>	Total number of successful backups of the TSX ETG 30** configuration to the server.
<b>Automatic</b>	Total number of successful automatic backups of the TSX ETG 30** configuration to the server.
<b>User</b>	Total number of successful backups of the TSX ETG 30** configuration to the server, initiated by the user by means of the Force Backup button on the FDR Client page ( <i>see page 389</i> ).
<b>Delete diagnostics</b>	
<b>Total</b>	Total number of backup configuration deleted on the server.
<b>FTP</b>	
<b>FTPConnections Errors</b>	Number of failed FTP connections in FDR mode. This counter shows FTP detected errors other than configuration file write or read detected errors.
<b>FTP Backup/Restore Errors</b>	Number of inoperative configuration files backup or restore attempts by FTP in FDR mode.
<b>Restore diagnostics</b>	
<b>Total</b>	Total number of successful restoration of the TSX ETG 30** configuration.
<b>Automatic</b>	Total number of successful automatic restoration of the TSX ETG 30** configuration.
<b>User</b>	Total number of successful restoration of the TSX ETG 30** configuration, initiated by the user by means of the Force Restore button on the <b>FDR Client</b> ( <i>see page 389</i> ) page.
<b>Reset Counters</b>	Click the <b>Reset counters</b> button to set all the counters to 0.

## NTP Statistics Page

### Introduction

Use this page to view NTP statistics.

### Accessing the NTP Statistics Page

Step	Action
1	Click the <b>Diagnostics</b> link on the Home page.
2	Click the <b>NTP</b> link on the diagnostics page.

### View of the NTP Statistics Page

NTP Status	
Status	IDLE

NTP Server Status	
Linked to NTP server	<input type="checkbox"/>
Server	

NTP Request Statistic			
NTP requests	0	NTP errors	0
NTP responses	0	Last error	

NTP Date and Time			
Date	Unknown	Time	Unknown
DST status	Unknown		
Time zone	(GMT+01:00)Romance Standard Time[Amsterdam CopenHagen Madrid Paris Vilnius]		

The following table describes the elements of the NTP statistics page:

Text	Description
NTP Status	Indicates the status of the NTP service: <ul style="list-style-type: none"> <li>● unconfigured,</li> <li>● configured.</li> </ul>
NTP server status	<ul style="list-style-type: none"> <li>● Indicates the status of the NTP server:               <ul style="list-style-type: none"> <li>● OK if the link is available,</li> <li>● NOK if the link is not available.</li> </ul> </li> <li>● Indicates the NTP servers available               <ul style="list-style-type: none"> <li>● NONE if there is no server,</li> <li>● PRIMARY give the name of the primary NTP server,</li> <li>● SECONDARY give the name of the secondary NTP server.</li> </ul> </li> </ul>
NTP request statistics	<ul style="list-style-type: none"> <li>● Number of requests to the NTP server.</li> <li>● Number of detected errors.</li> <li>● Number of responses from the NTP server.</li> <li>● Last detected error.</li> </ul>
NTP date and time	<ul style="list-style-type: none"> <li>● DST status: daylight saving time can be configured automatically or not</li> <li>● Date: format Day Month Year.</li> <li>● Time: format Hour Minute Second.</li> </ul>
Time zone	Name of the zone: GMT +/- X.
Reset Counters	Set all the counters to 0.

## **MIB Upload Page**

### **Presentation**

Use this page to upload the MIB of the TSX ETG 30•• to a PC connected to the module.



## Device Explorer Page

### Introduction

This page enables you to access statistics on the devices connected to the module.  
 The content of these pages are described in more detail in the **Device Explorer** section.

### Accessing the Device Explorer Page

Step	Action
1	Click the <b>Diagnostics</b> link on the <b>Home</b> page.
2	Click the <b>Device Explorer</b> link on the <b>Diagnostics</b> page.

### View of the Device Explorer Page

The screenshot displays the Schneider Electric FactoryCast™ TSX ETG 3021 web interface. At the top, there is a green header with the Schneider Electric logo and the product name. Below the header is a navigation bar with tabs for Home, Documentation, URL, Monitoring, Control, Diagnostics, Maintenance, and Setup. The Diagnostics tab is active, and the 'DEVICE EXPLORER' section is selected. On the left, a vertical menu lists various diagnostic categories: Gateway Statistics, Ethernet Statistics (with sub-items Global and Port), Modbus Statistics (with sub-items Serial Line and TCP/IP Messaging), PPP/Modem (with sub-items Statistics, Log File, VPN, and Stand-by), FDR Statistics, NTP, MIB Upload, and Device Explorer. The main content area shows a photograph of the TSX ETG 3021 device. At the bottom of the interface, a blue status bar displays 'Animation Started'.

© 2000-2009 Schneider Electric. All Rights Reserved.

## 6.4 Control Pages

---

### Overview

This section describes the control pages of the TSX ETG 30•• module.

### What's in this Section?

This section contains the following topics:

Topic	Page
<b>Control</b> Home Page	387
<b>Reboot</b> Page	388
FDR Command Page	389
Date and Time Page	390
<b>Port 502 Forwarding</b> Control Page	391

## Control Home Page

### Introduction

This page lists the control services supported by the web server of the TSX ETG 30\*\* module and provides links for accessing the services you require.

### Accessing the Control Page

Click **Control**. The various services are displayed on the left menu of the page.

### View of the Control Page



To access the service you require, click one of the links.

- **Reboot** to access the page for initializing TSX ETG 30\*\* module.
- **FDR command** to retrieve IP, Modbus, SNMP and email configurations.
- **Local date/time** to manually set the date/time of the module.
- **Port 502 Forwarding** to manually set the Port 502 Forwarding function.

## Reboot Page

### Introduction

Use this page to initialize the TSX ETG 30\*\* module. The module must be initialized (rebooted) in order for new configuration parameters to take effect. All connections are interrupted when the module is rebooted.

**NOTE:**

- Write access is controlled by a password (**Security**) whose default value is **USER**.
- All local registers are reset to initial values on module reboot.

### Accessing the Reboot Page

Step	Action
1	Click the <b>Control</b> link on the <b>Home</b> page.
2	Click the <b>Reboot</b> link on the <b>Control</b> page.

### View of the Reboot Page



## FDR Command Page

### Introduction

Use this page to configure the TSX ETG 30•• module as a FDR client.

**NOTE:** Write access is controlled by a password (**Security**) whose default value is **USER**.

The contents of this page are described in more detail in *Configuring the PSTN, GSM, or External Modem, page 157*.

### Accessing the FDR Command Page

Step	Action
1	Click the <b>Control</b> link on the Home page.
2	Click the <b>FDR Command</b> link on the control page.

### View of the FDR Command Page

Force Restore (server to module)	<input type="button" value="Restore"/>
Force Backup (module to server)	<input type="button" value="Backup"/>
Delete file	<input type="button" value="Delete"/>

To access the service you require, click one of the links.

- Restore: forces the TSX ETG 30•• to retrieve the configuration from the server.
- Backup: forces the TSX ETG 30•• to save its configuration to the server.
- Delete: delete a configuration file on the server.

## Date and Time Page

### Introduction

Use this page to manually set the date and time of the TSX ETG 30•• module.

**NOTE:** Write access is controlled by a password (**Security**) whose default value is **USER**.

### Accessing the Local Date/Time Page

Step	Action
1	Click the <b>Control</b> link on the Home page.
2	Click the <b>Local date/time</b> link on the control page.

### View of the Local Date/Time Applet

**Date**

**Time**

:  :

## Port 502 Forwarding Control Page

### Introduction

Use this page to manually set the **Port 502 Forwarding** function of the TSX ETG 30\*\* module.

**NOTE:** Write access is controlled by a password (**Security**) whose default value is **USER**.

The content of this page are described in more details in Port 502 Forwarding Configuration (*see page 197*).

### Accessing the Port 502 Forwarding Page

Step	Action
1	Click the <b>Control</b> link on the <b>Home</b> page.
2	Click the <b>Port 502 Forwarding</b> link on the <b>Control</b> page.

### View of the Port 502 Forwarding Applet

**Remote device**

IP address  .  .  .  : Port 502 (TCP)

### Activating the Port 502 Forwarding Function

Step	Action
1	Enter the IP address of the remote device you want to access in the <b>IP address</b> field.
2	Click the <b>Apply</b> button on the <b>Port 502 Forwarding</b> Control page.

**NOTE:** To de-activate the Port 502 Forwarding function, empty the **IP address** field from the **Port 502 Forwarding** Control page.

## 6.5 Monitoring Pages

---

### Overview

This section describes the monitoring pages of the TSX ETG 30\*\* module.

### What's in this Section?

This section contains the following topics:

Topic	Page
<b>Monitoring</b> Home Page	393
Data Editor Page	394
Data Editor Lite Page	395
Graphic Editor page	396
Graphic Viewer Page	397
Custom Pages	398



## Monitoring Home Page

### Introduction

This page lists the various monitoring services supported by the Web server of the TSX ETG 30 module and provides links for accessing the services you require.

### Accessing the Monitoring Page

Click **Monitoring**. The vertical menu on the left of the page displays links to the monitoring services.

### View of the Monitoring Page



To access the service you require, click one of the following links:

- **Data Editor** to create and display data monitoring tables. It provides read/write access to application or Modbus equipment variables.
- **Data Editor Lite** to access data tables created by the user with the data editor.
- **Graphic Editor** to edit graphical screens for monitoring. It provides read/write access to application or Modbus equipment variables.
- **Graphic Viewer** to display graphical screens for monitoring. It provides read/write access to application or Modbus equipment variables.
- **Custom Pages with password** to view user defined pages (accessible with password).
- **Custom Pages without password** to view user defined pages (accessible to all).

## Data Editor Page

### Introduction

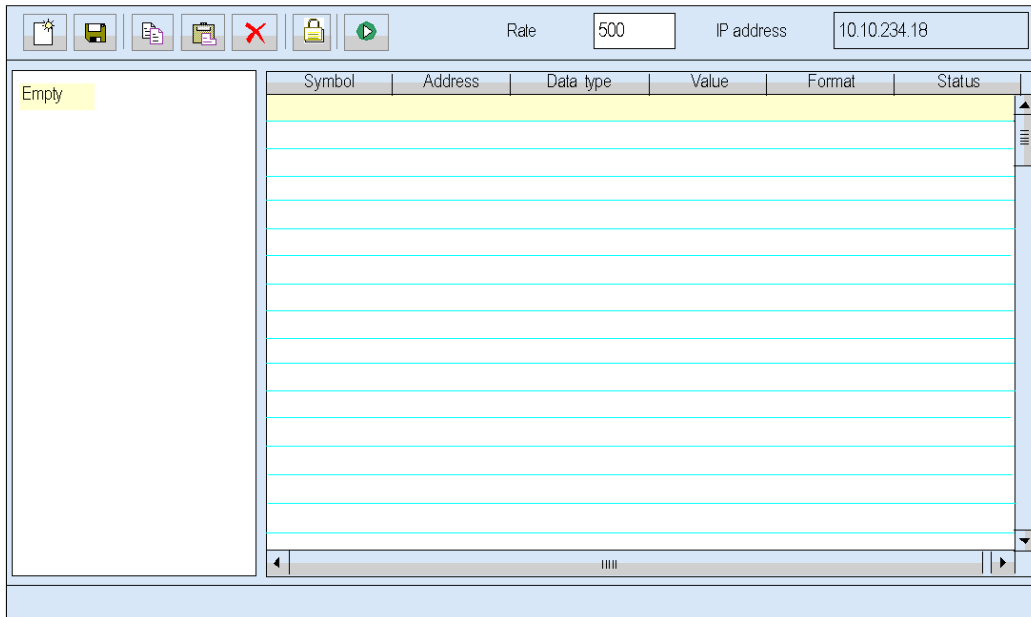
Use this page to display data tables created by Web Designer software or to edit / create data tables. Data tables give read/write access to application data and devices registers. Write access is password protected.

The contents of this page are described in more detail in the Data Editor section (*see page 412*).

### Accessing the Data Editor Page

Step	Action
1	Click the <b>Monitoring</b> link on the Home page.
2	Click the <b>Data Editor</b> link on the monitoring page.

### View of the Data Editor Page



## Data Editor Lite Page

### Introduction

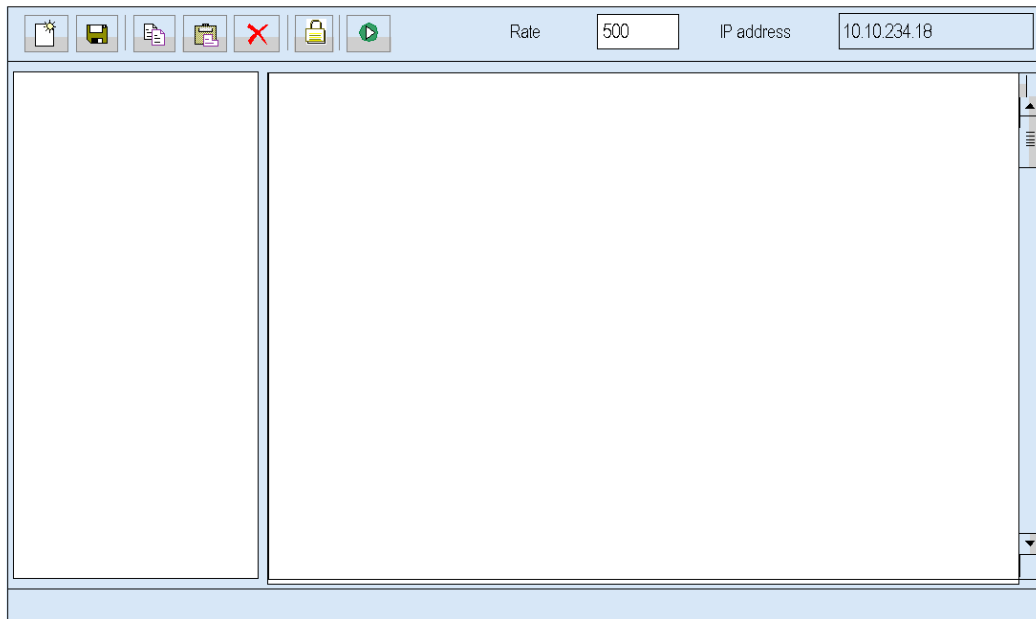
Data Editor Lite is a compact version of Data Editor that is faster to download, especially for use over a dialup connection. It provides the same interface than the standard Data Editor but with some restrictions on accessible data type.

The contents of this page are described in more detail in the Data Editor Lite section (*see page 412*).

### Accessing the Data Editor Lite Page

Step	Action
1	Click the <b>Monitoring</b> link on the Home page.
2	Click the <b>Data Editor Lite</b> link on the monitoring page.

### View of the Data Editor Lite Page



## Graphic Editor page

### Introduction

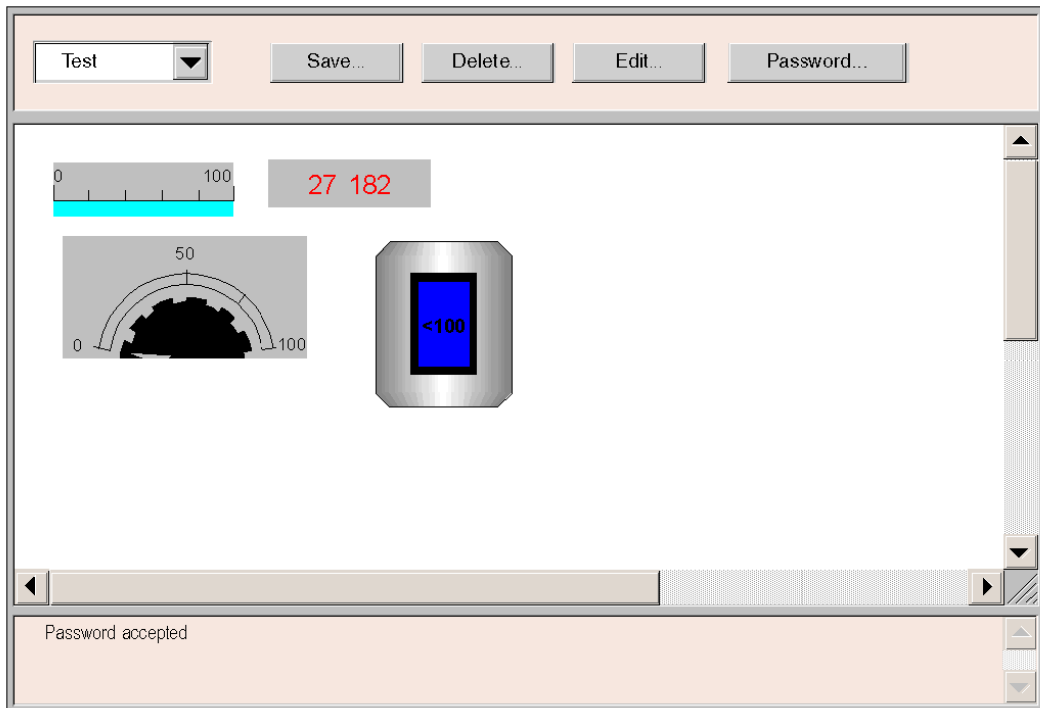
Use this page to display graphical screens created with Web Designer software or to edit/create graphical screens. Graphical screens give read/write access to application data and devices registers. Write access is password protected.

The contents of this page are described in more detail in Overview of the Graphic Editor (*see page 423*).

### Accessing the Graphic Editor Page

Step	Action
1	Click the <b>Monitoring</b> link on the Home page.
2	Click the <b>Graphic Editor</b> link on the monitoring page.

### View of the Graphic Editor Page



## Graphic Viewer Page

### Introduction

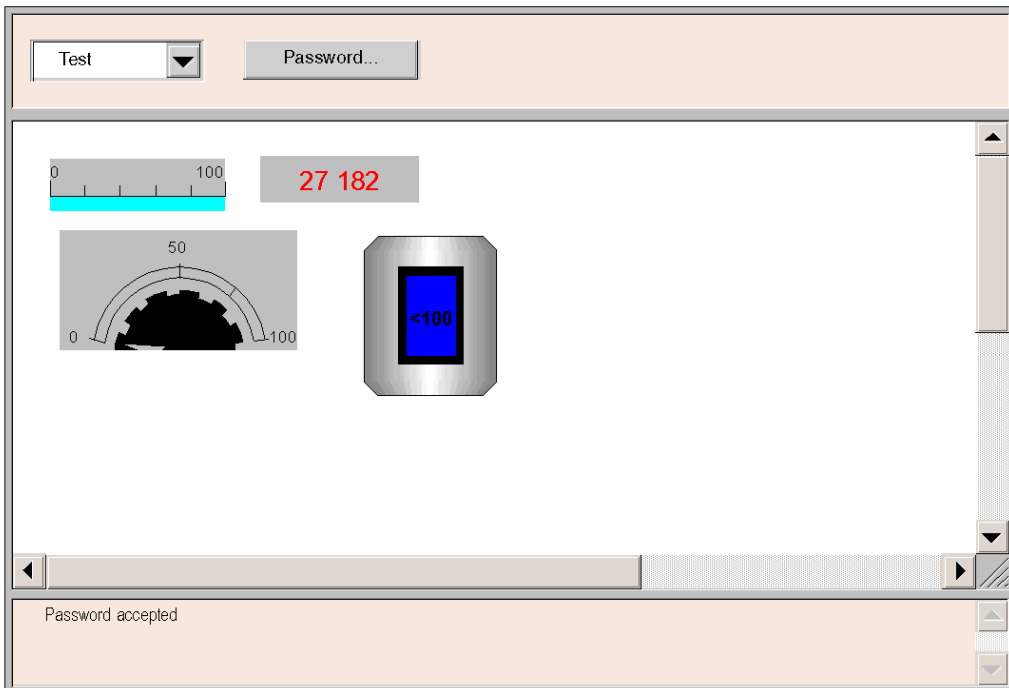
Use this page to display graphical screens created with Web Designer software. Graphical screens give read/write access to application data and devices registers. Write access is password protected.

The contents of this page are described in more detail in Overview of the Graphic Viewer (*see page 469*).

### Accessing the Graphic Viewer Page

Step	Action
1	Click the <b>Monitoring</b> link on the Home page.
2	Click the <b>Graphic Viewer</b> link on the monitoring page.

### View of the Graphic Viewer Page



## Custom Pages

### Introduction

This page gives access to pages created by the user with FrontPage or similar programs. They allow you to view your data in a custom format which may be more suited to your application.

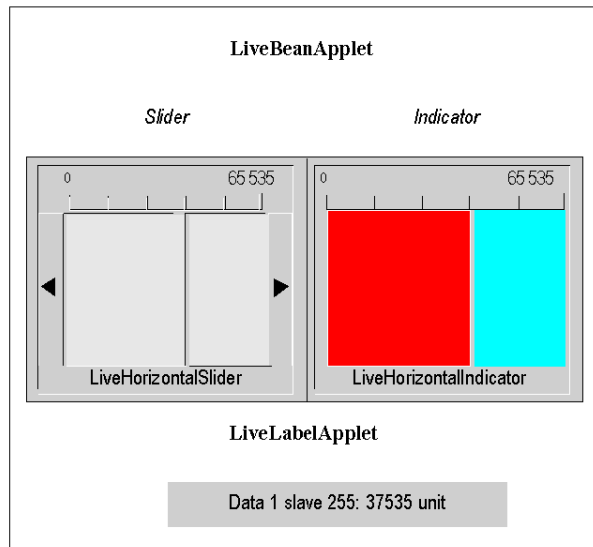
**NOTE:** These pages can be accessible to all users (pages without password) or restricted (with password).

The Creating Custom Pages (*see page 471*) chapter provides information about creating and transferring pages.

### Accessing the Custom Pages

Step	Action
1	Click the <b>Monitoring</b> link on the Home page.
2	<ul style="list-style-type: none"> <li>Click the <b>Custom Pages with Password</b> link on the monitoring page to create password protected pages.</li> <li>Click the <b>Custom Pages without Password</b> link on the monitoring page to create pages accessible to all users.</li> </ul>

### View of a Custom Page



---

# Data and Graphic Editors



---

## Overview

This chapter describes the Data and Graphic Editors Java applets that enable you to create either dynamic data tables or dynamic graphic displays. Both editors are dynamically updated with run-time data from the PLC.

## What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
7.1	Data Editor	400
7.2	Data Editor Lite	412
7.3	Pocket Data Editor	414
7.4	Graphic Editor	422
7.5	Graphic Viewer	469

## 7.1 Data Editor

### Overview

The Data Editor allows you to edit/create data monitoring tables or to display data tables. Data tables provide read/write access to application data and device registers. Write access is password protected.

Allowing write access can change system behavior.

### WARNING

#### UNINTENDED SYSTEM OPERATION

- Protect the access to writable variables by configuring passwords.
- Grant write access only to trained personnel.
- Do not grant write access to critical control variables.

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

This section shows how to use the Data Editor to display and modify the values of the symbol variables and direct addresses.

### What's in this Section?

This section contains the following topics:

Topic	Page
Data Editor	401
Creating a Data Template	403
Data Editor Spreadsheet	404
Inserting Symbol Variables in a Data Table	406
Inserting Direct Addresses in a table	407
Modifying Data Values in a Data Template	408
Saving a Data Template	410
Using an Existing Data Template	411

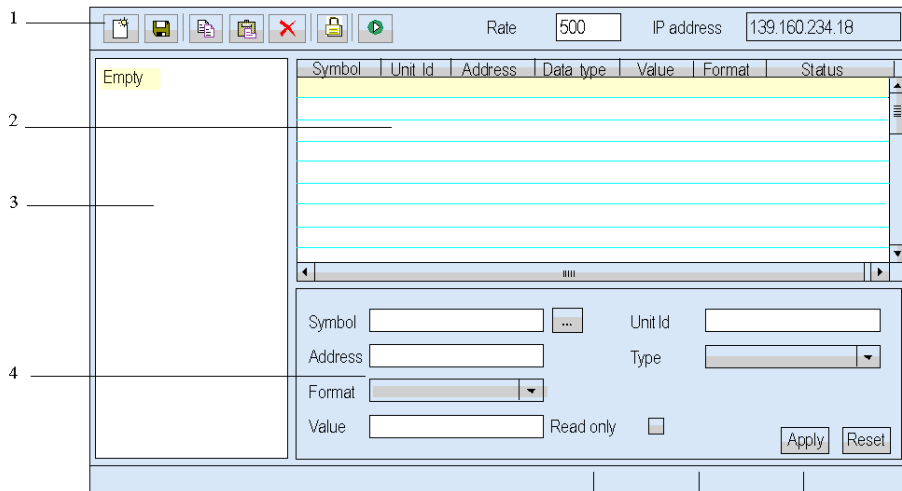


## Data Editor

### Presentation

This section describes the functions and the characteristics of the Data Editor.

### Elements of Data Editor



Number	Description
1	Toolbar (see page 402).
2	Data Editor Template (see page 404). This is a spreadsheet that contains your data.
3	List of all data templates.
4	The Configuration Area makes it possible to: <ul style="list-style-type: none"> <li>● select (see page 406) and/or modify (see page 408) a symbol,</li> <li>● modify (see page 408) a symbol value,</li> <li>● select (see page 407) and/or modify an Unit Id,</li> <li>● select (see page 407) and/or modify an address,</li> <li>● select the variable type,</li> <li>● select the variable's display format,</li> <li>● check the read-only option.</li> </ul>

## Tool bar



From left to right, the buttons indicate:


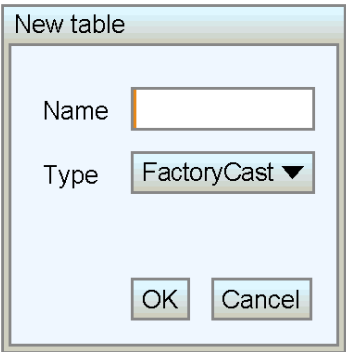
- create a new object,
- save the current application,
- copy the object,
- paste the object,
- delete the object,
- change the password,
- start or stop the animation,
- set the rate for polling the variable,
- display the server's IP address].

## Creating a Data Template

### Presentation

To display some symbols (variables) or direct addresses, you must create a new data template.

### Creating a Data Template

Step	Description
1	<p>If you would like a new one, select the  button on the tool bar. Result: The <b>New table</b> box appears.</p> 
2	Enter the name of the new Data template.
3	Click <b>OK</b> .

**NOTE:** Save (*see page 410*) the current spreadsheet before selecting a new spreadsheet (Selecting a new spreadsheet deletes the current spreadsheet).

## Data Editor Spreadsheet

### Presentation

The Data Editor spreadsheet displays data with the following fields:

- **Symbol,**
- **Unit Id,**
- **Address,**
- **Data Type,**
- **Value,**
- **Format,**
- **Status.**

This topic provides a snapshot of the spreadsheet and an explanation of each field.

### Spreadsheet

The screenshot shows a software window titled 'Data Editor Spreadsheet'. At the top, there is a toolbar with icons for file operations and a status bar showing 'Rate' set to 500 and 'IP address' set to 139.160.234.18. The main area is a spreadsheet with columns: Symbol, Unit Id, Address, Data type, Value, Format, and Status. The first row is highlighted in yellow and contains the word 'Empty'. Below the spreadsheet is a form with input fields for Symbol, Unit Id, Address, Type, Format, and Value, along with a 'Read only' checkbox and 'Apply' and 'Reset' buttons.

### Symbol Field

The **Symbol** column contains the name of variables created using Web Designer for TSX ETG 30\*\*.

The symbols are grouped in a file called Namespace.

#### NOTE:

- Variables must be declared in Web Designer for TSX ETG 30\*\* in order to be used in the Data Editor.
- You cannot change the parameters of symbols used in Data Editor via the website.

**Unit Id Field**

The **Unit Id** column contains the Modbus address of the device: 1 to 255.

**Address Field**

The **Address** column contains the symbol's address.

**Data Type Field**

The **Data Type** field contains the data type of the symbol (variable) or direct Modbus address. Direct Modbus address data types must be set by the user from a drop-down list.

The following data types are valid:

- DECIMAL
- HEXA
- BINARY
- ASCII

**Value Field**

The **Value** column is filled with the value of the symbol (variable) or direct Modbus address. This field is updated continuously.

**Format Field**

The **Format** field contains the format type for displaying the value of the symbol (variable) or direct address. The following formats are available:

- register
- coil
- discrete input
- input register
- int32
- int32swap

**Status Field**

The **Status** column contains messages about the status of communications with the symbol (variable) or direct Modbus address. If communications are normal, the status message will be "OK."


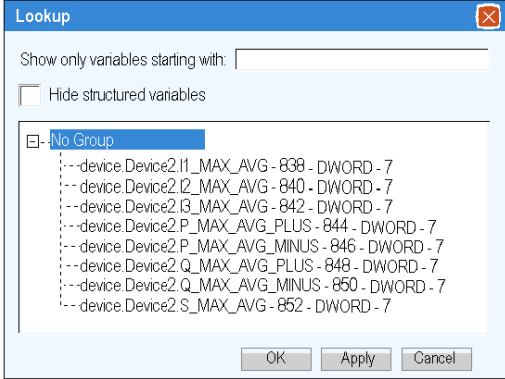
Otherwise, the Status column will contain a message describing the anomaly.

## Inserting Symbol Variables in a Data Table

### Presentation

If you want to display or modify the value of a symbol variable in the namespace, you must insert the variable in a Data Table.

### Inserting a Symbol

Step	Action
1	<p>Double-click on an empty line in the Data Table.  <b>Result:</b> The configuration zone of the Data Editor appears.</p>
2	<p>In the configuration zone, click  .</p> <p><b>Note:</b> Variables must be declared in Web Designer for TSX ETG 30** in order to be used in the Data Editor.  <b>Result:</b> The Lookup Variable dialog box appears.</p> 
3	<p>Select the symbol variables you want to insert in the Data Table.</p>
4	<p>Click <b>OK</b>.  <b>Result:</b> The symbol variables you have selected appear in the table.</p>

## Inserting Direct Addresses in a table

### Presentation

If you want to display or modify the value of a direct Modbus address, you must insert the address in a Data Table.

### Inserting a Direct Modbus Address

Step	Action
1	Double-click an empty line in the Data Table. <b>Result:</b> The configuration zone of the Data Editor appears.
2	Enter the device's Modbus address in the <b>Unit Id</b> field.
3	Enter the direct Modbus address of the variable in the <b>Address</b> field.
4	Select the data type of the direct Modbus address in the <b>Type</b> field.
5	In the configuration zone, click <b>Apply</b> . <b>Result:</b> A new line corresponding to the direct address is displayed in the Data Table.

## Modifying Data Values in a Data Template

### Presentation

You can use the Data Editor to modify the value of a symbol (variable) and of a direct Modbus address, and send the new value to the controller.

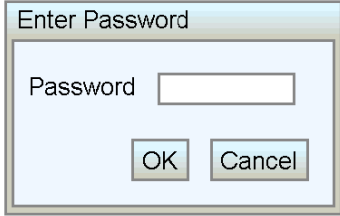
Allowing write access can change system behavior.

<b>⚠ WARNING</b>
<p><b>UNINTENDED EQUIPMENT OPERATION</b></p> <ul style="list-style-type: none"> <li>• Restrict access to the embedded server by configuring passwords.</li> <li>• Carefully select the symbols and direct addresses you authorize to be modified online.</li> <li>• Do not authorize online modification of variables of critical nature concerning human and material integrity.</li> </ul> <p><b>Failure to follow these instructions can result in death, serious injury, or equipment damage.</b></p>

### Restrictions on Modifying Data

You can only modify the value of a symbol (variable) that is write-enabled in the configurator.

### Modifying Data

Step	Action
1	Double-click on a row of the spreadsheet corresponding to the symbol (variable). <b>Result:</b> The data editor's configuration area appears.
2	In the <b>Value</b> field of the configuration area, change the value of the symbol (variable) or direct Modbus address.
3	Click <b>Apply</b> .
4	<p><b>Result:</b> A <b>Password</b> dialog appears.</p> 



<b>Step</b>	<b>Action</b>
5	Enter the password (USER by default).
6	Click <b>OK</b> . <b>Result:</b> The new value is sent to the controller.

## Saving a Data Template


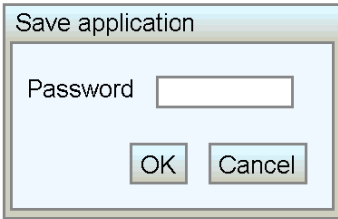
### Presentation

If you save a data template, you can use it again to view or modify the same symbols (variables) or direct addresses.

#### NOTE:

- Be careful when you modify and save a data table, the last modifications will be saved and will overwrite the existing table even if someone else has created it.
- If someone is viewing a data table which is overwritten, the modifications will be seen only on the next access to the data editor.

### Saving a Data Template

Step	Action
1	<p>Click the  button on the toolbar. <b>Result:</b> The <b>Save application</b> box appears.</p> 
2	Enter the password (USER by default).
3	Click <b>OK</b> .

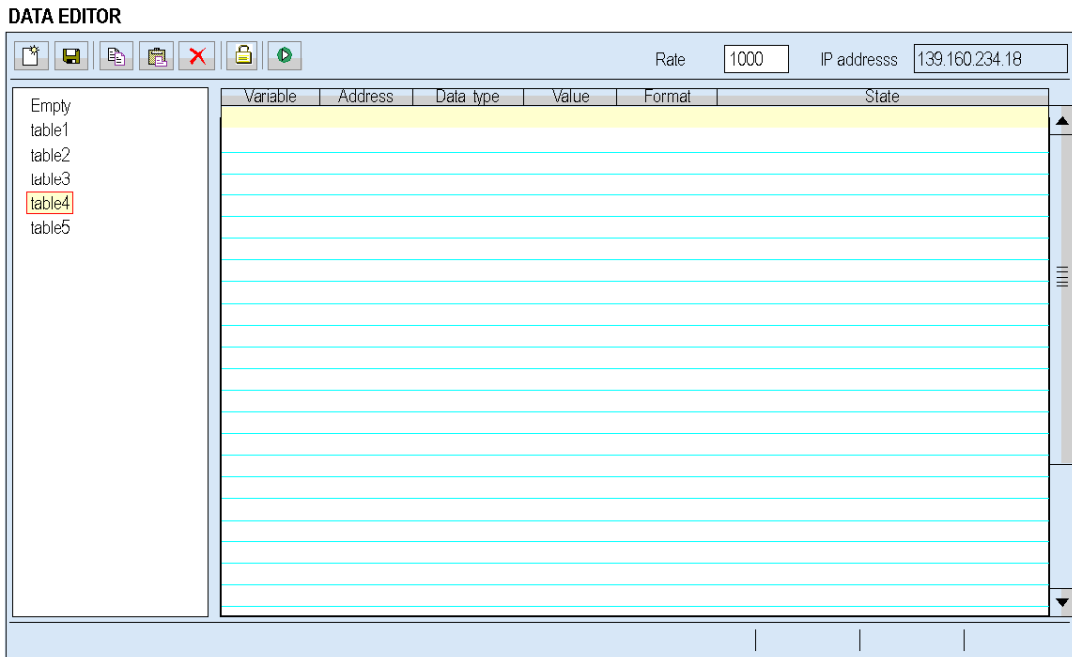
## Using an Existing Data Template

### Presentation

Once you have saved a data template, you can use it to view or modify the values of the same symbols (variables) and direct addresses.

### Finding a Data Template

A treeview menu on the Data Editor lists all the data templates which have been saved.



### Retrieving a Data Template

Select the data template you want from the treeview menu. It will appear on a spreadsheet.

### Modifying a Data Template

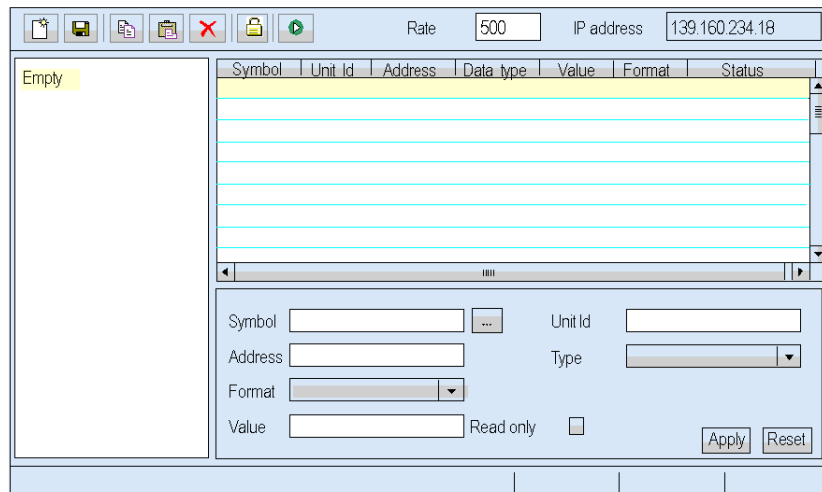
If you enter the **Write Password**, you can modify and save a new version of a data template.

## 7.2 Data Editor Lite

### Data Editor Lite

#### Presentation

Data Editor Lite is a compact version of Data Editor (*see page 400*) that is faster to download, especially for use over a modem connection. It provides the same interface than the standard Data Editor but with some restrictions on accessible data type.



#### Tables

Data Editor Lite can reuse the tables created with Data Editor. However, Data Editor tables use a wider range of variable types than in Data Editor Lite. When Data Editor Lite encounters a variable it cannot manage, it displays **Not Supported**. In this case, the variable cannot be edited.

#### Operating Mode

Refer to the Data Editor (*see page 400*) section.

**NOTE:** You cannot insert symbols (variables) in a Data Template with Data Editor Lite, you can only insert direct addresses.

## **Limitations**

The maximum number of persistent variables you can configure with the Data Editor Lite is restricted to 1000.

The maximum size of a symbol is 64 characters.

## 7.3 Pocket Data Editor

---

### Overview

This section describes the Pocket Data Editor, which runs on Pocket PC.

### What's in this Section?

This section contains the following topics:

Topic	Page
Pocket Data Editor	415
Using Pocket Data Editor	417

## Pocket Data Editor

### Presentation

Pocket Data Editor is a version of the Data Editor that operates in a Pocket PC environment. A Pocket PC with a WiFi connection can be used as a mobile client station to access data on remote equipment. Pocket Data Editor supports remote data monitoring, remote control, and remote diagnostics.

### Pocket PC Requirements

- only Pocket PCs with ARM processors
- Pocket PC Windows 2003 mobile edition

Pocket PC Windows 2002 mobile edition is not supported.

### Capabilities

Pocket Data Editor can open and edit the same data tables as the PC version in a format appropriate for Pocket PCs.

#### **NOTE:**

- You cannot save the tables edited by the Pocket Data Editor.
- If you have developed custom pages for your Web server, Internet Explorer for Pocket PC can access the same custom HTML pages on an equipment Web site. A Pocket PC screen is smaller than one on a PC, so the Web site must be adapted to return the appropriate page based on the client's browser type.

### Availability

Pocket Data Editor is available on the TSX ETG 30•• Web server.

#### **NOTE:**

- Pocket Data Editor is disabled when the Pocket PC is connected to and synchronized with a PC via USB using the ActivSync tool.
- Some communication interruptions can occur between a Pocket PC and the TSX ETG 30•• module when the WiFi (802.11) power save mode is activated on a Pocket PC. In such a case, it is not possible to connect to the module. See your Pocket PC user manual for the procedure to disable this feature.

## Variables

Pocket Data Editor accepts only symbols (variables).

Acceptable types of symbols are:

- INT,
- DINT,
- BOOL,
- EBOOL.

### NOTE:

- Direct Modbus addresses are not supported. The status bar displays **not supported** if you enter a direct Modbus address in a Pocket Data table.
- Variables that you want to monitor with the Pocket PC have to be declared to Persistent mode in Web Designer for TSX ETG 30\*\*.

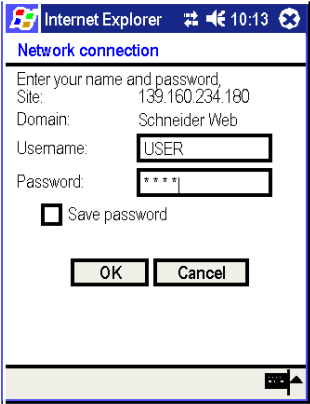


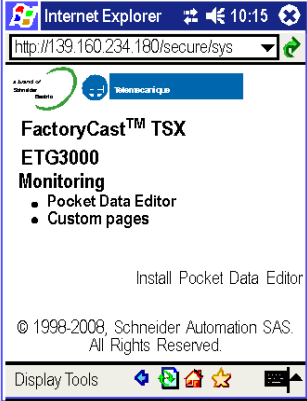
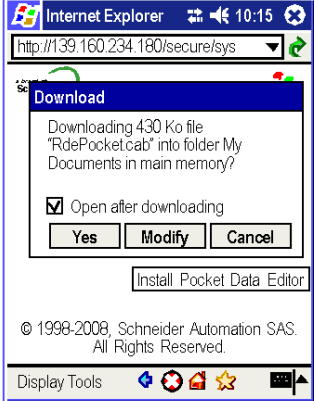
## Using Pocket Data Editor

### Presentation

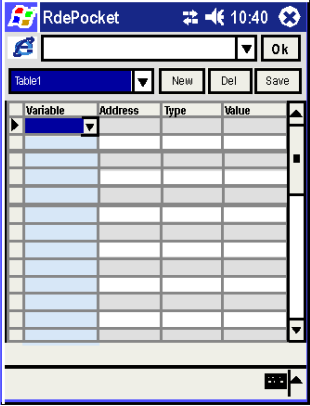
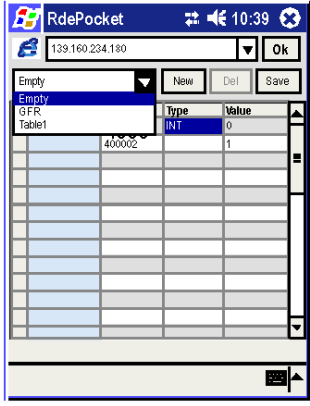
When you first connect a Pocket PC to the TSX ETG 30\*\* Web server, you must download and install the Pocket Data Editor software on your Pocket PC. After this, you can launch the Pocket Data Editor by clicking **Pocket Data Editor** on the equipment Web site home page.

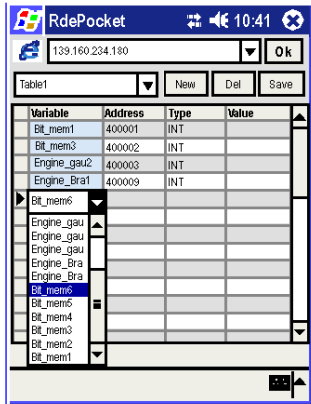
### Installing Pocket Data Editor

Step	Action
1	Turn on the Pocket PC.
2	Launch Internet Explorer.
3	<p>Enter the IP address of the remote equipment.  <b>Result:</b> The following window appears:</p> 

Step	Action
4	<p>Enter the username and password. (The defaults are USER and USER.) <b>Result:</b> the TSX ETG 30 Server home page appears:</p> 
5	<p>Click <b>Install Pocket Data Editor</b>. <b>Result:</b> The following window appears:</p> 
6	Click <b>YES</b> .

## Using Pocket Data Editor

Step	Action
1	<p>Click <b>Pocket Data Editor</b> on the equipment Web site home page.  <b>Result:</b> The following window appears:</p>  <p>Click <b>New</b> to create a new table after entering its name.  Click <b>Del</b> to delete the current table.  Click <b>Save</b> to save the new version of the table after entering the HTTP password.</p>
2	<p>Select the desired table via the drop-down menu:</p>  <p>The Pocket PC version can use the same tables defined in the Data Editor for PC and in Data Editor Lite. Not all variables are accessible. (Restrictions are detailed at the Variables discussion (<i>see page 416</i>.) A selected variable or a direct Modbus address may be reported as <b>not supported</b> in the status bar. For an unsupported variable type, the string ??? appears in the value cell.</p>

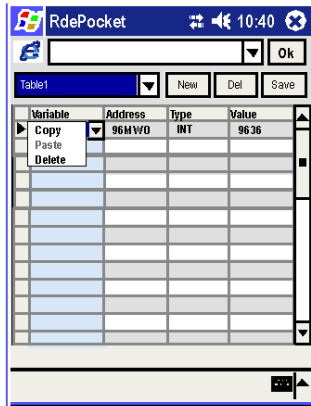
Step	Action
3	<p>If you want to view or modify the value of a TSX ETG 30•• variable, insert the symbol (variable) in the Pocket Data table.</p>  <p>You can add new symbols (variables), following the restrictions detailed in the Variables section (<i>see page 416</i>). To modify a TSX ETG 30•• variable, select the corresponding <b>value</b> cell and enter the new value.</p>

**NOTE:** Even if the **Save** button appears on the screen, you cannot save the tables edited by the Pocket Data Editor.

### Copy, Paste, or Delete Table Rows

To copy, paste, or delete, select one or more rows by clicking in the first cell (to the left of the **variable** cell). The selected rows are highlighted. Hold down the mouse button in this cell until a popup menu appears, then select the desired action.

- To copy, select one or more rows that have values (that are not empty).
- To paste the copied rows, select a row, empty or not. If several rows are copied, the other rows are pasted one after the other.
- To delete one or more rows, select rows that have values (that are not empty) and choose **delete**.

**NOTE:**

- If you want to display a custom page, click **Custom pages** on the equipment Web site home page.
- You can also launch the Pocket Data Editor from its program icon.

## 7.4 Graphic Editor

---

### Overview

This section describes the functions and characteristics of the Graphic Editor. The Graphic Editor allows you to create graphical screens for monitoring and to display/edit graphical screens created with Web Designer software. The Graphic Editor provides a predefined set of graphic objects libraries.

### What's in this Section?

This section contains the following topics:

Topic	Page
Graphic Editor Presentation	423
Graphic Editor Toolbar	425
User Functions in the Display Window	430
Properties Sheet	433
Security	435
Graphic Editor Applet Parameters	436
Graphic Objects	438
Extended Graphic Objects	459

## Graphic Editor Presentation

### Interface

The Graphic Editor applet is made up of three windows:

- **Top window:** features an area for presenting the user commands and dialog boxes for creating, saving, reading and editing a graphic display.
- **Display window:** presents the current graphic display. When you create a new graphic display, this window turns into a blank space into which you can add the graphic objects that will make up the required graphic display.
- **Message window:** contains messages generated by the Graphic Editor.

### View of the Graphic Editor

The figure below shows the Graphic Editor applet with its initial top window and empty display and message windows.




### Graphic Objects

All the graphic objects supplied with the Graphic Editor are able to communicate with the Modbus devices from which the Graphic Editor applet was downloaded. There is no additional "wiring" between the graphic objects and the "communication objects". All the graphic objects are designed as standalone objects, which means that no connection is needed between the objects, and that each object is capable of operating on its own.

## Obtaining a Graphic Display

Once the Graphic Editor applet has loaded in your Web browser, you will generally want to view a graphic display or create/modify a graphic display. Users who just wish to obtain existing graphic displays (operators, for example) and to interact with these displays can click on the **Graphic Viewer** link instead of **Graphic Editor**. A window containing the graphic elements of the interface appears. It does not have the Edit menu. The Graphic Viewer is smaller than the Graphic Editor, and thus loads more quickly.

Allowing write access can change system behavior.

 <b>WARNING</b>
<b>UNINTENDED EQUIPMENT OPERATION</b>
<ul style="list-style-type: none"><li>● Restrict access to the embedded server by configuring passwords.</li><li>● Carefully select the symbols and direct addresses you authorize to be modified online.</li><li>● Do not authorize online modification of variables of critical nature concerning human and material integrity.</li></ul>
<b>Failure to follow these instructions can result in death, serious injury, or equipment damage.</b>

**NOTE:** Write access is controlled by a password (**Security**) whose default value is **USER**.

## Creating and Modifying Graphic Displays

To create and modify graphic displays, click **Edit...** The standard functions of the Graphic Editor are displayed. With these tools, you can select objects from a palette, position them in an area, move and resize them with the mouse and define their properties. You can immediately test the graphic display modified with the execution data from Modbus devices by clicking **Done** to leave Edit mode. If you are satisfied with the graphic display you have created, you can save it in the module for later use by clicking **Save...**, provided that you have entered the correct password.

**NOTE:** Be careful when you modify and save a graphic page, the last modifications will be saved and will overwrite the existing page even if someone else has created it.

## User Functions

Most of the user functions in the Graphic Editor can be found in the top window (*see page 425*). You can modify the size and position of a graphic object directly in the display window. All the properties of a graphic object (such as its scale, labels and colors) are defined in the properties sheet (*see page 433*).



## Graphic Editor Toolbar

### Presentation

The Graphic Editor toolbar is composed of several dialog panels, only one of which is shown at a time. Switching from one dialog to another is done by clicking buttons on the current dialog. This section describes the dialog panels that comprise the top window.

### Top Dialog

The **Edit Dialog** allows you to select a graphic object for placement in the display window, and provides access to all graphic editing functions. The available graphic objects are presented in a set of palettes, with one palette visible at a time. There are two palettes.

The standard palette:



The extended palette:



The controls in the **top dialog box** provide the following functions.

- **Drop-down List.** The drop-down list box shows all the graphic pages that have been saved to the TSX ETG 30\*\* module, and are available for retrieving. When you select a graphic page from this list, the graphic display currently visible in the window is replaced with the one selected. If the current graphic page has been modified since it was last saved, you will be asked for confirmation that the changes are to be discarded. If the special entry <new> is chosen from the list, then the display window is cleared, and a new graphic page can be created.
- **Save.** The **Save** button causes the **Save dialog box** to become visible. This button is disabled until you have entered a correct write-enable password.
- **Delete.** The **Delete...** button causes the **Delete dialog box** to become visible. This button is disabled until you have entered a correct password, or if the current graphic display has not yet been saved.
- **Edit.** The **Edit...** button causes the **Edit dialog box** to become visible.
- **Password.** The **Password...** button causes the **Password dialog box** to become visible.
- **Information display area.** The information display area shows the name and version of the TSX ETG 30\*\*.

### Save Dialog

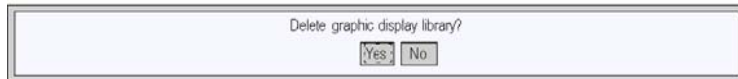
The **Save dialog box** allows you to save the current graphic display.



When the **Save dialog box** is presented, the name of the current graphic page is shown in the dialog's text field. If the current graphic page has never been saved (i.e., a "new" graphic display), then the text field is blank. Once you have either accepted the current name (a "save" operation) or provided a new name (a "save as" operation), then you can click **OK** to save the contents of the current graphic display to the Web server module. The **Cancel** button will cause the **Top dialog box** to be shown again, with no action being taken.

### Delete Dialog

The **Delete dialog box** allows you to delete the current graphic page.



If you click **Yes**, the existing graphic display window is cleared and the graphics file on the Web server module is deleted. Clicking **No** will cause the **Top dialog box** to be shown again, with no action being taken.

### Password Dialog

The **Password dialog box** allows you to enter the password that enables those user functions that modify graphic display files or device's variables data values.



If you enter the correct password and click **OK**, then you will be allowed to save and delete the current graphic display. Correct password entry also permits you to write new values to the device (via those graphic objects that support writing values to a device, if any). If you click **OK** when the text field is empty, then the current password permissions, if any, are cleared. The **Cancel** button will cause the **Top dialog box** to be shown again, with no changes made to current password permissions.

## Edit Dialog

The **Edit dialog box** allows you to create or modify a graphic page, by selecting a graphic object for placement in the display window, and accessing all the graphic editing functions. The graphic objects available are presented in a single object palette.



The controls of the **Edit dialog box** provide the following functions:

- The **Drop-down List Box** shows the set of palettes that are available. When you select the name of a palette from the list, the palette area of the dialog displays the selected one's graphic objects.
- The **Palette** shows the graphic objects that are in the current palette with an icon that depicts each graphic object's type (meter, button, etc.). When you click any of the icons in the palette, a graphic object of the corresponding type becomes selected for insertion. While the Graphic Editor is in "insert mode," if you click in an open area of the display window, an instance of the selected graphic object is inserted into the graphic display.
- The **Information Area** shows the name and size of the graphic object that is currently selected.
- The **Cut** button causes the currently selected graphic object(s) to be removed from the graphic display and saved to a buffer (i.e., an internal clipboard), replacing any existing contents of the buffer.
- The **Copy** button causes the currently selected graphic object(s) to be copied to the buffer, replacing any existing contents.
- The **Paste** button causes the content of the clipboard to be inserted into the upper left corner of the graphic display. The pasted graphic objects can then be moved to the desired location in the display.
- The **Properties** button causes the properties sheet (*see page 433*) for the currently selected graphic object to be shown.
- The **Customize** button causes the Customizer for the currently selected object to be shown, if the graphic object has been provided with one.
- The **Layout** button shows the **Layout dialog box**.
- The **Options** button shows the **Options dialog box**.
- The **Done** button causes the **Top dialog box** to be shown again.

## Layout Dialog

The **Layout dialog box** allows you to change the position and size of a group of graphic objects.



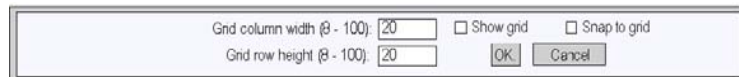
The controls of the **Layout dialog box** provide the following functions.

- The **Right**, **Bottom**, **Left**, and **Top** buttons can be used to align the edges of the selected graphic objects so that their specified sides are at the same position. At least two graphic objects must be selected for these buttons to be enabled.
- The **Horizontal**, and **Vertical** buttons are used to align the centers of the graphic objects. At least two graphic objects must be selected for these buttons to be enabled.
- The **Horizontal** and **Vertical** buttons are used to space the selected graphic objects regularly, in order that the horizontal or vertical spacing between the objects is the same. At least three graphic objects must be selected for these buttons to be enabled.
- The **Width** and **Height** buttons are used to achieve parity in dimensions of the graphic objects, so the selected width or height corresponds. At least two graphic objects must be selected for these buttons to be enabled.
- The **Done** button causes the **Edit dialog box** to be shown again.

**NOTE:** For all layout operations (except **Space evenly**) one of the selected objects is considered the "reference object" to which all other selected objects refer in order to know their new position or dimension. For example, when the "Width" button is pressed, all of the selected objects will have their width changed to match the width of the reference object. The reference object is differentiated from the other selected objects by making its selection box a different color than the others.

## Options Dialog

The **Options dialog box** is used to change the settings related to a grid drawn in the display window. The grid is solely for assistance in editing or creating a graphic display and is shown only when the Graphic Editor is in "edit mode."



The controls of the **Options dialog box** provide the following function.

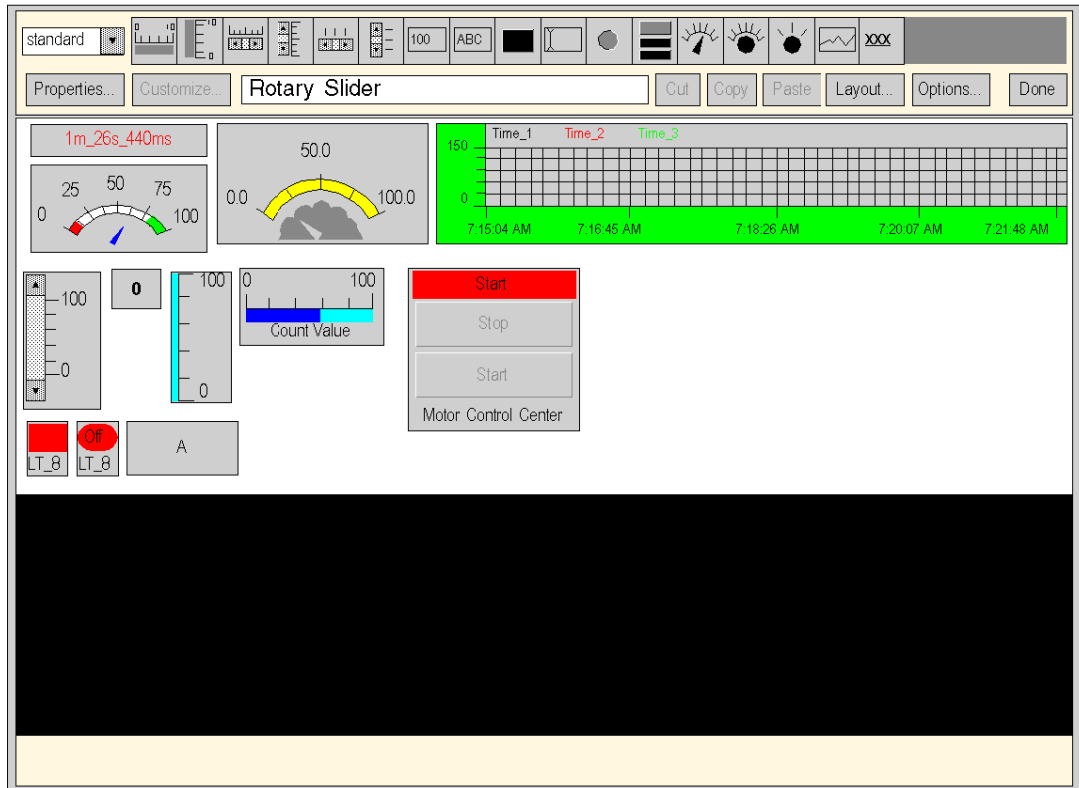
- The cell size of the grid can be changed by the entering the grid's column width and row height into the dialog's text fields.
- If the **Show grid** check-box is checked, the grid will be drawn; otherwise, no grid will be shown.
- If the **Snap to grid** check-box is checked, then, when you change the size or position of a graphic object, the changed coordinate(s) or dimension(s) is automatically adjusted so that it coincides with a grid point.
- The **OK** button causes the current option settings to become active, and the **Edit dialog box** to be shown again.
- The **Cancel** button causes the **Edit dialog box** to be shown again, with no option settings being changed.

## User Functions in the Display Window

### Presentation

The user functions available in the **Graphic Editor** display window enable objects to be selected, moved and resized. To move or resize an object or objects, start by selecting the graphic object(s) to be modified. An object is selected when it is surrounded by a selection box. Conversely, objects that are not selected do not have selection boxes.

The figure below shows the **Graphic Editor** display window.



## Selecting Graphic Objects

You can set the selection status (selected/deselected) for a graphic object using the following user actions:

- To select a graphic object, just click on it with the mouse. If other objects have already been selected, they will immediately be deselected.
- You can select several graphic objects at a time using the selection box in the display window. If you press the mouse button in a free area of the display window (not on a graphic object) and move the mouse without releasing the button, a box bordered with dotted lines will appear. One corner of the box remains fixed where you first pressed the button, while the opposite corner follows the mouse cursor position. When you release the mouse button, all the objects within the selection box are selected. Objects outside the selection box are deselected.
- You can select/deselect a graphic object without altering the selection status of other objects. To do this, hold down the CTRL key when you click on an object. This allows you to add or remove individual graphic objects in a group of selected objects.
- You can also select a graphic object without altering the selection status of other objects by holding down the SHIFT key when you click on the object. When you select an object in this way, it becomes the reference object (*see page 433*) in the group of selected objects. The main purpose of this action is to modify the reference object in a group of selected objects before using one of the **Page layout** operations.
- You can deselect all the graphic objects by clicking the mouse button in a free area of the display window, not on a graphic object.

## Sizing Graphic Objects

To change the size of a graphic object, select it and then use the mouse to change the size of the selection box around the object. When you move the mouse cursor over the object's selection box, the cursor's appearance changes according to the type of resizing you may perform. If you press the mouse button while the cursor is over the object's selection box and move the mouse without releasing the button, a box bordered with dotted lines will appear. When you release the mouse button, the object is resized to fit the dimensions of the area you have defined. You can carry out different resizing operations depending on which part of the object's selection box you move. Each corner of the box enables the adjacent sides to be moved, and each side enables that side only to be moved.

## Moving Graphic Objects

You can move a graphic object in the display window using the mouse. If you press the mouse button while the cursor is over an object and move the mouse without releasing the button, a selection box will appear. When you release the mouse button, the object is moved into the selection box.

To move several graphic objects, select them and then move the group of objects in the same way as you would a single object. When you move a group of objects, a selection box appears for each object in the group.

### Defining Graphic Object Properties

You can define the properties of a graphic object using the property sheet (*see page 433*). If this window is displayed, you can modify the properties of the selected graphic object. You can display the property sheet by clicking **Properties...** or by double-clicking the selected object in the display window.

### Customizing Complex Graphic Objects

Certain complex graphic objects have a very wide range of properties. Configuring these objects using the properties sheet can be a long-winded process. You can use a customization module to make it easier to configure complex graphic objects. The customization module is a dialog box designed specially for configuring the graphic object with which it is associated. When the Graphic Editor detects a customization module associated with the selected graphic object, the **Customize...** button is enabled so that the module can be accessed. When you double-click on a graphic object that has an associated customization module, the module is displayed instead of the properties sheet. If a graphic object has an associated customization module, only its name is displayed in the properties sheet.

### Display Background Image

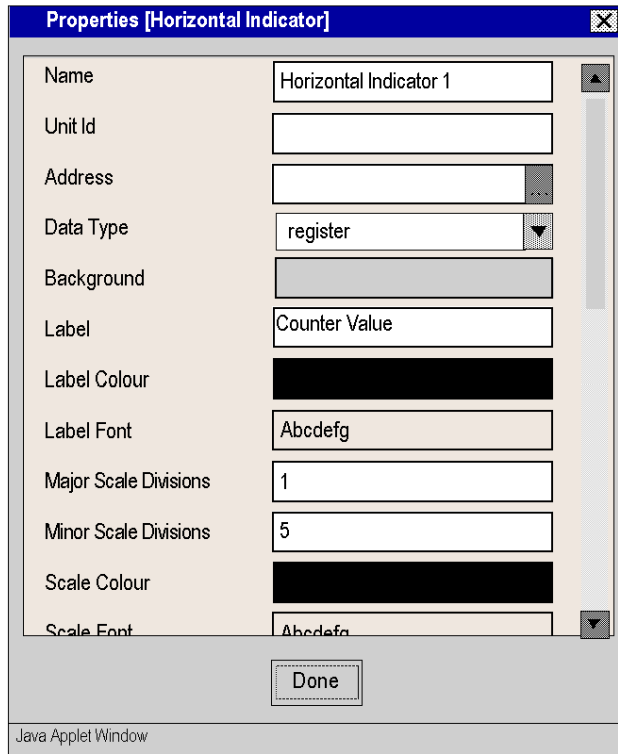
The Background image property of the Graphic Editor allows you to choose an image that will be used as the background for the display. The image may be a *.gif* file or a *.jpeg* file. Refer to the Adding Images part (*see page 461*) for information on adding images.



## Properties Sheet

### Presentation

The properties sheet is a floating (non-modal) dialog box which presents all the configurable properties of the selected graphic object:

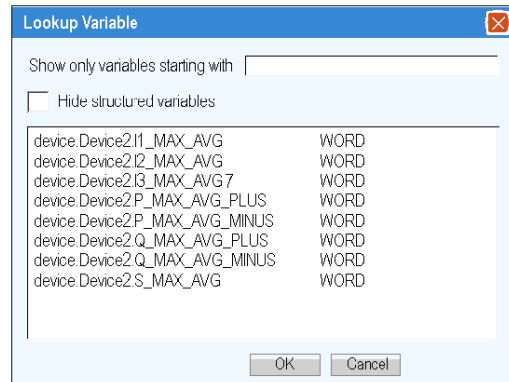


The properties of a graphic object are specific to its type. They are contained in a drop-down list, and are identified by a name and value. The Graphic Editor provides a description of the graphic objects (*see page 438*).

### Find variables dialog box

For each of the graphic objects provided with the Graphic Editor, a property editor is provided for its **Address** property. This property editor not only allows you to directly enter the address of a Modbus variable, but also provides access to the **Find variables dialog box**. The Lookup Dialog allows you to pick a symbol (variable) name from a list of symbol (variables) that have been "Web enabled" by Web Designer for TSX ETG 30••.

The Lookup Variable dialog box looks like this:



## Security

### Presentation

Your data is protected by three security devices.

- The HTML page which contains the Graphic Editor applet has been placed in a secure directory on the Web module. You need to provide the correct password which will allow him/her to download the HTML page.
- You must enter the correct password in the **Password** dialog box to be able to save/delete files or send data values. For the transfer of data values, the Graphic Editor reinforces the read only mode by deactivating all the graphic objects user commands.
- Web Designer for TSX ETG 30•• enables you to indicate if an element is read only (read only is set by default). The **Graphic Editor** reinforces the read only attribute of a symbol (variable) by rejecting any request which would define a new data value and by displaying a message window.

## Graphic Editor Applet Parameters

### Presentation

Three applet parameters allow the **Graphic Editor** behavior to be customized. These are defined by the <PARAM> tags inside the <APPLET> tag on the Graphic Editor HTML page. The parameters recognized by the **Graphic Editor** applet are as follows:

- **LOAD**: this parameter asks the **Graphic Editor** to automatically load a specific graphic file at start-up. If this file does not exist, a message is displayed. If this parameter does not appear in the <APPLET> tag, no file will be automatically loaded at start-up and you must choose an initial graphic file from the list proposed by the **Graphic Editor**.
- **MODE**: this parameter asks the **Graphic Editor** to start either in Edit (normal mode) or View mode (specific mode). When starting in View mode, the **Graphic Editor** only displays the display window. When this parameter is used with the **LOAD** parameter, you can design a Website with HTML pages dedicated to specific graphic display. The user does not need to select a graphic file so the behavior of the HMI screen is more standard. This parameter may take the following values:
  - **EDIT** (default value): The **Graphic Editor** starts up in Edit mode (normal mode).
  - **VIEW\_RO**: The **Graphic Editor** starts up in View mode (read only). The Web browser user is not authorized to send data values to Modbus devices.
  - **VIEW\_RW**: The **Graphic Editor** starts up in View mode (read/write). The Web browser user is authorized to send data values to Modbus devices after having entered the password to allow write access.
- **AUTO\_LOGIN**: this parameter asks the **Graphic Editor** to automatically indicate the password which authorizes write access to Modbus devices. If the **MODE** parameter is set to **VIEW\_RW** or **EDIT**, and if you set **AUTO\_LOGIN** to **TRUE**, the **Graphic Editor** authorizes write access to Modbus devices without asking the user to enter a password. This parameter may take the values **FALSE** (default value) and **TRUE**.

**Example**

Here is an example of an <APPLET> tag which asks the **Graphic Editor** to start in View mode and automatically load a graphic file called **UNIT\_1**. In this case, the Web browser allows you to send values to Modbus devices via any graphic object handling the sending of values (providing that the password to allow write access has been entered).

```
<APPLET codebase="/classes"
archive="SAComm.jar,GDE.jar,Widgets.jar"
code="com.schneiderautomation.gde.GdeApplet"
width="700" height="514">
<PARAM name="LOAD" value="UNIT_1">
<PARAM name="MODE" value="VIEW_RW">
<PARAM name="AUTO_LOGIN" value="FALSE">
</APPLET>
```

## Graphic Objects

### Presentation

All graphic objects offered by the **Graphic Editor** help you to create graphic displays imitating conventional instrument panels. All the data control and monitoring objects have integrated communication functions and are designed as standalone graphic objects.

Be aware, however, that if communication to the device linked to the graphic object is lost, the object becomes inoperative without the end device's knowledge.

For example, say you have programmed a pushbutton object to jog a motor when the button is depressed and to stop jogging when the button is released. If communications are lost while the button is depressed, the motor will continue to jog even when the button is released. Graphic objects should not be used to control situations such as this unless other interlock precautions are installed in the system.

### **WARNING**

#### **UNINTENDED EQUIPMENT OPERATION**

Do not use graphic objects in situations where loss of communication to the TSX ETG 30\*\* can put human integrity or material damage at risk.

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

In addition, all the objects in the **Graphic Editor** exist in the form of applets to help clients who wish to insert several simple applets into a single HTML page. When combined with the *LiveBeanApplet*, the graphic objects in the **Graphic Editor** can be used in the same way as the *LiveLabelApplet*.

### Horizontal Indicator

A horizontal indicator gives an analogue representation of the value of a variable in a device. This is a horizontal bar which represents a percentage of its range in physical units. It is possible to display the value's digital indication in the centre of the bar.

The table below describes the horizontal indicator's properties:

Property	Description	Limits
<b>Name</b>	Name of the graphic object	–
<b>Unit ID</b>	Address of the Modbus slave containing the variable to monitor	
<b>Address</b>	Direct address or symbol's name to monitor	Note 1 (see page 468)

Property	Description	Limits
<b>Data Type</b>	Data type of the direct address or data type of the symbol	Note 2 (see page 468)
<b>Background</b>	Graphic object background color	–
<b>Label</b>	Label to be displayed as part of the graphic object	
<b>Label Color</b>	Color of the label	
<b>Label Font</b>	Font used for the label	
<b>Major Scale Divisions</b>	Number of major scale divisions (marked)	0 to 100
<b>Minor Scale Divisions</b>	Number of minor scale divisions (not marked)	0 to 100
<b>Scale Color</b>	Color of the scale and its labels	–
<b>Scale Font</b>	Font used for the scale labels	
<b>Scale Precision</b>	Number of decimal places to be shown for the scale labels (set to -1 to use a general exponential format)	-1 to 6
<b>Maximum EU Value</b>	Maximum value of the variable in physical units	–
<b>Minimum EU Value</b>	Minimum value of the variable in physical units	
<b>Maximum Value</b>	Maximum gross value (without scale) of the device variable	Note 3 (see page 468)
<b>Minimum Value</b>	Minimum gross value (without scale) of the device variable	Note 3 (see page 468)

Property	Description	Limits
<b>Value Visible</b>	Indicates if there should be digital display of the value on the scale	–
<b>Value Font</b>	Font used for digital display of the value (where this exists)	
<b>Bar Background</b>	Background color of the indicator bar	
<b>Bar Color</b>	Color of the indicator bar (if the scale value is within the High/Low range)	
<b>High High Limit Value</b>	Value expressed in physical units of the "High High" limit.	
<b>High High Limit Color</b>	Color of the indicator bar if the scale value is greater than the "High High" limit	
<b>High Limit Value</b>	Value expressed in physical units of the "High" limit.	
<b>High Limit Color</b>	Color of the indicator bar if the scale value is greater than the "High" limit.	
<b>Low Limit Value</b>	Value expressed in physical units of the "Low" limit	
<b>Low Limit Color</b>	Color of the indicator bar if the scale value is less than the "Low" limit	
<b>Low Low Limit Value</b>	Value expressed in physical units of the "Low Low" limit	
<b>Low Low Limit Color</b>	Color of the indicator bar if the scale value is less than the "Low Low" limit	
<b>Limit Deadband</b>	Neutral range (as a percentage of the EU range) to apply to verification of the High/Low limit	0 to 10
<b>Border Width</b>	Width (in pixels) of the graphic object border	0...32
<b>Border Color</b>	Color of the graphic object border	–
<b>Value</b>	Simulated gross starting value (without scale) for testing the graphic object	Note 3 (see page 468)



## Vertical Indicator

A vertical indicator gives an analogue representation of the value of a variable in a device. This is a vertical bar which represents a percentage of its range in physical units.

The table below describes the vertical indicator's properties:

Property	Description	Limits
<b>Name</b>	Name of the graphic object	–
<b>Unit ID</b>	Address of the Modbus slave containing the variable to monitor	
<b>Address</b>	Direct address or symbol's name to monitor	Note 1 (see page 468)
<b>Data Type</b>	Data type of the direct address or data type of the symbol	Note 2 (see page 468)
<b>Background</b>	Graphic object background color	–
<b>Label</b>	Label to be displayed as part of the graphic object	
<b>Label Color</b>	Color of the label	
<b>Label Font</b>	Font used for the label	
<b>Major Scale Divisions</b>	Number of major scale divisions (marked)	0...100
<b>Minor Scale Divisions</b>	Number of minor scale divisions (not marked)	0...100
<b>Scale Color</b>	Color of the scale and its labels	–
<b>Scale Font</b>	Font used for the scale labels	
<b>Scale Precision</b>	Number of decimal places to be shown for the scale labels (set to -1 to use a general exponential format)	-1...6
<b>Maximum EU Value</b>	Maximum value of the variable in physical units	–
<b>Minimum EU Value</b>	Minimum value of the variable in physical units	
<b>Maximum Value</b>	Maximum gross value (without scale) of the device variable	Note 3 (see page 468)
<b>Minimum Value</b>	Gross minimum value (without scale) of the variable in the device	Note 3 (see page 468)

Property	Description	Limits
<b>Bar Background</b>	Background color of the indicator bar	–
<b>Bar Color</b>	Color of the indicator bar (if the scale value is within the High/Low range)	
<b>High High Limit Value</b>	Value expressed in physical units of the "High High" limit.	
<b>High High Limit Color</b>	Color of the indicator bar if the scale value is greater than the "High High" limit	
<b>High Limit Value</b>	Value of the "High" limit expressed in physical units	
<b>High Limit Color</b>	Color of the indicator bar if the scale value is greater than the "High" limit.	
<b>Low Limit Value</b>	Value of the "Low" limit expressed in physical units	
<b>Low Limit Color</b>	Color of the indicator bar if the scale value is less than the "Low" limit	
<b>Low Low Limit Value</b>	Value of the "Low Low" limit expressed in physical units	
<b>Low Low Limit Color</b>	Color of the indicator bar if the scale value is less than the "Low Low" limit	
<b>Limit Deadband</b>	Neutral range (as a percentage of the EU range) to apply to verification of the High/Low limit	
<b>Border Width</b>	Width (in pixels) of the graphic object border	0...32
<b>Border Color</b>	Color of the graphic object border	–
<b>Value</b>	Simulated gross starting value (without scale) for testing the graphic object	Note 3 (see page 468)

### Horizontal or Vertical Slider

A horizontal or vertical slider gives an analogue representation of the value of a variable in a device. This is a slider, whose position is indicated by the cursor, which represents a percentage of its range in physical units. Using the mouse, you can change the value of the slider by sending a new value to the device.

The table below describes the horizontal or vertical slider's properties:

Property	Description	Limits
<b>Name</b>	Name of the graphic object	–
<b>Unit ID</b>	Address of the Modbus slave containing the variable to monitor	
<b>Address</b>	Direct address or symbol's name to monitor	Note 1 (see page 468)

Property	Description	Limits
<b>Data Type</b>	Data type of the direct address or data type of the symbol	Note 2 (see page 468)
<b>Background</b>	Graphic object background color	–
<b>Label</b>	Label to be displayed as part of the graphic object	
<b>Label Color</b>	Color of the label	
<b>Label Font</b>	Font used for the label	
<b>Major Scale Divisions</b>	Number of major scale divisions (marked)	0...100
<b>Minor Scale Divisions</b>	Number of minor scale divisions (not marked)	0...100
<b>Scale Color</b>	Color of the scale and its labels	–
<b>Scale Font</b>	Font used for the scale labels	
<b>Scale Precision</b>	Number of decimal places to be shown for the scale labels (set to -1 to use a general exponential format)	-1...6
<b>Maximum EU Value</b>	Maximum value of the variable in physical units	–
<b>Minimum EU Value</b>	Minimum value of the variable in physical units	
<b>Maximum Value</b>	Maximum gross value (without scale) of the device variable	Note 3 (see page 468)
<b>Minimum Value</b>	Minimum gross value (without scale) of the device variable	Note 3 (see page 468)
<b>Block Increment</b>	Amount by which the scale value is modified when the user clicks on the bar's slide area.	–
<b>Unit Increment</b>	Amount by which the scale value is modified when the user clicks on the slider arrows	
<b>Border Width</b>	Width (in pixels) of the graphic object border	0...32
<b>Border Color</b>	Color of the graphic object border	–

### Horizontal or Vertical Selector

A horizontal or vertical selector allows you to choose from a number of options. Once the selection has been made, the value corresponding to the choice is sent to the device. The choices are represented by the marks on a "scale", the current selection being indicated by the position of the cursor on a slider.

The table below describes the horizontal or vertical selector's properties:

Property	Description	Limits
<b>Name</b>	Name of the graphic object	–
<b>Unit ID</b>	Address of the Modbus slave containing the variable to monitor	
<b>Address</b>	Direct address or symbol's name to monitor	Note 1 (see page 468)
<b>Data Type</b>	Data type of the direct address or data type of the symbol	Note 2 (see page 468)
<b>Background</b>	Graphic object background color	–
<b>Choices</b>	Selector choices Each choice is indicated in the form of a "label=value" input (when you select a "label", the "value" is sent to the device).	At least two choices required
<b>Label</b>	Label to be displayed as part of the graphic object	–
<b>Label Color</b>	Color of the label	
<b>Label Font</b>	Font used for the label	
<b>Scale Visible</b>	Indicates if a "scale", labeled with the choices should be displayed	
<b>Scale Color</b>	Color of the scale and its labels	
<b>Scale Font</b>	Font used for the scale labels	
<b>Border Width</b>	Width (in pixels) of the graphic object border	
<b>Border Color</b>	Color of the graphic object border	–

## Digital Indicator

A digital indicator gives a digital representation of the value of a variable in a device. The value may be displayed in different formats and may be set to change color when a predefined high or low limit is exceeded.

The table below describes the digital indicator's properties:

Property	Description	Limits
<b>Name</b>	Name of the graphic object	–
<b>Unit ID</b>	Address of the Modbus slave containing the variable to monitor	
<b>Address</b>	Direct address or symbol's name to monitor	Note 1 (see page 468)
<b>Data Type</b>	Data type of the direct address or data type of the symbol	Note 2 (see page 468)

Property	Description	Limits
<b>Background</b>	Graphic object background color	–
<b>Label</b>	Label to be displayed as part of the graphic object	
<b>Label Color</b>	Color of the label	
<b>Label Font</b>	Font used for the label	
<b>Value Format</b>	Format (decimal, hexadecimal, etc.) to be used to display the value on the scale	
<b>Value Precision</b>	Number of decimal places to be shown for the value on the scale (set to -1 to use a general exponential format)	-1...6
<b>Value Background</b>	Background color of the value's display zone	–
<b>Value Color</b>	Color of the value's digital display text	
<b>Value Font</b>	Font used for digital display of the value	
<b>Units</b>	Label of the value's physical units (attached to the value's digital display)	
<b>Maximum EU Value</b>	Maximum value of the variable in physical units	
<b>Minimum EU Value</b>	Minimum value of the variable in physical units	
<b>Maximum Value</b>	Maximum gross value (without scale) of the device variable	Note 3 (see page 468)
<b>Minimum Value</b>	Minimum gross value (without scale) of the device variable	Note 3 (see page 468)
<b>High High Limit Value</b>	Value of the "High High" limit expressed in physical units	–
<b>High High Limit Color</b>	Color of the indicator bar if the scale value is greater than the "High High" limit	
<b>High Limit Value</b>	Value of the "High" limit expressed in physical units	
<b>High Limit Color</b>	Color of the indicator bar if the scale value is greater than the "High" limit.	
<b>Low Limit Value</b>	Value of the "Low" limit expressed in physical units	
<b>Low Limit Color</b>	Color of the indicator bar if the scale value is less than the "Low" limit	
<b>Low Low Limit Value</b>	Value of the "Low Low" limit expressed in physical units	
<b>Low Low Limit Color</b>	Color of the indicator bar if the scale value is less than the "Low Low" limit	

Property	Description	Limits
<b>Limit Deadband</b>	Neutral range (as a percentage of the EU range) to apply to verification of the High/Low limit	0...10
<b>Border Width</b>	Width (in pixels) of the graphic object border	0...32
<b>Border Color</b>	Color of the graphic object border	–
<b>Value</b>	Simulated gross starting value (without scale) for testing the graphic object	Note 3 (see page 468)

## Message Display

A message display shows a text message based on the value of a variable in a device. For each specified message, a set value triggers its display.

The table below describes the message display's properties:

Property	Description	Limits
<b>Name</b>	Name of the graphic object	–
<b>Unit ID</b>	Address of the Modbus slave containing the variable to monitor	
<b>Address</b>	Direct address or symbol's name to monitor	Note 1 (see page 468)
<b>Data Type</b>	Data type of the direct address or data type of the symbol	Note 2 (see page 468)
<b>Background</b>	Graphic object background color	–
<b>Messages</b>	All messages to be displayed. A "value=text" input corresponds to each message (when the device value is equal to "value", the "text" message is displayed).	At least one message required
<b>Message Background</b>	Background color of the message display zone	–
<b>Message Color</b>	Message text color	
<b>Message Font</b>	Message text font	
<b>Label</b>	Label to be displayed as part of the graphic object	
<b>Label Color</b>	Color of the label	
<b>Label Font</b>	Font used for the label	0...32
<b>Border Width</b>	Width (in pixels) of the graphic object border	
<b>Border Color</b>	Color of the border of the graphic object	
<b>Value</b>	Simulated input value for testing the graphic object	Note 3 (see page 468)

## Push Button

When activated with the mouse, a push button allows you to send one or more preset values to a device.

The table below describes the push button's properties:

Property	Description	Limits
<b>Name</b>	Name of the graphic object	–
<b>Unit ID</b>	Address of the Modbus slave containing the variable to monitor	
<b>Address</b>	Direct address or symbol's name to monitor	Note 1 (see page 468)
<b>Data Type</b>	Data type of the direct address or data type of the symbol	Note 2 (see page 468)
<b>Background</b>	Background color of the graphic object	–
<b>Values</b>	Values to send to the device	Note 4 (see page 468)
<b>Reset Values</b>	Values to send to the device once the reset delay has expired. If no reset value is given, the reset will not take place.	–
<b>Reset Delay</b>	Delay (in milliseconds) that the push button must comply with between sending the values to the device and sending the reset values	0...2000
<b>Label</b>	Label to be displayed as part of the graphic object	–
<b>Label Color</b>	Color of the label	
<b>Label Font</b>	Font used for the label	
<b>Button Label</b>	Text of the button label	
<b>Button Background</b>	Button color	0...100
<b>Button Label Color</b>	Color used for the button label	–
<b>Button Label Font</b>	Font used for the button label	
<b>Border Width</b>	Width (in pixels) of the graphic object border	0...32
<b>Border Color</b>	Color of the graphic object border	–

## Direct Output Station

The direct output station allows you to enter a digital value in a text zone directly from the keyboard. If the value entered is between the upper and lower preset limits, a **Set** button is activated. In this case, the value entered is sent to the device when you click **Set** or press the ENTER key (if keyboard input is authorized for the input zone).

The table below describes the direct output station's properties:

Property	Description	Limits
<b>Name</b>	Name of the graphic object	–
<b>Unit ID</b>	Address of the Modbus slave containing the variable to monitor	
<b>Address</b>	Direct address or symbol's name to monitor	Note 1 (see page 468)
<b>Data Type</b>	Data type of the direct address or data type of the symbol	Note 2 (see page 468)
<b>Background</b>	Graphic object background color	–
<b>Label</b>	Label to be displayed as part of the graphic object	
<b>Label Color</b>	Color of the label	
<b>Label Font</b>	Font used for the label	
<b>Maximum EU Value</b>	Maximum value of the variable in physical units	
<b>Minimum EU Value</b>	Minimum value of the variable in physical units	
<b>Maximum Value</b>	Maximum gross value (without scale) of the device variable	Note 3 (see page 468)
<b>Minimum Value</b>	Minimum gross value (without scale) of the device variable	Note 3 (see page 468)
<b>Maximum Input</b>	Maximum value, expressed in physical units, authorized for the value entered in input	–
<b>Minimum Input</b>	Minimum value, expressed in physical units, authorized for the value entered in input	
<b>Border Width</b>	Width (in pixels) of the graphic object border	0...32
<b>Border Color</b>	Color of the graphic object border	–

## Indicator Light

The indicator light provides a double indication of the value of a variable in a device. If the Input Inverted property is not set to TRUE, a zero input value is declared as being OFF and a non-zero value is declared as being ON. If the Flash Interval property is set to a positive value, the indicator light will flash when the input value is equal to ON.



The table below describes the indicator light's properties:

Property	Description	Limits
<b>Name</b>	Name of the graphic object	–
<b>Unit ID</b>	Address of the Modbus slave containing the variable to monitor	
<b>Address</b>	Direct address or symbol's name to monitor	Note 1 (see page 468)
<b>Data Type</b>	Data type of the direct address or data type of the symbol	Note 2 (see page 468)
<b>Background</b>	Graphic object background color	–
<b>Label</b>	Label to be displayed as part of the graphic object	
<b>Label Color</b>	Color of the label	
<b>Label Font</b>	Font used for the label	
<b>Off Word</b>	Text to be displayed when the input value is OFF	
<b>Off Word Background</b>	Background color of the indicator light when <b>Off Word</b> is displayed	
<b>Off Word Color</b>	Color of the <b>Off Word</b> text	
<b>Off Word Font</b>	Font used for the <b>Off Word</b> text	
<b>On Word</b>	Text to be displayed when the input value is ON	
<b>On Word Background</b>	Background color of the indicator light when <b>On Word</b> is displayed	
<b>On Word Color</b>	Color of the <b>On Word</b> font	
<b>On Word Font</b>	Font used for the <b>On Word</b> text	
<b>Flash Interval</b>	The flashing time for the indicator light (expressed in milliseconds) when the input value is ON. Set to zero for no flashing.	
<b>Shape</b>	Shape (circular, rectangular, etc.) of the indicator light	–
<b>Input Inverted</b>	On <b>TRUE</b> , inverts the input value. (The indicator displays <b>Off Word</b> when the input value is ON.)	
<b>Border Width</b>	Width (in pixels) of the graphic object border	0...32
<b>Border Color</b>	Color of the border of the graphic object	–
<b>Value</b>	Simulated input value for testing the graphic object	Note 3 (see page 468)

## Motor Control Station

The motor control station is designed to imitate the on/off push button standard station which is frequently used to control the motors. This graphic object is essentially composed of two push buttons and an indicator light. To facilitate the configuration of this object's many properties, a custom module is provided. It is by means of this module, and not the **Graphic Editor** properties sheet, that all the properties (apart from the name) are configured.

The table below describes the motor control station's properties:

Property	Description	Limits
<b>Name</b>	Name of the graphic object	–
<b>Background</b>	Graphic object background color	
<b>Label</b>	Label to be displayed as part of the graphic object	
<b>Label Color</b>	Color of the label	
<b>Label Font</b>	Font used for the label	
<b>Border Width</b>	Width (in pixels) of the graphic object border	0...32
<b>Border Color</b>	Color of the graphic object border	–
<b>Indicator Light</b>	Properties identical to those of the Indicator Light graphic object with the exception of the shared properties listed above	
<b>Top Push Button</b>	Properties identical to those of the Push Button graphic object with the exception of the shared properties listed above	
<b>Bottom Push Button</b>	Properties identical to those of the Push Button graphic object with the exception of the shared properties listed above	

## Analog Meter

An analog meter gives an analog representation of the value of a variable in a device. It is represented by a pointer on a circular dial whose position corresponds to a percentage of its range in physical units. You can set the size of the meter's circular dial (circle degrees sweep), its colors and the style of the pointer.

The table below describes the analog meter's properties:

Property	Description	Limits
<b>Name</b>	Name of the graphic object	–
<b>Unit ID</b>	Address of the Modbus slave containing the variable to monitor	
<b>Address</b>	Direct address or symbol's name to monitor	Note 1 (see page 468)

Property	Description	Limits
<b>Data Type</b>	Data type of the direct address or data type of the symbol	Note 2 (see page 468)
<b>Background</b>	Graphic object background color	–
<b>Label</b>	Label to be displayed as part of the graphic object	
<b>Label Color</b>	Color of the label	
<b>Label Font</b>	Font used for the label	
<b>Major Scale Divisions</b>	Number of major scale divisions (marked)	0...100
<b>Minor Scale Divisions</b>	Number of minor scale divisions (not marked)	0...100
<b>Scale Color</b>	Color of the scale and its labels	–
<b>Scale Font</b>	Font used for the scale labels	
<b>Scale Precision</b>	Number of decimal places to be shown for the scale labels (set to -1 to use a general exponential format)	-1...6
<b>Maximum EU Value</b>	Maximum value of the variable in physical units	–
<b>Minimum EU Value</b>	Minimum value of the variable in physical units	
<b>Maximum Value</b>	Maximum gross value (without scale) of the device variable	Note 3 (see page 468)
<b>Minimum Value</b>	Minimum gross value (without scale) of the device variable	Note 3 (see page 468)
<b>Dial Degrees Sweep</b>	Portion of circular arc to be used to draw the dial	60...300

Property	Description	Limits
<b>Pointer Type</b>	Type of pointer used (needle, arrow, etc.)	–
<b>Pointer Color</b>	Color used for the pointer	
<b>Dial Color</b>	Color used for the dial (for the part in the High/Low range)	
<b>High High Limit Value</b>	Value of the "High High" limit expressed in physical units	
<b>High High Limit Color</b>	Color of the indicator bar if the scale value is greater than the "High High" limit	
<b>High Limit Value</b>	Value of the "High" limit expressed in physical units	
<b>High Limit Color</b>	Color of the indicator bar if the scale value is greater than the "High" limit.	
<b>Low Limit Value</b>	Value of the "Low" limit expressed in physical units	
<b>Low Limit Color</b>	Color of the indicator bar if the scale value is less than the "Low" limit	
<b>Low Low Limit Value</b>	Value of the "Low Low" limit expressed in physical units	
<b>Low Low Limit Color</b>	Color of the indicator bar if the scale value is less than the "Low Low" limit	
<b>Border Width</b>	Width (in pixels) of the graphic object border	
<b>Border Color</b>	Color of the graphic object border	–
<b>Value</b>	Simulated gross starting value (without scale) for testing the graphic object	Note 3 (see page 468)

## Rotary Slider

A rotary slider gives an analog representation of the value of a variable in a device. It is represented by a knob on a circular dial whose position corresponds to a percentage of its range in physical units. You can set the size of the dial and the color of the knob. Using the mouse, you can change the position of the knob by sending a new value to the device.

The table below describes the rotary slider's properties:

Property	Description	Limits
<b>Name</b>	Name of the graphic object	–
<b>Unit ID</b>	Address of the Modbus slave containing the variable to monitor	
<b>Address</b>	Direct address or symbol's name to monitor	Note 1 (see page 468)

Property	Description	Limits
<b>Data Type</b>	Data type of the direct address or data type of the symbol	Note 2 (see page 468)
<b>Background</b>	Graphic object background color	–
<b>Label</b>	Label to be displayed as part of the graphic object	
<b>Label Color</b>	Color of the label	
<b>Label Font</b>	Font used for the label	
<b>Major Scale Divisions</b>	Number of major scale divisions (marked)	0...100
<b>Minor Scale Divisions</b>	Number of minor scale divisions (not marked)	0...100
<b>Scale Color</b>	Color of the scale and its labels	–
<b>Scale Font</b>	Font used for the scale labels	
<b>Scale Precision</b>	Number of decimal places to be shown for the scale labels (set to -1 to use a general exponential format)	-1...6
<b>Dial Degrees Sweep</b>	Portion of circular arc to be used to draw the dial	60...300
<b>Dial Color</b>	Color of the dial	–
<b>Knob Color</b>	Color used for the knob	
<b>Maximum EU Value</b>	Maximum value of the variable in physical units	
<b>Minimum EU Value</b>	Minimum value of the variable in physical units	
<b>Maximum Value</b>	Maximum gross value (without scale) of the device variable	
<b>Minimum Value</b>	Minimum gross value (without scale) of the device variable	Note 3 (see page 468)
<b>Border Width</b>	Width (in pixels) of the graphic object border	0...32
<b>Border Color</b>	Color of the graphic object border	–

## Rotary Selector

A rotary selector allows you to choose from a number of options. Once the selection has been made, the value corresponding to the choice is sent to the device. The choices are represented by the marks on a "scale", the current selection being indicated by the position of the knob. The size of the circular dial (circle degrees sweep) and the color of the knob can be configured.

The table below describes the rotary selector's properties:

Property	Description	Limits
<b>Name</b>	Name of the graphic object	–
<b>Unit ID</b>	Address of the Modbus slave containing the variable to monitor	
<b>Address</b>	Direct address or symbol's name to monitor	Note 1 (see page 468)
<b>Data Type</b>	Data type of the direct address or data type of the symbol	Note 2 (see page 468)
<b>Background</b>	Graphic object background color	–
<b>Choices</b>	Selector choices Each choice is indicated in the form of a "label=value" input (when you select a "label", the "value" is sent to the device).	At least two choices required
<b>Label</b>	Label to be displayed as part of the graphic object	–
<b>Label Color</b>	Color of the label	
<b>Label Font</b>	Font used for the label	
<b>Scale Visible</b>	Indicates if a "scale", labeled with the choices should be displayed	
<b>Scale Color</b>	Color of the scale and its labels	
<b>Scale Font</b>	Font used for the scale labels	
<b>Dial Degrees Sweep</b>	Portion of circular arc to be used to draw the dial	
<b>Knob Color</b>	Color used for the knob	–
<b>Border Width</b>	Width (in pixels) of the graphic object border	0...32
<b>Border Color</b>	Color of the graphic object border	–

## Trend Recorder

A trend recorder enables you to obtain a continuous, time-based graphic of the values of a maximum of six variables in a device. It emulates a strip-chart recorder, with the pens on the right and the "paper" moving from right to left. A vertical scale to the left of the graphic indicates the range of registered values and a horizontal scale beneath the graphic displays the range's time frame. You can set the update frequency and the appearance of the graphic.

To facilitate the configuration of the many properties of this object, a custom module is provided. It is by means of this module, and not the **Graphic Editor** properties sheet, that all the properties (apart from the name) are set.

The table below describes the trend recorder's properties and the properties available for each of the pens are described in the second table:

Property	Description	Limits
<b>Name</b>	Name of the graphic object	–
<b>Background</b>	Graphic object background color	
<b>Label</b>	Label to be displayed as part of the graphic object	
<b>Label Color</b>	Color of the label	
<b>Label Font</b>	Font used for the label	
<b>Major Scale Divisions</b>	Number of major scale divisions (marked)	0...100
<b>Minor Scale Divisions</b>	Number of minor scale divisions (not marked)	0...100
<b>Scale Color</b>	Color of the scale and its labels	–
<b>Scale Font</b>	Font used for the scale labels	
<b>Scale Precision</b>	Number of decimal places to be shown for the scale labels (set to -1 to use a general exponential format)	-1...6
<b>Maximum EU Value</b>	Maximum value of the variable in physical units	–
<b>Minimum EU Value</b>	Minimum value of the variable in physical units	
<b>Update Period</b>	Graphic update interval (in seconds)	0.5...120
<b>Time Scale Divisions</b>	Number of divisions on the horizontal scale	0...6
<b>Chart Background</b>	Color of the graphic zone	–
<b>Grid Color</b>	Color of the grid drawn in the graphic zone	
<b>Vertical Grid Divisions</b>	Number of vertical divisions in the grid	0...100
<b>Border Width</b>	Width (in pixels) of the graphic object border	0...32
<b>Border Color</b>	Color of the graphic object border	–

The following trend recorder properties are available for each pen:

Property	Description	Limits
<b>Unit ID</b>	Address of the Modbus slave containing the variable to monitor	–
<b>Address</b>	Direct address or symbol's name to monitor	Note 1 (see page 468)

Property	Description	Limits
<b>Data Type</b>	Data type of the direct address or data type of the symbol	Note 2 (see page 468)
<b>Maximum Value</b>	Maximum gross value (without scale) of the device variable	Note 3 (see page 468)
<b>Minimum Value</b>	Minimum gross value (without scale) of the device variable	Note 3 (see page 468)
<b>Pen Color</b>	Color of the "pen" which allows the value placed on the scale to be recorded	–
<b>Pen Label</b>	Label used to identify the pen	

## Display Link

A display link is a special graphic object which allows you to move to another graphic display by clicking it. To indicate that the object represents a link towards another display, the link's text label is underlined and the mouse cursor changes to a hand when it passes over it. This object is especially useful when the **Graphic Editor** is used in **View mode** which has no drop-down list of graphic displays.

A display link can also be used as a hypertext link to an HTML file. If you enter a URL such as Link Display Name, you can open it in a new browser window by pressing the SHIFT key while clicking on the link. If you only click on the link, the existing browser window is replaced by the URL.

If the Link Display Name is blank, the **label** is not displayed underlined and the object displayed becomes a simple text label.

The table below describes the display link's properties:

Property	Description
<b>Label</b>	Label of the link
<b>Link Display Name</b>	Name of the graphic display to load when the user clicks on the link, or URL of a web page
<b>Label Color</b>	Color of the label
<b>Label Font</b>	Font used for the label

## Datalogging History

A Datalogging History provides a continuous, time-based charting of the value of up to six symbols (variables) coming from the log file of the Datalogging service. A Datalogging History emulates a strip-chart recorder, with the pens on the right, and the "paper" moving from right to left. A vertical scale can be shown on the left side of the chart for showing the range of the values being recorded, and a horizontal scale can be shown below the chart for showing the time span of the chart.



In order to make it easier to set this object's many properties, a Customizer is provided. All properties (except Name) are set with its Customizer, not with the **Graphic Editor's** Property Sheet.

3 buttons are available in edition and animation mode:

- **Reload:** the Datalogging History object is a static widget. This button enables to refresh the value used to build the chart.
- **+: zoom on the trend.** It decreases the time scale in order to have a better vision of a part of the trend.
- **-: zoom out on the trend.** It increases the time scale in order to have a larger vision of the trend.

If you place the cursor of the mouse on a point of the trend, a tooltip appears to showing the exact value at that point. Click on that point to display the tooltip permanently. Right click on it to make it disappear.

The following table describes properties for the Datalogging History. Properties available for each pen are described in the next table:

Property	Description	Limits
<b>Name</b>	The name for the graphic object	–
<b>Background</b>	The background color for the graphic object	
<b>Label</b>	The label to be displayed as part of the graphic object	
<b>Label Color</b>	The color for the label	
<b>Label Font</b>	The font for the label	
<b>Major Scale Divisions</b>	The number of major (labeled) scale divisions	0...100
<b>Minor Scale Divisions</b>	The number of minor (unlabeled) scale divisions	0...100
<b>Scale Color</b>	The color for the scale and its labels	–
<b>Scale Font</b>	The font for scale labels	
<b>Scale Precision</b>	The number of fractional digits to be shown for scale labels (Set to -1 to use a general exponential format.)	-1...6
<b>Maximum EU Value</b>	The maximum value, in engineering units, of the symbol (variable)	–
<b>Minimum EU Value</b>	The minimum value, in engineering units, of the symbol (variable)	
<b>Update Period</b>	The update interval (in seconds) for the chart	0.5...120
<b>Time Scale Divisions</b>	The number of horizontal scale divisions	0...6
<b>Chart Background</b>	The color for the chart area	–
<b>Grid Color</b>	The color of the grid drawn in the chart area	

Property	Description	Limits
<b>Vertical Grid Divisions</b>	The number of vertical divisions for the grid	0...100
<b>Border Width</b>	The width (in pixels) for the graphic object's border	0...32
<b>Border Color</b>	The color for the graphic object's border	–


These Datalogging History properties are available for each pen:

Property	Description	Limits
<b>Name of the CSV file</b>	The name of the CSV file used to build the trend. Location: <ul style="list-style-type: none"> <li>● Default (no path): the file is located on the FLASH memory.</li> <li>● /CFA00/USERDATA/TABLEx: the file is located on the CF card.</li> <li>● /USBHD/00/USERDATA/TABLEx: the file is located on the USB memory.</li> <li>● /RAMDISK/USERDATA/TABLEx: the file is located on the saved RAM.</li> </ul> <b>Note:</b> the log file must include timestamps (see page 309).	–
<b>Address</b>	The name of a symbol (variable) to monitor.	Note 1 (see page 468)
<b>Data Type</b>	The data type of the symbol (variable). <b>Note:</b> the data type must be numerical.	–
<b>Maximum PLC Value</b>	The maximum raw (unscaled) value of the symbol (variable) in the PLC.	Note 3 (see page 468)
<b>Minimum PLC Value</b>	The minimum raw (unscaled) value of the symbol (variable) in the PLC.	Note 3 (see page 468)
<b>Pen Color</b>	The color of the "pen" used to record the scaled value.	–
<b>Pen Label</b>	The label used to identify the pen.	

## Extended Graphic Objects

### Presentation

The extended graphic objects available in the Graphic Editor are designed to help you to create graphic displays imitating advanced graphic display panels. All the data control and monitoring objects have integrated communication functions and are designed as standalone graphic objects.

<b> WARNING</b>
<b>UNINTENDED EQUIPMENT OPERATION</b>
Do not use data control graphic objects to send information of critical nature concerning human and material integrity.
<b>Failure to follow these instructions can result in death, serious injury, or equipment damage.</b>

In addition, to help clients who wish to insert several simple applets into a single HTML page, all objects in the Graphic Editor exist in the form of applets. When combined with the *LiveBeanApplet*, the graphic objects in the Graphic Editor can be used in the same way as the *LiveLabelApplet*.

### ASCII Text Editor

The ASCII text editor is based on the message display graphic element. It enables new text to be entered.

The properties of the ASCII text editor are as follows:

Property	Description	Limits
<b>Name</b>	Name of the graphic object	
<b>Unit ID</b>	Address of the Modbus slave containing the variable to monitor	
<b>Address</b>	Direct address to monitor	Note 1 (see page 468)
<b>Max. Text Length</b>	Maximum length of the text	
<b>Text Color</b>	Color of the text	
<b>Text Font</b>	Font of the text	
<b>Swap Bytes</b>	False if the target byte order is the same as that of the PC	
<b>Value</b>	The text itself	

## Bar Graph

A bar graph gives an analog representation of the value of a variable in a device. It draws a vertical bar whose length is proportional to the value and represents a percentage of its range in physical units.

The properties of the bar graph are as follows:

Property	Description	Limits
<b>Name</b>	Name of the graphic object	
<b>Unit ID</b>	Address of the Modbus slave containing the variable to monitor	
<b>Address</b>	Direct address to monitor	Note 1 (see page 468)
<b>Data Type</b>	Data type of the direct address	Note 2 (see page 468)
<b>Background</b>	Graphic object background color	
<b>Label</b>	Label to be displayed as part of the graphic object	
<b>Label Color</b>	Color of the label	
<b>Label Font</b>	Font used for the label	
<b>Maximum EU Value</b>	Maximum value of the variable in physical units	
<b>Minimum EU Value</b>	Minimum value of the variable in physical units	
<b>Maximum Value</b>	Maximum gross value (without scale) of the device variable	Note 3 (see page 468)
<b>Minimum Value</b>	Minimum gross value (without scale) of the variable in the device	Note 3 (see page 468)
<b>Bar Background</b>	Background color of the indicator bar	
<b>Bar Color</b>	Color of the indicator bar (if the scale value is within the High/Low range)	
<b>High High Limit Value</b>	Value of the "High High" limit expressed in physical units	
<b>High High Limit Color</b>	Color of the indicator bar if the scale value is greater than the "High High" limit	
<b>High Limit Value</b>	Value of the "High" limit expressed in physical units	
<b>High Limit Color</b>	Color of the indicator bar if the scale value is greater than the "High" limit.	
<b>Low Limit Value</b>	Value of the "Low" limit expressed in physical units	

Property	Description	Limits
<b>Low Limit Color</b>	Color of the indicator bar if the scale value is less than the "Low" limit	
<b>Low Low Limit Value</b>	Value of the "Low Low" limit expressed in physical units	
<b>Low Low Limit Color</b>	Color of the indicator bar if the scale value is less than the "Low Low" limit	
<b>Limit Deadband</b>	Neutral range (as a percentage of the EU range) to apply to verification of the High/Low limit	0 to 10
<b>Border Width</b>	Width (in pixels) of the graphic object border	0 to 32
<b>Border Color</b>	Color of the graphic object border	
<b>Value</b>	Simulated gross input value (without scale) for testing the graphic object	Note 3 (see page 468)

## Bitmap

The bitmap interface graphic element displays a static bitmap on the screen.

The properties of the bitmap interface graphic element are as follows:

Property	Description	Limits
<b>Name</b>	Name of the graphic object	
<b>Background</b>	Graphic object background color	Note 1 (see page 468)
<b>Label</b>	Label to be displayed as part of the graphic object	
<b>Label Color</b>	Color of the label	
<b>Label Font</b>	Font used for the label	
<b>Border Width</b>	Width (in pixels) of the graphic object border	
<b>Border Color</b>	Color of the graphic object border	
<b>Bitmap Choices</b>	File names of custom bitmaps to display. Refer to the next paragraph for information on adding images into the module.	

## Adding Images

You can add images into the module in one of the following way:

- Add your images into the *images.zip* file (path: */NAND/FLASH1/wwwroot*).
- Create a directory into the module (i.e. */NAND/FLASH1/wwwroot/bitmaps*). Copy your images into this directory. In this case, you need to specify the path of the images you want to use (i.e. */NAND/FLASH1/wwwroot/bitmaps/key.gif*).

Step	Action
1	Create an <i>images</i> folder on your PC.
2	Copy the images you want to use in this folder.
3	Import the <i>user.jar</i> file from the TSX ETG 30** to the PC (path: <i>/NAND/FLASH1/wwwroot/classes</i> ) using a FTP client.
4	Open the <i>user.jar</i> file using a file archiver.
5	Drag and drop the <i>images</i> folder in the <i>user.jar</i> file and check that the relative path of the image files is 'images/"/>.
6	Transfer the <i>user.jar</i> file back to the module using a FTP client.

### Generic Bitmap

The generic Bitmap interface graphic element can display a static bitmap for each separate value of a variable. It can be used to display dynamic animations, such as the variation in level of a reservoir.

The properties of the generic Bitmap interface graphic element are as follows:

Property	Description	Limits
<b>Name</b>	Name of the graphic object	
<b>Unit ID</b>	Address of the Modbus slave containing the variable to monitor	
<b>Address</b>	Direct address to monitor	Note 1 (see page 468)
<b>Data Type</b>	Data type of the direct address	Note 2 (see page 468)
<b>Background</b>	Graphic object background color	Note 1 (see page 468)
<b>Label</b>	Label to be displayed as part of the graphic object	
<b>Label Color</b>	Color of the label	
<b>Label Font</b>	Font used for the label	
<b>Bitmap Choices</b>	File names of custom bitmaps to display. Refer to the previous paragraph for information on adding images into the module.	
<b>Border Width</b>	Width (in pixels) of the graphic object border	
<b>Border Color</b>	Color of the graphic object border	
<b>Value</b>	Simulated starting value for testing the behavior of the graphic object	

## Graphic Link

A graphic link is a special graphic object that moves to another graphic display when you click it. Graphic links can also be recognized by their underlined labels, and by the fact that the mouse cursor changes to a hand when it passes over them. They are especially useful when the Graphic Editor is used in Display mode, in which there is no pull-down list of graphic displays.

A graphic link can also be used as a hypertext link to an HTML file. If you enter a URL such as **Link Display Name**, you can open the URL in a new browser window by pressing the SHIFT key while clicking on the link. If you only click on the link, the URL will open in the existing browser window.

If the **Link Display Name** is not filled in, the label will not be underlined and the object displayed becomes a simple text label.

The properties of the graphic link are as follows:

Property	Description	Limits
<b>Label</b>	Label of the link	
<b>Link Display Name</b>	Name of the graphic display to load when the user clicks on the link, or URL of a Web page	
<b>Label Color</b>	Color of the label	
<b>Label Font</b>	Font used for the label	
<b>Bitmap Choices</b>	Name of the bitmap file to be clicked on	

## Indicator Light

The indicator light displays the value of a variable in a device. The input value of 0 is equal to OFF, and any value other than 0 is equal to ON. If the **Flash Interval** property is set to a positive value, the indicator light will flash when the input value is equal to ON. There is one bitmap for the ON state and another for the OFF state.

The properties of the indicator light are as follows:

Property	Description	Limits
<b>Name</b>	Name of the graphic object	
<b>Unit ID</b>	Address of the Modbus slave containing the variable to monitor	
<b>Address</b>	Direct address to monitor	Note 3 (see page 468)
<b>Data Type</b>	Data type of the direct address	Note 2 (see page 468)
<b>Background</b>	Graphic object background color	
<b>Label</b>	Label to be displayed as part of the graphic object	
<b>Label Color</b>	Color of the label	

Property	Description	Limits
<b>Label Font</b>	Font used for the label	
<b>OFF Word</b>	Text to be displayed when the input value is OFF	
<b>OFF Bitmap Choice</b>	Indicator bitmap when the OFF word is displayed	
<b>OFF Word Color</b>	Color of the OFF word text	
<b>OFF Word Font</b>	Font of the OFF word text	
<b>ON Word</b>	Text to be displayed when the input value is ON	
<b>ON Bitmap Choice</b>	Indicator bitmap when the ON word is displayed	
<b>ON Word Color</b>	Color of the ON word font	
<b>ON Word Font</b>	Font of the ON word text	
<b>Flash Interval</b>	The flashing time for the indicator light (expressed in milliseconds) when the input value is ON. Set to 0 for no flashing.	200 to 2,000
<b>Input Inverted</b>	On TRUE, inverts the input value. (The indicator displays the OFF word when the input value is ON.)	
<b>Border Width</b>	Width (in pixels) of the graphic object border	0 to 32
<b>Border Color</b>	Color of the graphic object border	
<b>Value</b>	Simulated input value for testing the graphic object	Note 3 (see page 468)

## Motor

The Motor graphic interface element displays the value of a variable in a device. The input value of 0 is equal to OFF, the value 1 is equal to ON and other values are equal to DEFAULT. These three states are represented by different bitmaps.

The properties of the Motor graphic interface element are as follows:

Property	Description	Limits
<b>Name</b>	Name of the graphic object	
<b>Unit ID</b>	Address of the Modbus slave containing the variable to monitor	
<b>Address</b>	Direct address to monitor	Note 1 (see page 468)
<b>Data Type</b>	Data type of the direct address	Note 2 (see page 468)
<b>Background</b>	Graphic object background color	
<b>Label</b>	Label to be displayed as part of the graphic object	
<b>Label Color</b>	Color of the label	



Property	Description	Limits
<b>Label Font</b>	Font used for the label	
<b>OFF Word</b>	Text to be displayed when the input value is OFF	
<b>OFF Bitmap Choice</b>	Motor bitmap when the OFF word is displayed	
<b>OFF Word Color</b>	Color of the OFF word text	
<b>OFF Word Font</b>	Font of the OFF word text	
<b>ON Word</b>	Text to be displayed when the input value is ON	
<b>ON Bitmap Choice</b>	Motor bitmap when the ON word is displayed	
<b>ON Word Color</b>	Color of the ON word font	
<b>ON Word Font</b>	Font of the ON word text	
<b>DEFAULT Word</b>	Text to be displayed when the input value is ON	
<b>DEFAULT Bitmap Choice</b>	Motor bitmap when the DEFAULT word is displayed	
<b>DEFAULT Word Color</b>	Color of the DEFAULT word font	
<b>DEFAULT Word Font</b>	Font of the DEFAULT word text	
<b>Border Width</b>	Width (in pixels) of the graphic object border	0 to 32
<b>Border Color</b>	Color of the graphic object border	
<b>Value</b>	Simulated input value for testing the graphic object	Note 3 (see page 468)

## Pipe

A pipe displays the value of a variable in a device that has two possible states. The input value of 0 is equal to OFF, and any value other than 0 is equal to ON. There is one bitmap for the ON state and another for the OFF state.

The properties of the pipe are as follows:

Property	Description	Limits
<b>Name</b>	Name of the graphic object	
<b>Unit ID</b>	Address of the Modbus slave containing the variable to monitor	
<b>Address</b>	Direct address to monitor	Note 1 (see page 468)
<b>Data Type</b>	Data type of the direct address	Note 2 (see page 468)
<b>Background</b>	Graphic object background color	
<b>Label</b>	Label to be displayed as part of the graphic object	

Property	Description	Limits
<b>Label Color</b>	Color of the label	
<b>Label Font</b>	Font used for the label	
<b>OFF Word</b>	Text to be displayed when the input value is OFF	
<b>OFF Bitmap Choice</b>	Pipe bitmap when the OFF word is displayed	
<b>OFF Word Color</b>	Color of the OFF word text	
<b>OFF Word Font</b>	Font of the OFF word text	
<b>ON Word</b>	Text to be displayed when the input value is ON	
<b>ON Bitmap Choice</b>	Pipe bitmap when the ON word is displayed	
<b>ON Word Color</b>	Color of the ON word font	
<b>ON Word Font</b>	Font of the ON word text	
<b>Border Width</b>	Width (in pixels) of the graphic object border	0 to 32
<b>Border Color</b>	Color of the graphic object border	
<b>Value</b>	Simulated input value for testing the graphic object	Note 3 (see page 468)

## Push button

When activated with the mouse, a push button allows the user to send one or more preset values to a device.

The properties of the push button are as follows:

Property	Description	Limits
<b>Name</b>	Name of the graphic object	
<b>Unit ID</b>	Address of the Modbus slave containing the variable to monitor	
<b>Address</b>	Direct address to monitor	Note 3 (see page 468)
<b>Data Type</b>	Data type of the direct address	Note 2 (see page 468)
<b>Background</b>	Graphic object background color	
<b>Values</b>	Values to send to the device	Note 4 (see page 468)
<b>Reset Values</b>	Values to send to the device once the reset delay has expired. If no reset value is given, the reset will not take place.	
<b>Reset Delay</b>	Delay (in milliseconds) that the push button must comply with between sending the values to the device and sending the reset values.	0-2000

Property	Description	Limits
<b>Label</b>	Label to be displayed as part of the graphic object	
<b>Label Color</b>	Color of the label	
<b>Label Font</b>	Font used for the label	
<b>Button Label</b>	Button label text	
<b>Button Label Color</b>	Color used for the button label	
<b>Button Label Font</b>	Font used for the button label	
<b>OFF Bitmap Choice</b>	Button bitmap when the OFF state is displayed	
<b>ON Bitmap Choice</b>	Button bitmap when the ON state is displayed	
<b>Border Width</b>	Width (in pixels) of the graphic object border	0 to 32
<b>Border Color</b>	Color of the graphic object border	

## Distributor

A distributor displays the value of a variable in a device that has two possible states. The input value of 0 is equal to OFF, and any value other than 0 is equal to ON. There is one bitmap for the ON state and another for the OFF state.

The properties of the distributor are as follows:

Property	Description	Limits
<b>Name</b>	Name of the graphic object	
<b>Unit ID</b>	Address of the Modbus slave containing the variable to monitor	
<b>Address</b>	Direct address to monitor	Note 1 (see page 468)
<b>Data Type</b>	Data type of the direct address	Note 2 (see page 468)
<b>Background</b>	Graphic object background color	
<b>Label</b>	Label to be displayed as part of the graphic object	
<b>Label Color</b>	Color of the label	
<b>Label Font</b>	Font used for the label	
<b>OFF Word</b>	Text to be displayed when the input value is OFF	
<b>OFF Bitmap Choice</b>	Distributor bitmap when the OFF word is displayed	
<b>OFF Word Color</b>	Color of the OFF word text	
<b>OFF Word Font</b>	Font of the OFF word text	
<b>ON Word</b>	Text to be displayed when the input value is ON	

Property	Description	Limits
<b>ON Bitmap Choice</b>	Distributor bitmap when the ON word is displayed	
<b>ON Word Color</b>	Color of the ON word font	
<b>ON Word Font</b>	Font of the ON word text	
<b>Flash Interval</b>	The flashing time for the indicator light (expressed in milliseconds) when the input value is ON. Set to 0 for no flashing.	200 to 2,000
<b>Border Width</b>	Width (in pixels) of the graphic object border	0 to 32
<b>Border Color</b>	Color of the graphic object border	
<b>Value</b>	Simulated input value for testing the graphic object	Note 3 (see page 468)

## Notes

The notes relating to this section are as follows:

1.	A direct Modbus address is a number between 0 and 65535.	
2.	The various values of the Data Type property for a direct Modbus address have the following meanings:	
	Type	Signification
	COIL	output bit (Boolean)
	DISCRETE INPUT	input bit (Boolean)
	REGISTER	16-bit signed integer
	INT32	32-bit signed integer
	INT32SWAP	32-bit signed integer with least significant and most significant words inverted
	INPUT REGISTER	16-bit signed integer for analog input
3.	The limits of the Maximum PLC Value and Minimum PLC Value properties are the natural limits of the configured Data Type property.	
4.	For a push button you must specify at least one value. If several values are entered, they will be assigned to an address table starting with the direct address indicated.	
5.	For the applet to display a numerical data value instead of a label, specify parameters in the HTML code as follows: name = "label" value = "\$data\$".	

## 7.5 Graphic Viewer

---

### Graphic Viewer

#### Presentation

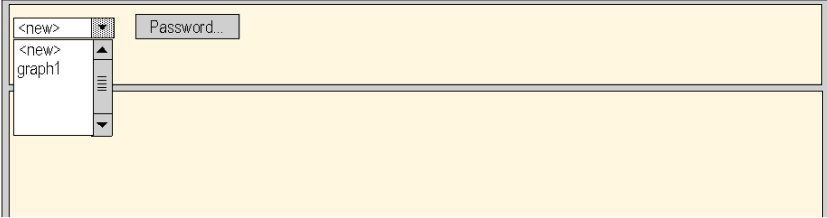
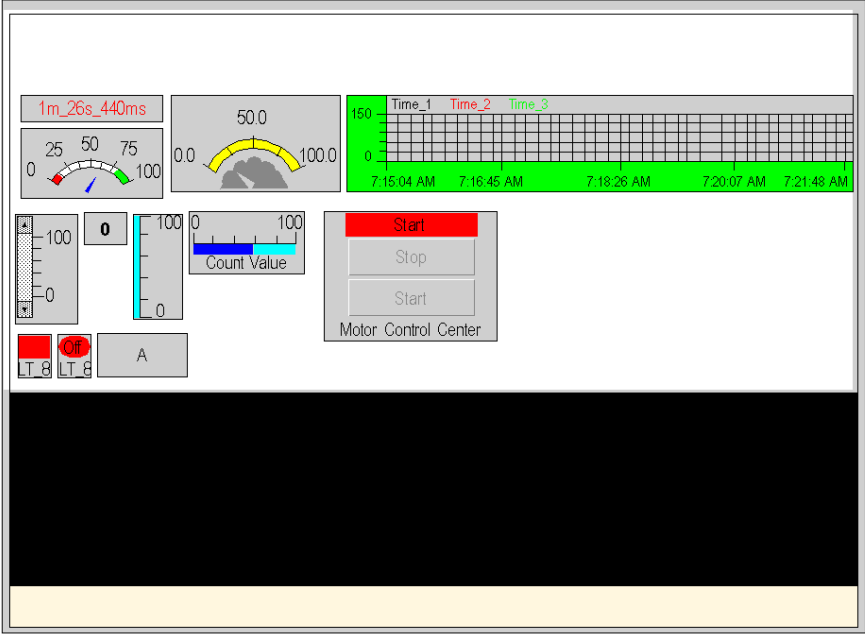
The Graphic Viewer allows you to display graphical screens created with Web Designer software. It is dedicated for operator dialog. With the Graphic Viewer you can only visualize Graphic screens. Full screen display is possible.

The Graphic Viewer is made up of two windows:

- **Top Window:** This area displays the user commands. You can select a graphic to modify in the drop-down menu.
- **Display Window:** This area displays the graphics selected.

When you view a graphic in full screen mode the **Top Window** is not displayed.

## Selecting a Graphic

Step	Action
1	<p>Click <b>Graphic Viewer</b>.</p> <p><b>Result:</b> The <b>top window and display window</b> appears:</p> 
2	<p>Select a graphic from the drop-down menu.</p> <p><b>Result:</b> The graphic selected appears in the <b>display window</b>:</p>  <p><b>Note:</b> Click the active graphic's <b>display window</b> twice (unless the active graphic is a command type interface graphic element) to refresh the <b>top window</b>.</p>

---

# Creating Custom Web Pages



---

## Scope of this Chapter

This chapter describes how to create and upload custom pages. It also describes the graphic objects provided on the CD-ROM. The TSX ETG 30•• can host user defined Web pages and any other documents such as maintenance documents, schematic documents and any DOC, PDF, XLS file that can help users in operations and maintenance. This capability helps you to create a fully customized human machine interface embedded in the module. It will perfectly match your needs with all the openness given by Web pages (Hyperlinks to other websites, display of PDF, DOC, XLS documents, custom animations (Flash, Silverlight, etc.))

The TSX ETG 30•• CD-ROM includes a library of graphic objects in Java applet format that you can include in your HTML pages. The objects can be used to give a graphical and dynamic representation of register values, coil values, etc., for Modbus devices. A macro (or file) in Microsoft FrontPage format is provided to help the user to create HTML pages more easily, along with a description of the graphic objects.

### NOTE:

When planning custom Web pages, you must keep them within the limits of the memory available for customization:

- If you are using a third party software to transfer a website and if you reach the limits of the memory, some pages of the website will not appear.
- If you are using Web Designer for TSX ETG 30•• to transfer a website, it will prompt you if your website exceeds the memory capacity.

## What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
8.1	Working with Custom Pages	472
8.2	Creating Custom Pages with an HTML Editor	477
8.3	Creating Custom Pages with FrontPage	489
8.4	Description of Graphic Objects	509

## 8.1 Working with Custom Pages

---

### Overview

You may develop any custom pages you want in order to customize the Web site. The TSX ETG 30\*\* module gives you 3 different methods for adding these pages to the site:

- You may add custom Web pages and give them password protection.
- You may add custom Web pages and make them available to all users.
- You may replace the default home page with one of your own.

### What's in this Section?

This section contains the following topics:

Topic	Page
Uploading Custom Pages via an FTP Client	473
Uploading a Custom Home Page	475



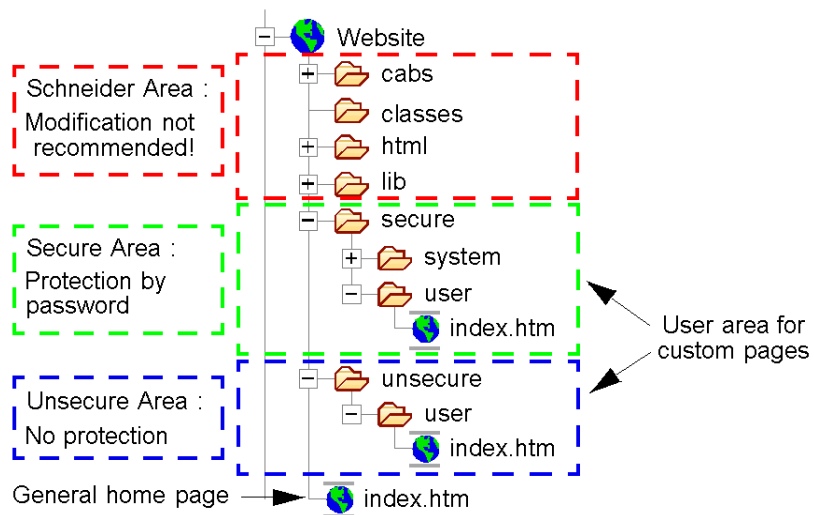
## Uploading Custom Pages via an FTP Client

### Presentation

If you add custom Web pages to the default website, you may choose to protect them with the same user name and the same password as those used for the default pages, or to authorize access to all users (no password required).

To add custom pages to the site, you must specify the folder in which the pages are stored before you uploading them to the server.

### Website Structure



### Indicating the File Location

Place the folders containing your Web pages and any associated graphics in a shared directory. Each subdirectory should normally contain a page called *index.htm* that is the home page of the site.

All graphics file names must be in DOS 8.3 format (a maximum of eight characters for the name and three characters for the extension).

**NOTE:** Do not forget to create new hyperlinks to other Web pages to enable these pages to be accessed via the browser.

### Uploading to a Web site

Step	Action
1	Save your project.
2	Open an FTP client (e.g.: filezilla).
3	Enter the IP address, the user name and the password to connect to the site (default: <b>wsupgrade/wsupgrade</b> ).
4	Find your Web project in the tree structure on your local disk.
5	<p>Depending on the type of pages you want to add, the location of the pages on the module can be different:</p> <ul style="list-style-type: none"> <li>● Pages accessible to all must be copied to the <code>/{MEDIA}/wwwroot/unsecure/user</code> directory of the Web server.</li> <li>● Pages with secure access must be copied to the <code>/{MEDIA}/wwwroot/secure/user</code> directory of the Web server.</li> </ul> <p>Where the value of {MEDIA} can be:</p> <ul style="list-style-type: none"> <li>● <b>FLASH</b>: save to the internal memory of the module.</li> <li>● <b>USB</b>: save to the USB memory key.</li> <li>● <b>CF</b>: save to the CF card.</li> </ul>
6	Copy the relevant files from your local disk to the desired location in the remote server (module).

**NOTE:** You can manage and transfer custom Web pages using Web Designer for TSX ETG 30\*\*, in that case the transfer of the pages is managed by the software.

## Uploading a Custom Home Page

### Presentation

To replace the default home page with a page of your choice, follow the steps below:

- Save the initial configuration so that you can restore it later if necessary.
- Create your home page.
- Move the default home page to a secure location.
- Replace it with your own home page.
- Upload your home page to the embedded server.

### Saving the Initial Configuration

Before modifying the default home page, you should save the configuration. In this way, if an anomaly occurs on the embedded server, you can restore the initial configuration.

**NOTE:** The CD-ROM contains a backup copy of the default site under *ETG30\wwwroot*.

### Creating your Home Page

You must give your home page the same name as the default home page: `index.htm`

### Placing your Home Page

Copy your home page to the *wwwroot* directory containing the default home page.

### Uploading your Home Page

Step	Action
1	Save your project.
2	Open an FTP client (e.g.: filezilla).
3	Enter the IP address, the user name and the password to connect to the site (default: <b>userdata/userdata</b> ).
4	Find your Web project in the tree structure on your local disk.

Step	Action
5	<p>Depending on the type of pages you want to add, the location of the pages on the module can be different:</p> <ul style="list-style-type: none"><li>● Pages accessible to all must be copied to the <code>/{MEDIA}/wwwroot/unsecure/user</code> directory of the Web server.</li><li>● Pages with secure access must be copied to the <code>/{MEDIA}/wwwroot/secure/user</code> directory of the Web server.</li></ul> <p>Where the value of {MEDIA} can be:</p> <ul style="list-style-type: none"><li>● FLASH: save to the internal memory of the module.</li><li>● USB: save to the USB memory key.</li><li>● CF: save to the CF card.</li></ul>
6	Copy the relevant files from the local site to the desired location in the remote site.

---

## 8.2 Creating Custom Pages with an HTML Editor

---

### Overview

This section describes how to integrate custom pages for the TSX ETG 30•• module with an HTML editor.

### What's in this Section?

This section contains the following topics:

Topic	Page
Inserting Applets into a Web Page	478
Inserting a LiveBeanApplet	479
Inserting a LiveBeanApplet Using HTML Code	481
Inserting a LiveLabelApplet Using HTML Code	484

## Inserting Applets into a Web Page

### Presentation

FactoryCast includes graphic objects (or Java beans) which are used to view the devices' dynamic data in a graphic display. Generally, the final constructs a graphic display using the **Graphic Editor** tool. You can however construct a graphic display by using the graphic objects inserted into the Web page using the *LiveBeanApplet*. Any graphic object, such as an analog dial or a push button, can be inserted into a Web page in such a way that the dynamic data may be displayed outside the Graphic Editor.

To display the devices' dynamic data using a browser, you can choose one of the two methods of inserting a FactoryCast applet into a Web page.

- (1) Enter the HTML code which is given in sections:
  - Inserting a *LiveBeanApplet* Using HTML Code
  - Inserting a *LiveBeanApplet* Using HTML Code
- (2) Insert a Java applet and then complete the dialog boxes using the FactoryCast extension for Microsoft FrontPage 2000 (*see page 489*).

## Inserting a LiveBeanApplet

### Presentation

This section covers general concepts relating to inserting a *LiveBeanApplet*.

**NOTE:**

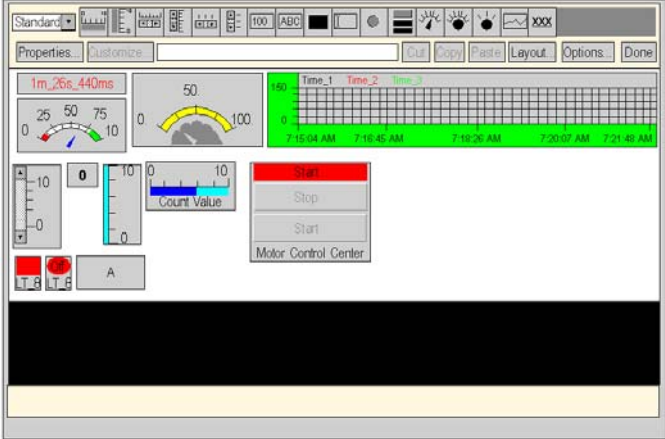
In this guide you will find information about inserting a *LiveBeanApplet* using:

- HTML code (see page 481),
- The Microsoft FrontPage extension (see page 496).

**NOTE:** Before inserting a *LiveBeanApplet* into a Web page, you must create a JavaBeans library using the Graphic Editor included in the Web Designer for TSX ETG 30••. In general, a user will create a JavaBeans library that contains an instance of each object he wants to use in a Web page. Consider this library as a set of templates, copied and customized in your Web pages. A library could, for example, contain an analog meter, a rotary selector and a push button. It would then be possible to add several instances of each bean to a Web page, each with a unique set of parameters, such as an address.

### Inserting a LiveBeanApplet

Step	Action
1	<p>Using your Web browser, enter the name or the IP address of your Web server module and access the Graphic Editor applet via the <b>Monitoring</b> link.</p> 

Step	Action
2	<p>Go into <b>Edit</b> mode and create the JavaBeans library to use in your Web pages. Customize each bean using a common set of parameters, so that only unique parameters are configured, such as the register address.</p>  <p><b>Note:</b> for further information on the use of the Graphic Editor applet, refer to the <b>Graphic Editor</b> section.</p>
3	<p>Once you have created your library, click <b>Done</b> in the editor window and save your library.</p>



## Inserting a LiveBeanApplet Using HTML Code

### Presentation

To graphically display data, use graphic objects such as the analog dial or the push button. Before inserting an applet into a Web page, a special applet called *LiveBeanMgrApplet* must be installed on the server.

### LiveBeanMgrApplet

The Web page can display the devices' dynamic data using the *LiveBeanMgrApplet*. This applet **must** feature **once** on the page if it already contains instances of *LiveBeanApplet*.

The *LiveBeanMgrApplet* applet can be included in a Web page in two different ways.

- Invisible applet — if the Web page is only used to check the device values, no input is required by the user.
- Key icon — if the Web page is used both to send new values to the devices and to check these values, the user must enter elements in such a way as to be able to send new values.

**NOTE:** If the applet is used as a key icon, the user must enter a password to be able to send data values to the devices. When the user clicks the applet (key icon) in the Web browser, a dialogue box invites him/her to enter a password which will allow him/her to write to the devices.

The HTML code which enables the applet to be included in a Web page used purely for monitoring purposes is as follows:

```
<APPLET codebase="/"
archive="classes/SAComm.jar,classes/GDE.jar,classes/Widgets.
jar,images.zip"
code="com.schneiderautomation.gde.LiveBeanMgrApplet"
width=0 height=0>
<PARAM name="FTP_ROOT" value="/NAND/FLASH1">
</APPLET>
```

The HTML code which enables the applet to be included in a Web page used both to send values to the devices and for monitoring purposes is as follows:

```
<APPLET codebase="/"
archive="classes/SAComm.jar,classes/GDE.jar,classes/Widgets.
jar,images.zip"
code="com.schneiderautomation.gde.LiveBeanMgrApplet"
width=32 height=32>
<PARAM name="FTP_ROOT" value="/NAND/FLASH1">
<PARAM name=MODE value="READWRITE">
<PARAM name=AUTO_LOGIN value="FALSE">
</APPLET>
```

In the example given above, if the value of the **AUTO\_LOGIN** parameter is set to **TRUE** as opposed to **FALSE**, the password is entered automatically (the user does not need to enter it).

## LiveBeanApplet

The *LiveLabelApplet* applet is included once for each symbol (variable) or direct address monitored/checked on the Web page. For example, if you monitor three symbols (variables), you must include the applet three times. Using the *LiveBeanApplet* applet, all graphic objects/Java beans created using the **Graphic Editor** (see page 422) can be included in a Web page as separate applet. All graphic objects saved as part of a **Graphic Editor** graphic display can be retrieved in the graphic file and presented by the applet.

## LiveBeanApplet Parameters

The *LiveBeanApplet* applet uses parameters which allow you to specify the graphic object presented by the applet and to define the applet's background color.

Parameter	Definition
LIBRARY	The name of the graphic display containing the graphic object which must be presented by the applet. (This name is the same as that used when the graphic display is saved with the <b>Graphic Editor</b> ). <b>This parameter is mandatory.</b>
BEAN	The name of the graphic object which must be retrieved in the graphic display specified by the LIBRARY parameter. (The name will be that which is displayed as the graphic object's 'Name' property). <b>This parameter is mandatory.</b>
BACKGRND	Applet background color. The authorized values are as follows: WHITE, LT_GRAY, GRAY, DK_GRAY, BLACK, RED, PINK, ORANGE, YELLOW, GREEN, MAGENTA, CYAN and BLUE. It is also possible to use an RGB value in the "0xRRGGBB" format, where RR, GG and BB are the hexadecimal values of the red, green and blue components respectively. This parameter is optional but is usually set to correspond to the color of the HTML page.

In addition to the parameters above, the *LiveBeanApplet* <APPLET> must mention the **width** and **height** attributes. A *LiveBeanApplet* applet must usually be the same size as the graphic object it presents. To know the size of a graphic object, select it whilst the **Graphic Editor** is in Edit mode. The name and size of the selected object are specified in the **Information Area** which can be found at the top of the **Graphic Editor** applet.

## Example of LiveBeanApplet

All instances of the *LiveBeanApplet* applet featuring in a Web page follow the same template; only the applet parameters and size vary.

This is the HTML code which allows the inclusion of a *LiveBeanApplet* presenting the **MyDial** graphic object, saved by the **Graphic Editor** as part of the **Library1** graphic display.

```
<APPLET codebase="/"
archive="classes/SAComm.jar,classes/GDE.jar,classes/Widgets.
jar,images.zip"
code="com.schneiderautomation.gde.LiveBeanApplet"
width=180 height=160>
<PARAM name=LIBRARY value="Library1">
<PARAM name=BEAN value="MyDial">
<PARAM name=BACKGRND value="0xDDEEFF">
</APPLET>
```

## Inserting a LiveLabelApplet Using HTML Code

### Presentation

Before inserting an interactive label into a Web page, you must first insert a special applet called *LiveLabelMgrApplet* into the page.

**NOTE:** However, if a web page contains both a *LiveLabelApplet* and a *LiveBeanApplet*, it must only contain a single instance of *LiveBeanMgrApplet* and not *LiveLabelMgrApplet*.

(*LiveBeanMgrApplet* handles *LiveLabelApplet* and *LiveBeanApplet*, while *LiveLabelMgrApplet* only handles *LiveLabelApplet*.)

### LiveLabelMgrApplet

Using the *LiveLabelMgrApplet*, the Web page can display dynamic data from the PLC. You **must** include this applet **once** in the page if it already contains instances of *LiveLabelApplet*.

This is the HTML code to use to insert the applet into a page.

```
<APPLET codebase=" / "
archive="classes/SAComm.jar,classes/GDE.jar,classes/Widgets.
jar,images.zip"
code="com.schneiderautomation.factorycast.LiveLabelMgrApple"
width=0 height=0>
<PARAM name="FTP_ROOT" value="/NAND/FLASH1">
</APPLET>
```

### LiveLabelApplet

Use a *LiveLabelApplet* for each symbol (variable) or direct address displayed on the Web page being used. For example, if you display three symbols (variables), you must include the applet three times.

Field	Description
Label	Label of your variable
Value	Execution value of the variable
Units	Units that you specify for the value

### Data Parameters

Parameter	Definition	With the default value of
LABEL	A text label to identify the data element	No label
UNITS	A text label to identify the physical units of the value	No unit displayed

Parameter	Definition	With the default value of	
UNITID	Address of the Modbus slave (not used for symbols).		
ADDRESS	Direct Modbus address or symbol name.	None	
DATATYPE	The data type of the direct Modbus address. Valid values for this parameter are:	REGISTER	
	COIL		output bit (Boolean)
	DISCRETE INPUT		input bit (Boolean)
	REGISTER		16-bit signed integer
	INT32		32-bit signed integer
	INT32SWAP		32-bit signed integer with least significant and most significant words inverted
	INPUT REGISTER		16-bit signed integer for analog input
FORMAT	Display format of the value. Valid values for this parameter are:	DEC for most data types BOOL for the coil or discrete input data type	
	DEC		decimal
	HEX		hexadecimal
	BIN		binary
	ASCII		bytes displayed as ASCII characters
	BOOL		ON_WORD or OFF_WORD (see below)
GAIN	The gain (multiplier) used for the scale of the value retrieved in physical units.	1.0	
	<b>NOTE:</b> It is only possible to use the scale if GAIN or BIAS is set and if the FORMAT is DEC. The linear scale follows the formula: $SCALED\_VALUE = GAIN \times RAW\_VALUE + BIAS$		
BIAS	The bias (offset) used for the scale of the value retrieved in physical units. See NOTE under GAIN	0.0	
ON_WORD	Text value to be displayed when the value is not zero (To be used only if the FORMAT is BOOL).	ON	
OFF_WORD	Text value to be displayed when the value is zero (To be used only if the FORMAT is BOOL).	OFF	

Parameter	Definition	With the default value of
FOREGRND	Color of the applet foreground Valid values are: WHITE, LT_GRAY, DK_GRAY, BLACK, RED, PINK, ORANGE, YELLOW, GREEN, MAGENTA, CYAN, and BLUE It is also possible to use an RGB value in the format "0xRRGGBB" where RR, GG and BB are the hexadecimal values of the red, green and blue components respectively.	BLACK
BACKGRND	Color of the applet background For the valid values, see FOREGRND.	LT_GRAY
ERROR_COLOR	Color of the foreground of the VALUE field if the value cannot be retrieved from the PLC. For the valid values, see FOREGRND.	MAGENTA
LABEL_ALIGN	Alignment of text in the LABEL field if the size of the field is greater than the length of the text Valid values are: LEFT, CENTER and RIGHT.	LEFT
VALUE_ALIGN	Alignment of text in the VALUE field if the size of the field is greater than the length of the text Valid values are: LEFT, CENTER and RIGHT.	LEFT
UNITS_ALIGN	Alignment of text in the UNITS field if the size of the field is greater than the length of the text Valid values are: LEFT, CENTER and RIGHT.	LEFT
FONT_NAME	The name of the font used by the applet Valid values are: SERIF, SANSSERIF and MONOSPACE.	SANSSERIF
FONT_BOLD	If this parameter is set, all text in the applet will be displayed in bold Valid values are: TRUE and FALSE.	FALSE
FONT_ITALIC	If this parameter is set, all text in the applet will be displayed in italics Valid values are: TRUE and FALSE.	FALSE
FONT_SIZE	The size of the font used by the applet	12
LABEL_WIDTH	The width of the LABEL field.	
UNITS_WIDTH	The width of the UNITS field.	

## Size Parameters

The size of a *LiveLabelApplet* is specified in the width and height attributes of its <APPLET> label. The LABEL and UNITS fields of the applet will always have the necessary sizes to display the text values of the associated applet parameters, unless the width of the LABEL or UNITS field is configured using the LABEL\_WIDTH or UNITS\_WIDTH parameters. The remaining width of the applet is allocated to the associated VALUE field.

### LiveLabelApplet, Example 1

The applet example in this section contains almost all the applet parameters. This is the HTML code of the example.

```
<APPLET codebase="/"
archive="classes/SAComm.jar,classes/GDE.jar,classes/Widgets.
jar,images.zip"
code="com.schneiderautomation.factorycast.LiveLabelApplet"
width=300 height=30>
<PARAM name=LABEL value="Reactor 1 Temperature">
<PARAM name=UNITS value="F">
<PARAM name=UNITID value="1">
<PARAM name=ADDRESS value="100">
<PARAM name=DATATYPE value="REGISTER">
<PARAM name=FORMAT value="DEC">
<PARAM name=GAIN value="2.0">
<PARAM name=BIAS value="100.0">
<PARAM name=FOREGRND value="WHITE">
<PARAM name=BACKGRND value="BLACK">
<PARAM name=ERROR_COLOR value="RED">
<PARAM name=FONT_NAME value="SERIF">
<PARAM name=FONT_BOLD value="TRUE">
<PARAM name=FONT_ITALIC value="FALSE">
<PARAM name=FONT_SIZE value="10">
</APPLET>
```

## LiveLabelApplet, Example 2

This minimal applet example uses the default values for most of the parameters. This is the HTML code of the example.

```
<APPLET codebase="/"
archive="classes/SAComm.jar,classes/GDE.jar,classes/Widgets.jar,images.zip"
code="com.schneiderautomation.factorycast.LiveLabelApplet"
width=300 height=30>
<PARAM name=LABEL value="Reactor 1 Pressure">
<PARAM name=UNITS value="PSI">
<PARAM name=ADDRESS value="Symbol1">
</APPLET>
```



---

## 8.3 Creating Custom Pages with FrontPage

---

### Overview

This section describes how to create custom pages for the TSX ETG 30\*\* module with FrontPage.

### What's in this Section?

This section contains the following topics:

Topic	Page
Installing a Microsoft FrontPage Extension	490
Inserting a LiveBeanApplet with FrontPage	496
Inserting LiveLabelApplet Using FrontPage	504
Uploading Custom Pages with FrontPage	508

## Installing a Microsoft FrontPage Extension

### Presentation

This section describes how to install/remove the Web Designer for TSX ETG 30•• extension for Microsoft FrontPage 2000.

### Installing FactoryCast

During the installation of Web Designer for TSX ETG 30••, if FrontPage 2000 has been installed on the same PC, then the Web Designer Extension for FrontPage 2000 is automatically installed as a FrontPage "Macro File." If this is the case, then proceed to the next section, which provides instructions for adding the Extension to FrontPage's menu.

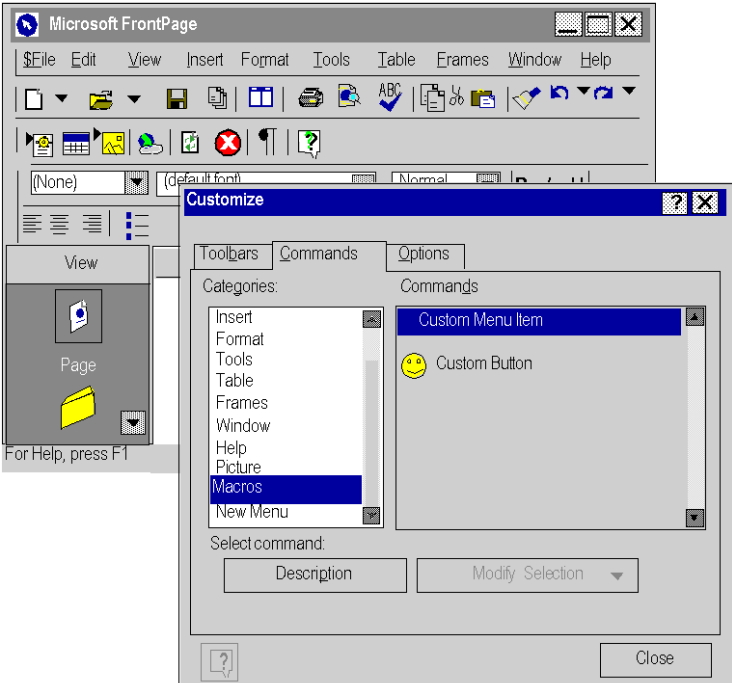
However, if FrontPage 2000 is installed after Web Designer for TSX ETG 30•• has been installed, then the Extension can be installed by either re-installing Web Designer for TSX ETG 30•• or manually copying the macro file to FrontPage's macro folder. For manual installation, after installing FrontPage 2000, the following will install the Extension:

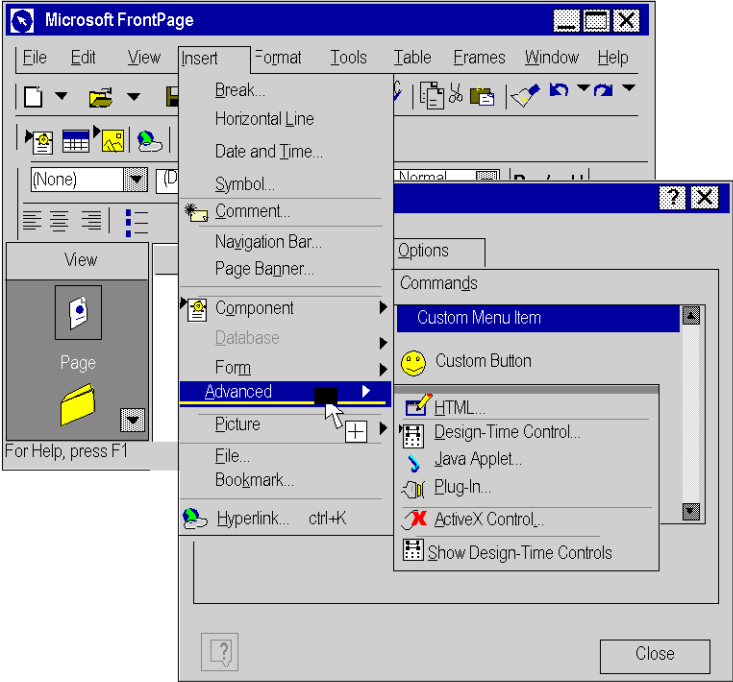
Copy the file *Microsoft FrontPage.fpm* from CD-ROM to the folder *%USERPROFILE%\Application Data\Microsoft\FrontPage\Macros* (create the final 'Macros' folder, if it does not already exist).

### Adding Extension

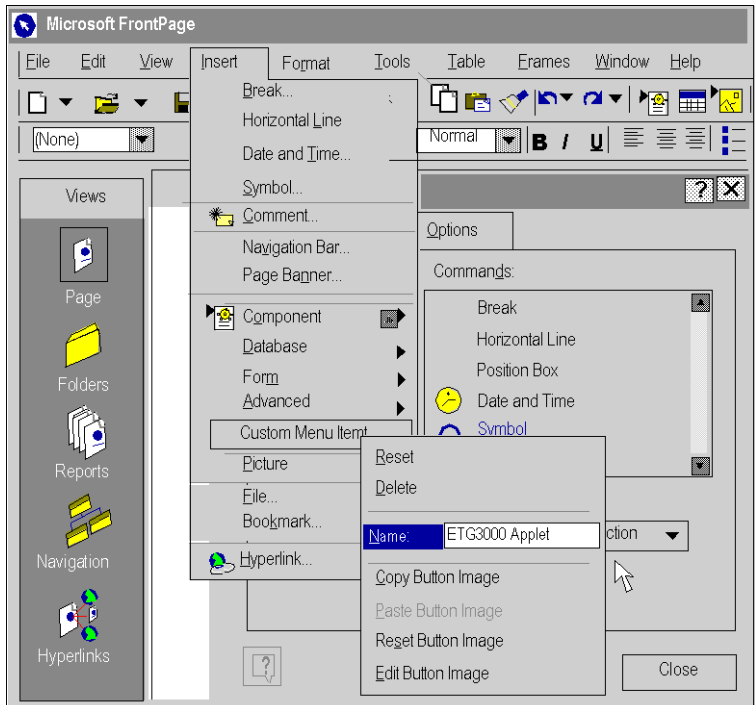
To add the Web Designer or the Modbus extension to the FrontPage **Insert** menu, do the following.

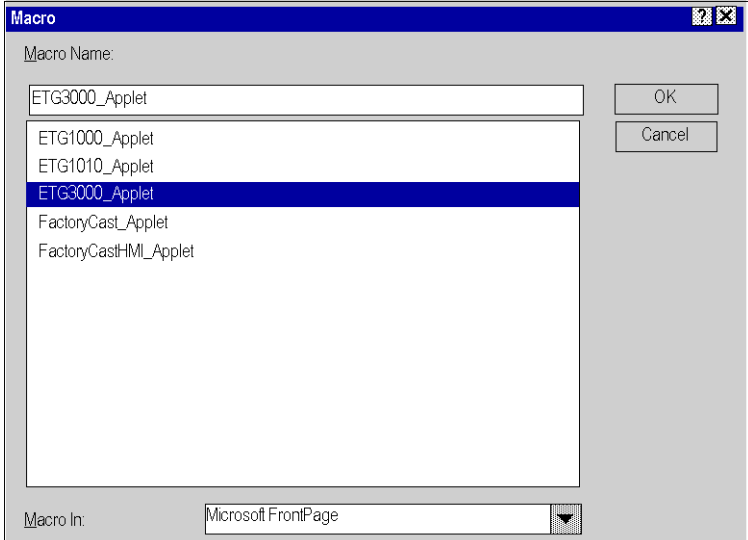
Step	Action
1	Start FrontPage 2000.
2	Click <b>Customize</b> on the <b>Tools</b> menu.

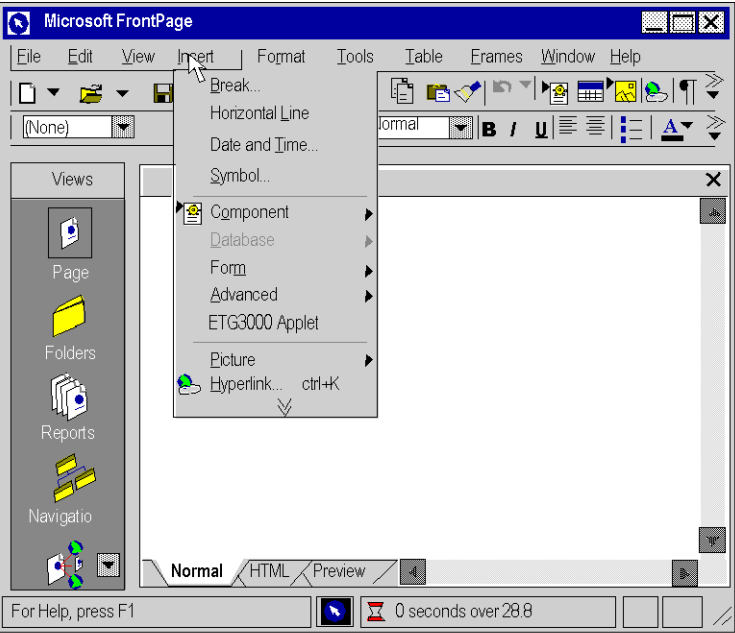
Step	Action
3	<p>Click the <b>Commands</b> tab, and then select <b>Macros</b> from the <b>Categories</b> list.</p>  <p>The screenshot shows the Microsoft FrontPage application window. The 'Customize' dialog box is open, with the 'Commands' tab selected. In the 'Categories' list on the left, 'Macros' is highlighted. The 'Commands' list on the right contains two items: 'Custom Menu Item' (highlighted) and 'Custom Button' (with a yellow smiley face icon). Below the lists, there are buttons for 'Description' and 'Modify Selection'. The 'Close' button is at the bottom right of the dialog box.</p>

Step	Action
4	<p>Drag and drop the entry <b>Custom Menu Item</b> from the <b>Commands</b> list to the <b>Insert</b> menu, beneath the <b>Advanced</b> command. (The <b>Insert</b> menu will automatically drop down when you drag over it.)</p>  <p>The screenshot shows the Microsoft FrontPage application window. The 'Insert' menu is open, and the 'Advanced' command is selected. The 'Commands' list is also open, showing 'Custom Menu Item' selected. The 'Advanced' command in the Insert menu has a right-pointing arrow, and the 'Custom Menu Item' in the Commands list is highlighted in blue. A mouse cursor is positioned over the 'Advanced' command in the Insert menu. The 'Commands' list is a scrollable list with a 'Close' button at the bottom right.</p>

Step	Action
5	<p>Click <b>Modify Selection</b>, change the name to <b>ETG 3000 Applet</b>, and then press ENTER key.</p>



Step	Action
6	<p>Click <b>Modify Selection</b> again, and then choose <b>Assign Macro</b>.</p>  <p>The screenshot shows a dialog box titled "Macro". At the top, there is a "Macro Name:" label followed by a text input field containing "ETG3000_Applet". Below this is a list box containing the following items: "ETG1000_Applet", "ETG1010_Applet", "ETG3000_Applet" (which is highlighted with a blue background), "FactoryCast_Applet", and "FactoryCast-HML_Applet". To the right of the list box are two buttons: "OK" and "Cancel". At the bottom of the dialog, there is a "Macro In:" label followed by a dropdown menu currently showing "Microsoft FrontPage".</p>
7	<p>Choose <b>ETG3000_Applet</b> from the list, and then click <b>OK</b>.</p>

Step	Action
8	<p>Click <b>Close</b>. The <b>ETG3000 Applet</b> command is added to the <b>Insert</b> menu.</p>  <p>The screenshot shows the Microsoft FrontPage application window. The 'Insert' menu is open, displaying options such as 'Break...', 'Horizontal Line', 'Date and Time...', 'Symbol...', 'Component', 'Database', 'Form', 'Advanced', 'ETG3000 Applet', 'Picture', and 'Hyperlink... ctrl+K'. The 'ETG3000 Applet' option is highlighted. The interface includes a menu bar (File, Edit, View, Insert, Format, Tools, Table, Frames, Window, Help), a toolbar, a 'Views' sidebar with icons for Page, Folders, Reports, and Navigation, and a status bar at the bottom showing 'Normal', 'HTML', 'Preview', and a timer '0 seconds over 28.8'.</p>

### Removing FactoryCast Extension

Step	Action
1	In FrontPage, click <b>Customize</b> on the <b>Tools</b> menu.
2	Click the <b>Insert</b> menu, and then select <b>ETG3000 Applet</b> .
3	Right click and select <b>Delete</b> from the popup menu.

### Editing Applets

There are two ways to edit an applet that has been inserted into your Web page. First, you can double-click the object and make changes via dialog boxes. Or you can switch to the HTML editor in FrontPage and do your editing in this environment. It is suggested you edit via the first approach unless you are comfortable programming in the HTML language used to build Web pages.

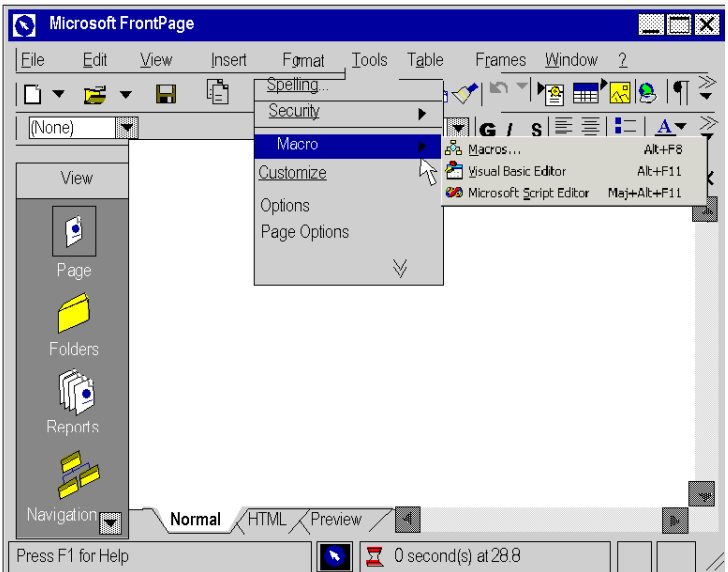
## Inserting a LiveBeanApplet with FrontPage

### Presentation

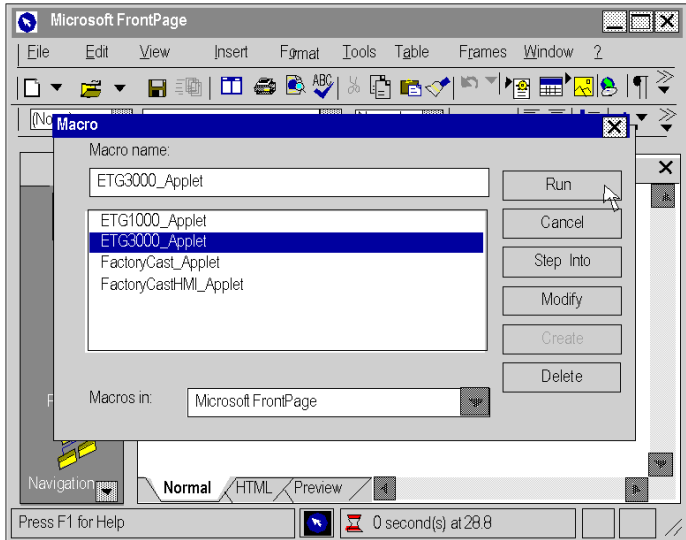
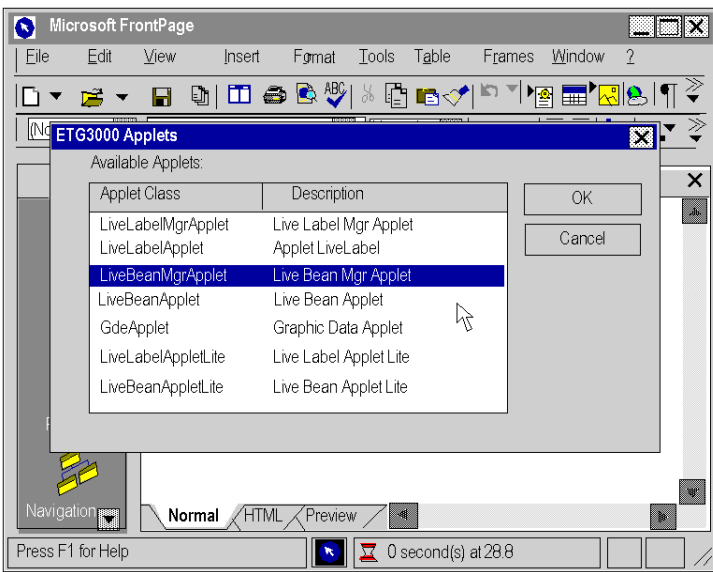
This section describes how to insert a *LiveBeanApplet* into a Web page. Before proceeding with the insertion, read the following note then carry out the 14 operations described below.

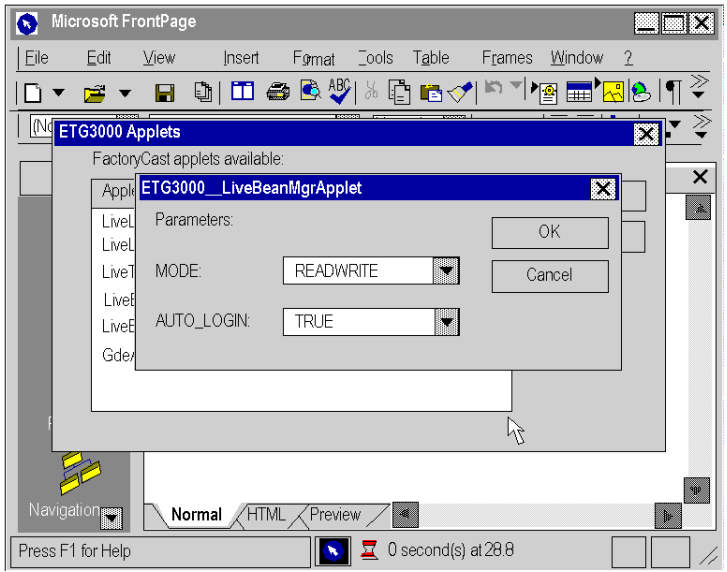
**NOTE:** Prior to inserting a *LiveBeanApplet* into a Web page, you must create a JavaBeans library using the Graphic Editor. Generally a user will create a JavaBeans library that has one instance of every object that they would like to use in a Web page. Think of this library as a set of templates that are copied to and customized for your Web pages. For example a library may have one analog meter, one rotary selector, and one push button. Multiple instances of each bean can then be added to a Web page, each with a set of unique parameters such as an address.

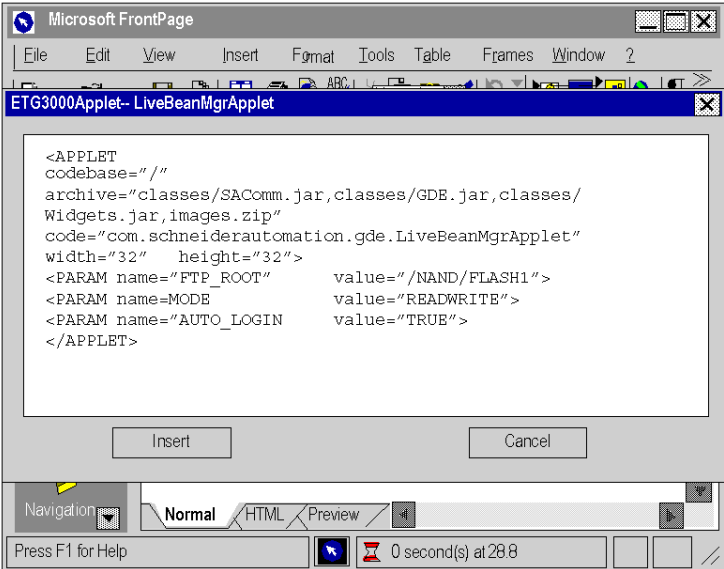
### Inserting a LiveBeanApplet

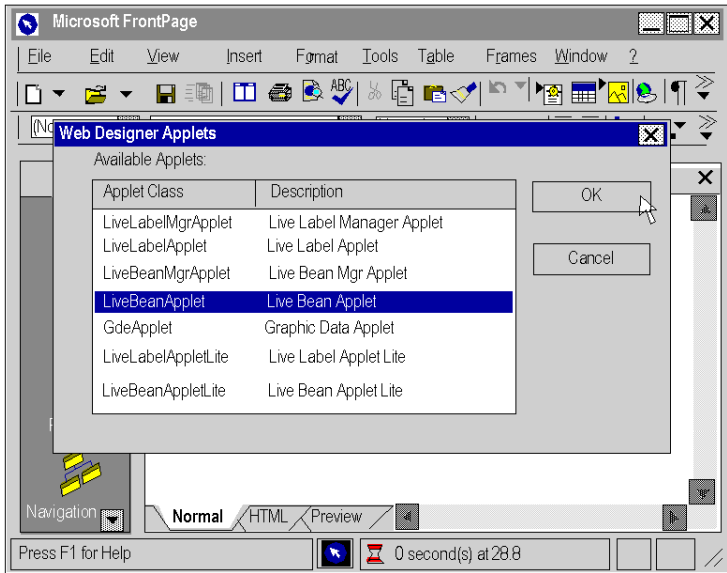
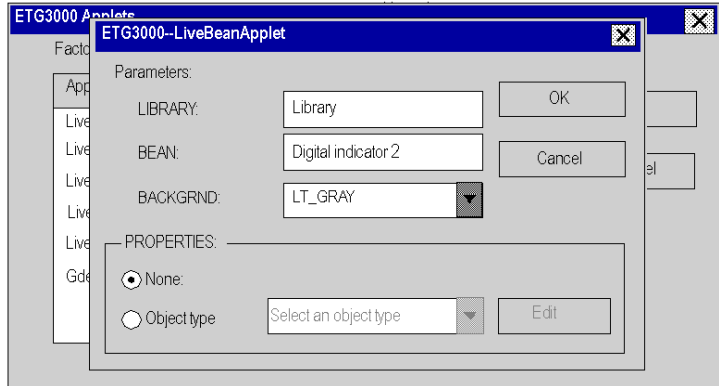
Step	Action
1	<p>Select <b>Tools   Macro</b>, then to display the applet selection window click <b>Macros</b>.</p>  <p>The screenshot shows the Microsoft FrontPage application window. The menu bar includes File, Edit, View, Insert, Format, Tools, Table, Frames, and Window. The Tools menu is open, and the 'Macro' option is highlighted. Other options in the Tools menu include Spelling..., Security, Customize, Options, and Page Options. The status bar at the bottom indicates 'Press F1 for Help' and a timer showing '0 second(s) at 28.8'.</p>

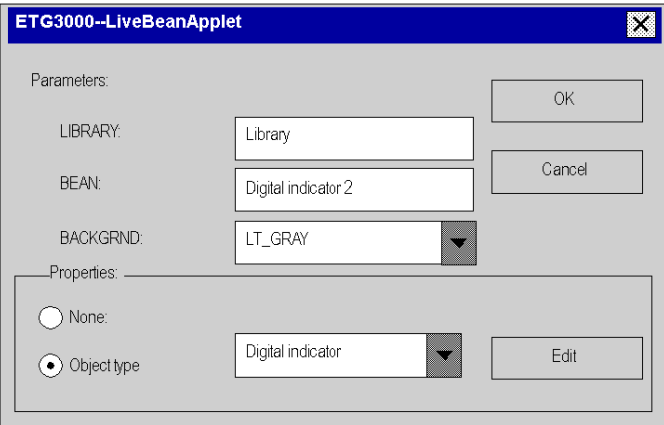
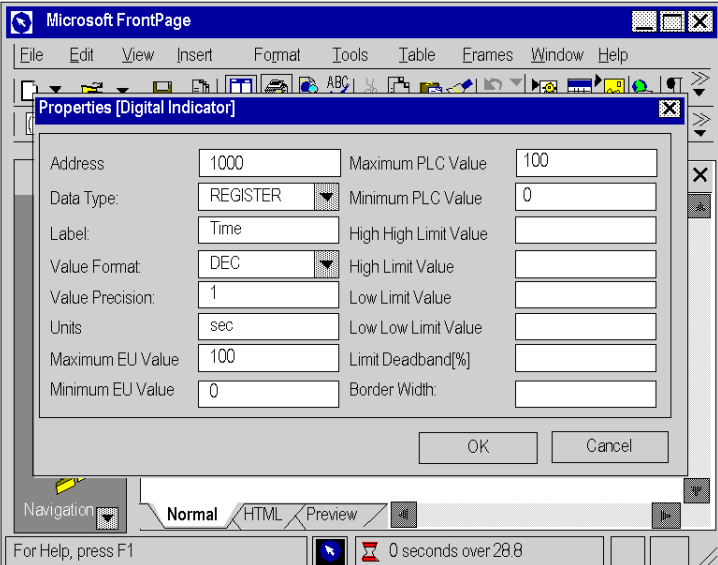


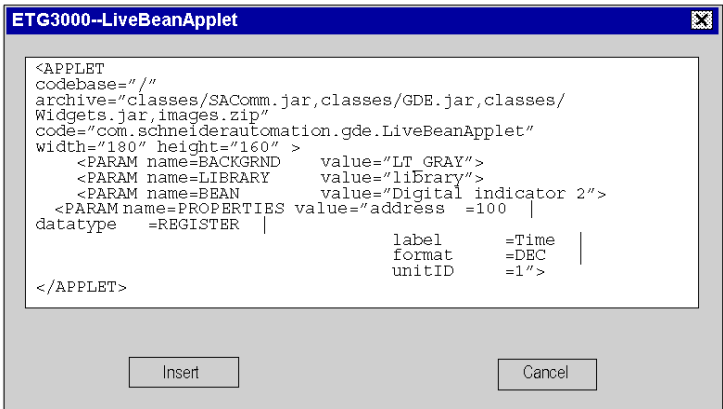
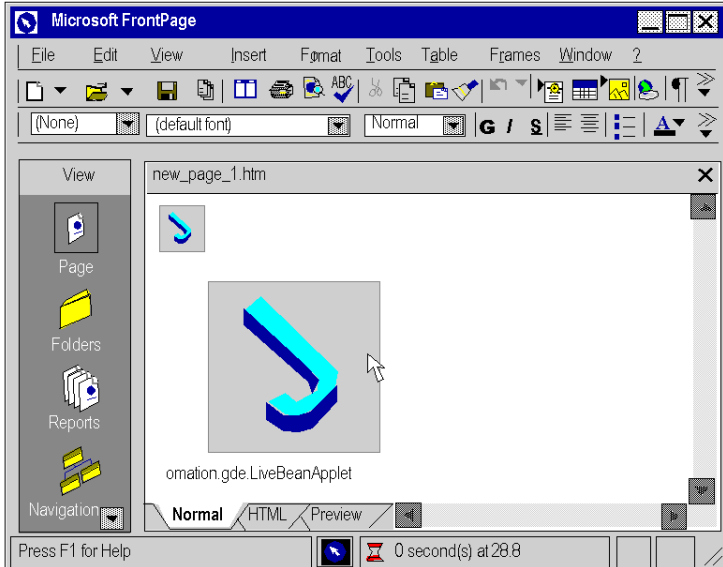
Step	Action																
2	<p>Select the <b>ETG3000_Applet</b> applet, then click <b>Run</b>.</p>  <p>The screenshot shows the Microsoft FrontPage interface with a 'Macro' dialog box open. The 'Macro name' field is filled with 'ETG3000_Applet'. Below it, a list of macros is shown, with 'ETG3000_Applet' selected. To the right of the list are buttons for 'Run', 'Cancel', 'Step Into', 'Modify', 'Create', and 'Delete'. The 'Run' button is highlighted with a mouse cursor. The background shows the FrontPage menu bar and toolbar.</p>																
3	<p>Select the <b>LiveBeanMgrApplet</b> applet , then click <b>OK</b>.</p>  <p>The screenshot shows the Microsoft FrontPage interface with an 'ETG3000 Applets' dialog box open. It features a table titled 'Available Applets' with two columns: 'Applet Class' and 'Description'. The 'LiveBeanMgrApplet' row is highlighted. To the right of the table are 'OK' and 'Cancel' buttons. The background shows the FrontPage menu bar and toolbar.</p> <table border="1" data-bbox="562 1015 980 1242"> <thead> <tr> <th>Applet Class</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>LiveLabelMgrApplet</td> <td>Live Label Mgr Applet</td> </tr> <tr> <td>LiveLabelApplet</td> <td>Applet LiveLabel</td> </tr> <tr> <td><b>LiveBeanMgrApplet</b></td> <td><b>Live Bean Mgr Applet</b></td> </tr> <tr> <td>LiveBeanApplet</td> <td>Live Bean Applet</td> </tr> <tr> <td>GdeApplet</td> <td>Graphic Data Applet</td> </tr> <tr> <td>LiveLabelAppletLite</td> <td>Live Label Applet Lite</td> </tr> <tr> <td>LiveBeanAppletLite</td> <td>Live Bean Applet Lite</td> </tr> </tbody> </table> <p><b>Note:</b> The <i>LiveBeanMgrApplet</i> applet is a special applet, which should only be inserted into your Web page once.</p>	Applet Class	Description	LiveLabelMgrApplet	Live Label Mgr Applet	LiveLabelApplet	Applet LiveLabel	<b>LiveBeanMgrApplet</b>	<b>Live Bean Mgr Applet</b>	LiveBeanApplet	Live Bean Applet	GdeApplet	Graphic Data Applet	LiveLabelAppletLite	Live Label Applet Lite	LiveBeanAppletLite	Live Bean Applet Lite
Applet Class	Description																
LiveLabelMgrApplet	Live Label Mgr Applet																
LiveLabelApplet	Applet LiveLabel																
<b>LiveBeanMgrApplet</b>	<b>Live Bean Mgr Applet</b>																
LiveBeanApplet	Live Bean Applet																
GdeApplet	Graphic Data Applet																
LiveLabelAppletLite	Live Label Applet Lite																
LiveBeanAppletLite	Live Bean Applet Lite																

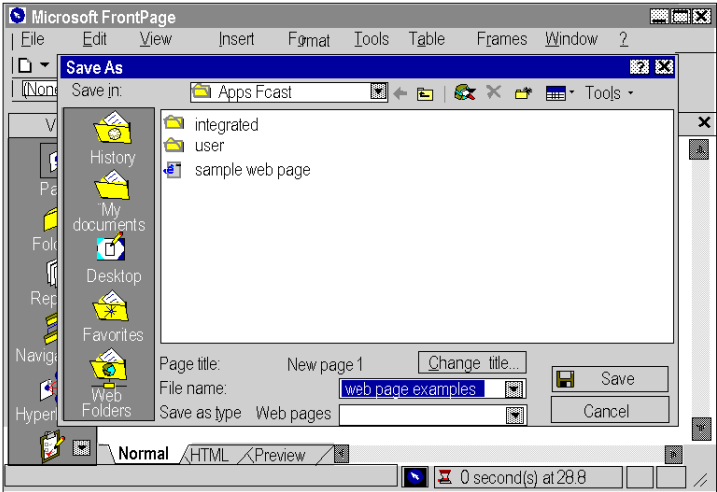
Step	Action
4	<p>Configure the <b>Mode</b> and <b>Auto-Login</b> parameters, then click <b>OK</b>.</p>  <p>The screenshot shows the Microsoft FrontPage application window. A dialog box titled 'ETG3000 Applets' is open, displaying a list of available applets. The 'ETG3000 LiveBeanMgrApplet' is selected. A smaller configuration dialog is overlaid on top, showing parameters for the selected applet. The 'MODE' parameter is set to 'READWRITE' and the 'AUTO_LOGIN' parameter is set to 'TRUE'. The dialog includes 'OK' and 'Cancel' buttons. The background shows the FrontPage interface with a menu bar (File, Edit, View, Insert, Format, Tools, Table, Frames, Window) and a toolbar. The status bar at the bottom indicates 'Press F1 for Help' and a timer showing '0 second(s) at 28.8'.</p>

Step	Action
5	<p data-bbox="473 201 1215 250">A window opens, showing the Java code which will be inserted in your HTML document.</p>  <p data-bbox="473 837 952 862">Click <b>Insert</b> to complete the insertion of the applet.</p> <p data-bbox="473 865 1221 1018"><b>Note:</b> Although a gray box containing a blue letter "J" appears on your Web page while editing, the applet will only be displayed on your downloaded Web page if you have set the Mode parameter to <b>ReadWrite</b>. It will then appear in the form of a Key when you view it using your browser. On the other hand, if the <b>Mode</b> is set to <b>ReadOnly</b>, the width and height of the applet will be set to zero and will not be visible in your browser.</p>

Step	Action
6	<p>Select the <i>LiveBeanApplet</i> in the Web Designer applet selection window, then click <b>OK</b>.</p> 
7	<p>Enter the name of a library and the name of the "bean" you wish to display in it. If you do not need to customize the bean properties (the address, for example), click <b>OK</b>.</p> 

Step	Action
8	<p>In general, you will at least need to customize the Address property of your beans. Once you have entered the names of the library and bean, select the object type in the <b>Object Type</b> box.</p>  <p>Click <b>Edit</b> when you have finished.</p>
9	<p>In the edit window, only change parameters that are specific to this bean, such as the address. All other parameters will be set to the same values as for the bean saved in your library. Click <b>OK</b> when you have finished.</p> 

Step	Action
10	<p>Click <b>OK</b>. A window opens, showing the Java code which will be inserted in your HTML document. Click <b>Insert</b> to complete the insertion of the applet.</p>  <pre> &lt;APPLET codebase="/" archive="classes/SAComm.jar,classes/GDE.jar,classes/ Widgets.jar,images.zip" code="com.schneiderautomation.gde.LiveBeanApplet" width="180" height="180" &gt;   &lt;PARAM name=BACKGRND value="LT_GRAY"&gt;   &lt;PARAM name=LIBRARY value="library"&gt;   &lt;PARAM name=BEAN value="Digital indicator 2"&gt;   &lt;PARAM name=PROPERTIES value="address =100   datatype =REGISTER   label =Time   format =DEC   unitID =1"&gt; &lt;/APPLET&gt;                     </pre>
11	<p>Continue inserting instances of the <i>LiveBeanApplet</i> applet into your Web page. Once you have inserted the last applet, click <b>Cancel</b> in the Web Designer applet selection window to edit your Web page.</p> 

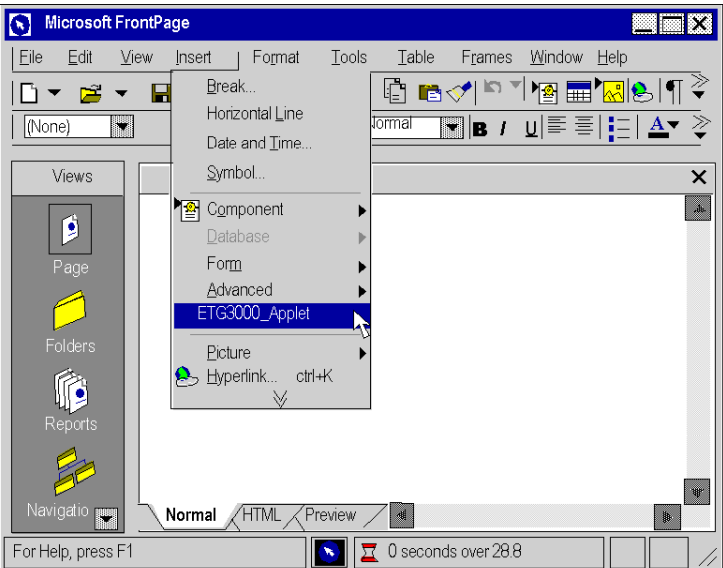
Step	Action
12	<p>Save your custom Web page.</p>  <p>The screenshot shows the Microsoft FrontPage application window with the 'Save As' dialog box open. The 'Save in' location is 'Apps Fcast'. The file list shows 'integrated', 'user', and 'sample web page'. The 'File name' field is 'web page examples' and the 'Save as type' is 'Web pages'. The 'Page title' is 'New page 1' and the 'File name' field is highlighted. The 'Save' button is visible.</p>
13	Transfer the custom Web page to the Web server.
14	Test the application.

## Inserting LiveLabelApplet Using FrontPage

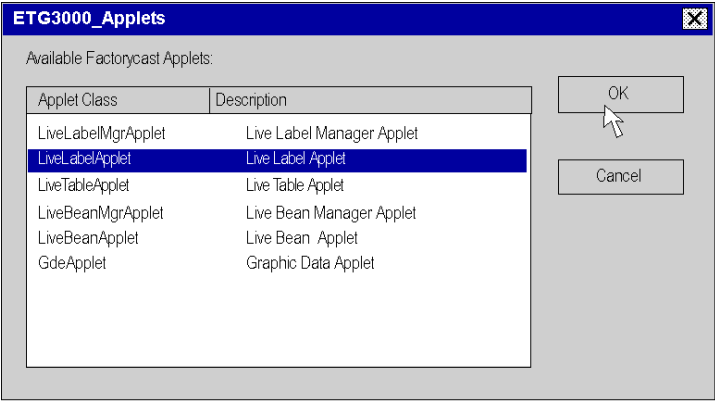
### Presentation

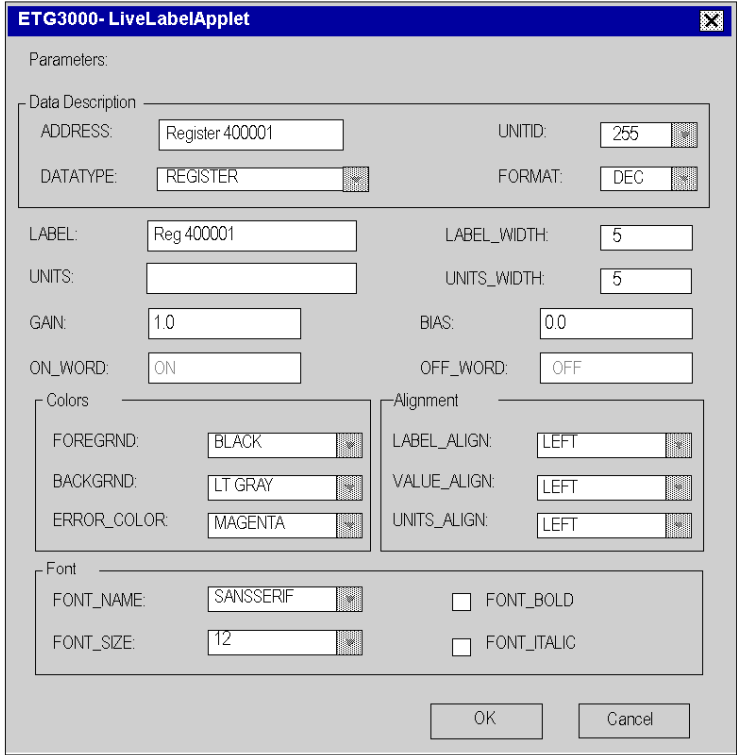
This section describes inserting *LiveLabelApplet* into a Web page.

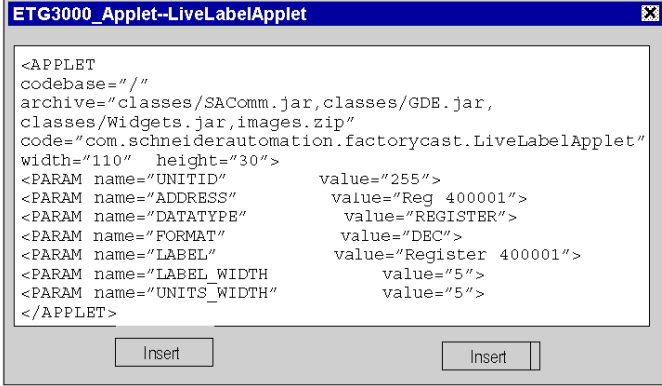
### Inserting LiveLabelApplet

Step	Action
1	<p>Select <b>Insert   ETG3000_Applet</b> to view the TSX ETG 3000 applet selection window.</p>  <p>The screenshot shows the Microsoft FrontPage application window. The 'Insert' menu is open, and the 'ETG3000_Applet' option is highlighted. Other visible options in the menu include 'Break...', 'Horizontal Line', 'Date and Time...', 'Symbol...', 'Component', 'Database', 'Form', 'Advanced', 'Picture', and 'Hyperlink... ctrl+K'. The 'Views' task pane on the left shows 'Page', 'Folders', 'Reports', and 'Navigation' options. The status bar at the bottom indicates 'For Help, press F1' and a timer showing '0 seconds over 28.8'.</p>



Step	Action
2	<p>Select the <i>LiveLabelMgrApplet</i> and then click <b>OK</b>.</p>  <p><b>Note:</b> The <i>LiveLabelMgrApplet</i> is a special applet that is inserted one time only into your Web page. Though a gray box with a blue letter "J" will appear on your Web page during editing, this applet will not be visible when you view your Web page from the Web server module.</p>
3	<p>Select the <i>LiveLabelApplet</i> and then click <b>OK</b>. The Parameters edit window appears.</p>

Step	Action
4	<p>Fill in the parameters for the label and then click <b>OK</b>.</p>  <p><b>Note:</b> For a description on each parameter, see <i>Inserting a LiveLabelApplet Using HTML Code</i>, page 484.</p>

Step	Action
5	<p>A window will appear showing the HTML code that will be inserted into your HTML document. Click <b>Insert</b>.</p>  <pre data-bbox="491 342 1126 610">&lt;APPLET codebase="" /" archive="classes/SAComm.jar,classes/GDE.jar, classes/Widgets.jar,images.zip" code="com.schneiderautomation.factorycast.LiveLabelApplet" width="110" height="30"&gt; &lt;PARAM name="UNITID" value="255"&gt; &lt;PARAM name="ADDRESS" value="Reg 400001"&gt; &lt;PARAM name="DATATYPE" value="REGISTER"&gt; &lt;PARAM name="FORMAT" value="DEC"&gt; &lt;PARAM name="LABEL" value="Register 400001"&gt; &lt;PARAM name="LABEL_WIDTH" value="5"&gt; &lt;PARAM name="UNITS_WIDTH" value="5"&gt; &lt;/APPLET&gt;</pre>
6	<p>Continue to add additional instances of <i>LiveLabelApplet</i> to your Web page. Once you have inserted your last applet, click <b>Cancel</b> on the applet selection window to edit your Web page.</p>

## Uploading Custom Pages with FrontPage

### Presentation

If you add custom web pages to the default website, you may choose to protect them with the same user name and the same password as those used for the default pages, or to authorize access to all users (no password required).

To add pages to the site, you must specify the folder in which the pages are stored before you can upload them to the server.

### Indicating the File Location

Place the folders containing your web pages and any associated graphics in a common directory. Each subdirectory should normally contain a page called `index.htm`.

All graphics file names must be in DOS 8.3 format (a maximum of eight characters for the name and three characters for the extension).

Security pages (with a password) should be copied to the `\wwwroot\secure\user` directory of the embedded server.

Pages accessible to all should be copied to the `\wwwroot\unsecure\user` directory of the embedded server.

**NOTE:** Do not forget to create new hyperlinks to other web pages to enable these pages to be accessed via the browser.

### Uploading to a Server

Step	Action
1	Save your project.
2	Select <b>File</b> → <b>Publish Web Site</b> in the menu. <b>Result:</b> the <b>Publish Web Site</b> dialog box opens with the FTP address to be incorporated. Use <code>ftp://module_ip_address/wwwroot/unsecure/user</code> for your free access pages. Use <code>ftp://module_ip_address/wwwroot/secure/user</code> for your password-secured pages.
3	Click on <b>Publish</b> to confirm the start of uploading. <b>Note:</b> A window will appear, asking for a name and password. The default value for both is <b>'wsupgrade'</b> .

---

## 8.4 Description of Graphic Objects

---

### Graphic Objects Library Lite

#### Presentation

This section describes the Graphic Objects library version Lite. This library enables you to download the graphic interface faster, particularly if you use a modem access.

It is a smaller version of the standard graphics library.

The graphic objects version Lite provided in the ETG3000\_Applet are designed to help you to create graphic displays imitating operator interface screens.

All the data control and monitoring objects have integrated communication functions and are designed as standalone graphic objects.

This section describes the standard graphic objects Lite and their properties:

- *LiveLabelAppletLite*: text object
- *LiveBeanAppletLite*: graphic objects library

**NOTE:** In the Graphic Objects Library Lite, you can monitor direct Modbus addresses or symbols coming from devices that use the Push Data mode (refer to the *Web Designer for TSX ETG 30\*\* User Manual* for more information). Other direct modbus addresses or symbols are not supported.

## LiveLabelAppletLite

This window gives the value of a direct address of a Modbus slave in a text field:

The screenshot shows a configuration window titled "ETG 3000-LiveLabelAppletLite". It is organized into several sections:

- Parameters:** A sub-section containing "Data Description" with fields for ADDRESS (5), UNITID (255), DATATYPE (REGISTER), and POLLING\_RATE (1000).
- Display Settings:** Fields for LABEL (label), LABEL\_WIDTH (25), UNITS (mm), and UNIT\_WIDTH (5).
- Scaling and Labels:** Fields for GAIN (1.0), BIAS (0.0), ON\_WORD (ON), and OFF\_WORD (OFF).
- Colors:** Dropdown menus for FOREGRND (BLACK), BACKGRND (LT GRAY), and ERROR\_COLOR (MAGENTA).
- Alignment:** Dropdown menus for LABEL\_ALIGN (LEFT), VALUE\_ALIGN (LEFT), and UNIT\_ALIGN (LEFT).
- Font:** Fields for FONT\_NAME (SANSSERIF) and FONT\_SIZE (12), along with checkboxes for FONT\_BOLD and FONT\_ITALIC.

At the bottom of the window are "OK" and "Cancel" buttons.

The properties of this window are as follows:

Property	HTML code	Description	Limits
Address	Address	Modbus address of the data type (1 to 65535) or symbol name.	See Note 1 (see page 529)
Data type	Data type	Modbus address data type (not used for symbols).	See Note 2 (see page 529)
Label	Label	Label to be displayed as part of the graphic object.	See Note 5 (see page 529)
Label Width	Label_Width	Width of label.	
Unit Width	Units_Width	Width of unit.	
Unit ID	UnitId	Address of the Modbus slave (not used for symbols)	1 to 255
Polling rate [ms]	Rate	Scanning value	
Gain	Gain	The gain (multiplier) is used for the scale of the value retrieved in physical units.	1.0
Bias	bias	The bias (offset) is used for the scale of the value retrieved in physical units.	0.0
On_Word	On_Word	Text value to be displayed when the value is not zero (use if the data type format is binary).	ON

Property	HTML code	Description	Limits
Off_Word	Off_Word	Text value to be displayed when the value is not zero (use if the data type format is binary).	OFF
Foregrnd	Foregrnd	Color of the applet foreground.	BLACK
Backgrnd	Backgrnd	Color of the applet background.	LT_GRAY
Error_Color	Error_Color	Color of the applet foreground if the address value cannot be retrieved.	MAGENTA
Label_Align	Label_Align	Alignment of text in the Label field if the size is greater than the length of the text.	LEFT
Value_Align	Value_Align	Alignment of text in the Value field if the size is greater than the length of the text.	LEFT
Units_Align	Units_Align	Alignment of text in the Units field if the size is greater than the length of the text.	LEFT
Font_Name	Font_Name	Font name for the applet text.	SANSERIF
Font_Bold	Font_Bold	Applet text is bold if configured as TRUE.	FALSE
Font_Italic	Font_Italic	Applet text is italic if configured as TRUE.	FALSE
Font_Size	Font_Size	Applet text size.	12

The HTML code with the parameters in the window above is as follows:

```
<APPLET
codebase="/classes"
archive="widgetslite.jar"
code="com.schneiderautomation.factorycast.gateway.widgetslite.LiveLabelApplet"
width="130" height="30" >
  <PARAM name="progressbar" value="true">
  <PARAM name="progresscolor" value="#000000">
  <PARAM name="ADDRESS" value="1">
  <PARAM name="UNITID" value="255">
  <PARAM name="RATE" value="1000">
  <PARAM name="DATATYPE" value="REGISTER">
  <PARAM name="LABEL" value="label">
  <PARAM name="LABEL_WIDTH" value="25">
  <PARAM name="UNITS" value="mm">
  <PARAM name="UNITS_WIDTH" value="5">
</APPLET>
```

### Horizontal or Vertical Indicator

The indicator gives an analog representation of the value of a direct address of a Modbus slave by drawing a horizontal or vertical bar whose length is proportional to the value and which represents a percentage of its range in physical units.

The properties of the indicator are as follows:

Property	HTML code	Description	Limits
Address	Address	Modbus address of the data type (1 to 65535) or symbol name.	See Note 1 (see page 529)
Data type	Datatype	Modbus address data type (not used for symbols).	See Note 2 (see page 529)
Label	Label	Label to be displayed as part of the graphic object.	See Note 5 (see page 529)
Major scale graduation	MajorTics	Number of major graduations (marked) in the scale.	0 to 100
Minor scale graduation	MinorTics	Number of minor graduations (not marked) in the scale.	0 to 100
Scale precision	Precision	Number of decimal places to be shown for the scale graduations (set to -1 to use a general exponential format).	-1 to 6
Maximum EU Value	Maximum	Maximum value, in physical units, of the direct address for scaling.	
Minimum EU Value	Minimum	Minimum value, in physical units, of the direct address for scaling.	



Property	HTML code	Description	Limits
Maximum PLC Value	MaxValue	Gross maximum value (without scale) of the direct address in the PLC.	See Note 3 (see page 529)
Minimum PLC Value	MinValue	Gross minimum value (without scale) of the direct address in the PLC.	See Note 3 (see page 529)
Limit Deadband[%]	Deadband	Neutral range (as a percentage of the UP range) to apply to verification of the High/Low limit.	0 to 10
High High Limit Value	LimitHiHi	Value expressed in physical units of the "High High" limit.	
High Limit Value	LimitHi	Value expressed in physical units of the "High" limit.	
Low Limit Value	LimitLo	Value expressed in physical units of the "Low" limit.	
Low Low Limit Value	LimitLoLo	Value expressed in physical units of the "Low Low" limit.	
High High Limit Value Color	ColorHiHi	Color of the indicator bar if the scale value is greater than the "High High" limit.	
High Limit Value Color	ColorHi	Color of the indicator bar if the scale value is greater than the "High" limit.	
Low Limit Value Color	ColorLo	Color of the indicator bar if the scale value is less than the "Low" limit.	
Low Low Limit Value Color	ColorLoLo	Color of the indicator bar if the scale value is less than the "Low Low" limit.	
Border Width	BorderWidth	Width of border.	
Unit ID	UnitId	Address of the Modbus slave (not used for symbols).	1 to 255
Polling rate [ms]	Rate	Scanning value.	

Property	HTML code	Description	Limits
Maximum PLC Value	MaxValue	Gross maximum value (without scale) of the direct address in the PLC.	See Note 3 (see page 529)
Minimum PLC Value	MinValue	Gross minimum value (without scale) of the direct address in the PLC.	See Note 3 (see page 529)
Limit Deadband[%]	Deadband	Neutral range (as a percentage of the UP range) to apply to verification of the High/Low limit.	0 to 10
High High Limit Value	LimitHiHi	Value expressed in physical units of the "High High" limit.	
High Limit Value	LimitHi	Value expressed in physical units of the "High" limit.	
Low Limit Value	LimitLo	Value expressed in physical units of the "Low" limit.	
Low Low Limit Value	LimitLoLo	Value expressed in physical units of the "Low Low" limit.	
High High Limit Value Color	ColorHiHi	Color of the indicator bar if the scale value is greater than the "High High" limit.	
High Limit Value Color	ColorHi	Color of the indicator bar if the scale value is greater than the "High" limit.	
Low Limit Value Color	ColorLo	Color of the indicator bar if the scale value is less than the "Low" limit.	
Low Low Limit Value Color	ColorLoLo	Color of the indicator bar if the scale value is less than the "Low Low" limit.	
Border Width	BorderWidth	Width of border.	
Unit ID	UnitId	Address of the Modbus slave (not used for symbols).	1 to 255
Polling rate [ms]	Rate	Scanning value.	

The HTML code with the parameters in the window above is as follows:

```
<APPLET
codebase="/classes"
archive="widgets-lite.jar"
code="com.schneiderautomation.factorycast.gateway.
indicators.LiveHorizontalIndicatorApplet"
width="180" height="160" >
  <PARAM name="progressbar" value="true">
  <PARAM name="progresscolor" value="#000000">
  <PARAM name="debug" value="0">
  <PARAM name="BACKGRND" value="LT_GRAY">
  <PARAM name="address" value="1">
  <PARAM name="datatype" value="REGISTER">
  <PARAM name="label" value="label">
  <PARAM name="majorTics" value="100">
  <PARAM name="minorTics" value="0">
  <PARAM name="precision" value="1">
  <PARAM name="maximum" value="100">
  <PARAM name="minimum" value="0">
  <PARAM name="maxValue" value="65535">
  <PARAM name="minValue" value="0">
  <PARAM name="borderWidth" value="25">
  <PARAM name="limitHiHi" value="99">
  <PARAM name="limitHi" value="95">
  <PARAM name="limitLo" value="5">
  <PARAM name="limitLoLo" value="4">
  <PARAM name="deadband" value="1">
  <PARAM name="rate" value="1000">
  <PARAM name="unitId" value="255">
  <PARAM name="colorHiHi" value="RED">
  <PARAM name="colorHi" value="ORANGE">
  <PARAM name="colorLoLo" value="BLUE">
  <PARAM name="colorLo" value="CYAN">
</APPLET>
```

## Horizontal or Vertical Slider

A horizontal or vertical slider gives an analog representation of the value of a direct address of a Modbus device by drawing a slider whose cursor is proportional to the value and which represents a percentage of its range in physical units. Using the mouse, the user can change the value of the slider and trigger the sending of a new value to the Modbus slave.

The properties of the slider are as follows:

Property	HTML code	Description	Limits
Address	Address	Modbus address of the data type (1 to 65535) or symbol name.	See Note 1 (see page 529)
Data type	Datatype	Modbus address data type (not used for symbols).	See Note 2 (see page 529)
Label	Label	Label to be displayed as part of the graphic object.	See Note 5 (see page 529)
Major Scale Division	MajorTics	Number of major graduations (marked) in the scale.	0 to 100
Minor Scale Division	MinorTics	Number of minor graduations (not marked) in the scale.	0 to 100
Scale Precision	Precision	Number of decimal places to be shown for the scale graduations (set to -1 to use a general exponential format).	-1 to 6
Maximum EU value	Maximum	Maximum value, in physical units, of the direct address for scaling.	
Minimum EU Value	Minimum	Minimum value, in physical units, of the direct address for scaling.	

Property	HTML code	Description	Limits
Maximum PLC Value	MaxValue	Gross maximum value (without scale) of the direct address in the PLC.	See Note 3 (see page 529)
Minimum PLC Value	MinValue	Gross minimum value (without scale) of the direct address in the PLC.	See Note 3 (see page 529)
Unit Increment	UnitIncrement	Amount by which the scale value is modified when the user clicks on the slider arrows.	
Block Increment	BlockIncrement	Amount by which the scale value is modified when the user clicks on the slide area of the bar.	
Border Width	BorderWidth	Width (in pixels) of the border of the graphic object.	
Unit ID	UnitID	Address of the Modbus slave (not used for symbols).	1 to 255
Polling rate [ms]	Rate	Scanning value.	
Read Only	ReadOnly	TRUE = read value, FALSE = read/write value.	

The HTML code with the parameters in the window above is as follows:

```
<APPLET
codebase="/classes"
archive="widgetslite.jar"
code="com.schneiderautomation.factorycast.gateway.sliders.
LiveHorizontalSliderApplet"
width="180" height="160" >
  <PARAM name="progressbar" value="true">
  <PARAM name="progresscolor" value="#000000">
  <PARAM name="debug" value="0">
  <PARAM name="BACKGRND" value="LT_GRAY">
  <PARAM name="address" value="1">
  <PARAM name="datatype" value="REGISTER">
  <PARAM name="label" value="label">
  <PARAM name="majorTics" value="100">
  <PARAM name="minorTics" value="0">
  <PARAM name="precision" value="1">
  <PARAM name="maximum" value="100">
  <PARAM name="minimum" value="0">
  <PARAM name="maxValue" value="65535">
  <PARAM name="minValue" value="0">
  <PARAM name="unitIncrement" value="1">
  <PARAM name="blockIncrement" value="1">
  <PARAM name="rate" value="1000">
  <PARAM name="unitId" value="255">
  <PARAM name="borderWidth" value="20">
  <PARAM name="readOnly" value="False">
</APPLET>
```

### Horizontal or Vertical Selector

A horizontal or vertical selector allows the user to choose from a number of options. Once the selection has been made, the value corresponding to the choice is sent to the equipment. The choices are represented by the marks on a "scale", the current selection being indicated by the position of the cursor on a slider.

The properties of the slider are as follows:

Property	HTML code	Description	Limits
Address	Address	Modbus address of the data type (1 to 65535) or symbol name.	See Note 1 (see page 529)
Data type	Datatype	Modbus address data type (not used for symbols).	See Note 2 (see page 529)
Label	Label	Label to be displayed as part of the graphic object.	See Note 5 (see page 529)
Choices	Choices	Setpoints to apply to the address value.	
Border Width	BorderWidth	Width (in pixels) of the border of the graphic object.	
Unit ID	UnitId	Address of the Modbus slave (not used for symbols).	1 to 255
Polling rate [ms]	Rate	Scanning value.	
Read Only	ReadOnly	True = read value, False = read/write value.	

The HTML code with the parameters in the window above is as follows:

```
<APPLET
codebase="/classes"
archive="widgets-lite.jar"
code="com.schneiderautomation.factorycast.gateway.selectors
.LiveHorizontalSelectorApplet"
width="180" height="160" >
  <PARAM name="progressbar" value="true">
  <PARAM name="progresscolor" value="#000000">
  <PARAM name="debug" value="0">
  <PARAM name="BACKGRND" value="LT_GRAY">
  <PARAM name="address" value="1">
  <PARAM name="datatype" value="REGISTER">
  <PARAM name="label" value="label">
  <PARAM name="scaleVisible" value="True">
  <PARAM name="choices" value="1=1000,9=9000,50=50000">
  <PARAM name="rate" value="1000">
  <PARAM name="unitId" value="255">
  <PARAM name="borderWidth" value="20">
  <PARAM name="readOnly" value="False">
</APPLET>
```

## Push Button

When actuated with the mouse, a push button allows the user to send a preset value to one or more Modbus slaves.

The properties of the push button are as follows:

Property	HTML code	Description	Limits
Address	Address	Modbus address of the data type (1 to 65535) or symbol name.	See Note 1 (see page 529)
Data type	Datatype	Modbus address data type (not used for symbols).	See Note 2 (see page 529)
Label	Label	Label to be displayed as part of the graphic object.	See Note 5 (see page 529)
Values	Values	Setpoints to apply to the address value when the button is pressed.	See Note 4 (see page 529)
Reset Values	ResetValues	Value to apply when the Reset Delay ends.	
Reset Delay [ms]	ResetDelay	Time in ms counted down after the button is pressed.	
Button Label	ButtonLabel	Label to display on the button.	
Border Width	BorderWidth	Width (in pixels) of the border of the graphic object.	
Unit ID	UnitId	Address of the Modbus slave (not used for symbols).	1 to 255
Polling rate [ms]	Rate	Scanning value.	



The HTML code with the parameters in the window above is as follows:

```
<APPLET
codebase="/classes"
archive="widgets-lite.jar"
code="com.schneiderautomation.factorycast.gateway.pushButton
.LivePushButtonApplet"
width="180" height="160" >
  <PARAM name="progressbar" value="true">
  <PARAM name="progresscolor" value="#000000">
  <PARAM name="debug" value="0">
  <PARAM name="BACKGRND" value="LT_GRAY">
  <PARAM name="address" value="1">
  <PARAM name="datatype" value="REGISTER">
  <PARAM name="label" value="label">
  <PARAM name="values" value="10,50">
  <PARAM name="resetValues" value="0">
  <PARAM name="resetDelay" value="15000">
  <PARAM name="borderWidth" value="25">
  <PARAM name="buttonLabel" value="button">
  <PARAM name="rate" value="1000">
  <PARAM name="unitID" value="255">
</APPLET>
```

**NOTE:** In this example, when the user presses the button the value 10 is applied to address 1 and the value 50 to address 2 for 15,000 ms. After 15,000 ms the two addresses are set to the value 0 (resetValue).

## Direct Output Window

This applet allows a user to enter a value in a text input field directly from the keyboard. If the text entered is a numerical value between the upper and lower preset limits, an OK button is activated. The value entered is sent to the Modbus slave each time the user clicks **OK** or **ENTER** (if the input field is active for keyboard input).

The direct output properties are as follows:

Property	HTML code	Description	Limits
Address	Address	Modbus address of the data type (1 to 65535) or symbol name.	See Note 1 (see page 529)
Data type	Datatype	Modbus address data type (not used for symbols).	See Note 2 (see page 529)
Label	label	Label to be displayed as part of the graphic object.	See Note 5 (see page 529)
Maximum EU Value	maximum	Maximum value, in physical units, of the direct address for scaling.	
Minimum EU Value	mininum	Minimum value, in physical units, of the direct address for scaling.	
Maximum PLC Value	maxValue	Gross maximum value (without scale) of the direct address in the PLC.	See Note 3 (see page 529)
Minimum PLC Value	minValue	Gross minimum value (without scale) of the direct address in the PLC.	See Note 3 (see page 529)
Maximum Input [EU]	MaxInputValue	Maximum setpoint for the input.	
Minimum Input [EU]	MinInputValue	Minimum setpoint for the input.	
Border Width	BorderWidth	Width (in pixels) of the border of the graphic object.	
Unit ID	UnitId	Address of the Modbus slave (not used for symbols).	1 to 255
Polling rate [ms]	Rate	Scanning value.	

The HTML code with the parameters in the window above is as follows:

```
<APPLET
codebase="/classes"
archive="widgets-lite.jar"
code="com.schneiderautomation.factorycast.gateway.direct
Output.LiveDirectOutputApplet"
width="180" height="160" >
  <PARAM name="progressbar" value="true">
  <PARAM name="progresscolor" value="#000000">
  <PARAM name="debug" value="0">
  <PARAM name="BACKGRND" value="LT_GRAY">
  <PARAM name="address" value="1">
  <PARAM name="datatype" value="REGISTER">
  <PARAM name="label" value="label">
  <PARAM name="maximum" value="100">
  <PARAM name="minimum" value="0">
  <PARAM name="maxValue" value="65535">
  <PARAM name="minValue" value="0">
  <PARAM name="maxInputValue" value="100">
  <PARAM name="minInputValue" value="0">
  <PARAM name="rate" value="1000">
  <PARAM name="unitId" value="255">
  <PARAM name="borderWidth" value="25">
</APPLET>
```

## Indicator Light

The indicator light provides a double indication of the value of a direct address in a device. If the Input inverted property is not set to TRUE, a zero input value is declared OFF and a non-zero value is declared ON. If the Flash Interval property is set to a positive value, the indicator light will flash when the input value is equal to ON.

The screenshot shows a dialog box titled "Properties [Indicator Light]". It contains the following fields and values:

- Address: 2
- Data Type: REGISTER
- Label: label
- Off Word: off
- On Word: on
- Off Background Color: LT\_GRAY
- On Background Color: GREEN
- Off Foreground Color: DK\_GREY
- On Foreground Color: ORANGE
- Flash Interval [ms]: 1000
- Input Inverted: False
- Border Width: 25
- Border Color: BLUE
- Shape: ROUND RECTAN
- Unit ID: 255
- Polling rate [ms]: 1000

Buttons for "OK" and "Cancel" are located at the bottom right of the dialog.

The properties of the indicator light are as follows:

Property	HTML code	Description	Limits
Address	Address	Modbus address of the data type (1 to 65535) or symbol name.	See Note 1 (see page 529)
Data type	Datatype	Modbus address data type (not used for symbols).	See Note 2 (see page 529)
Label	Label	Label to be displayed as part of the graphic object.	See Note 5 (see page 529)
Off Word	OffWord	Text to be displayed when the input value is OFF.	
On Word	OnWord	Text to be displayed when the input value is ON.	
Off Background Color	OffWordBackground	Background color of the indicator light when Off Word is displayed	
ON Background Color	OnWordBackground	Background color of the indicator light when ON Word is displayed	
Off Foreground Color	OffWordForeground	Color of the text of the Off Word.	

Property	HTML code	Description	Limits
On Foreground Color	OnWordForeground	Color of the text of the On Word.	
Flash Interval	FlashInterval	The flashing time for the indicator light (expressed in milliseconds) when the input value is ON. Set to zero for no flashing.	200 to 2000
Input inverted	InputInverted	On TRUE, inverts the input value (the indicator light displays the Off Word when the input value is ON).	
Border Width	BorderWidth	Width (in pixels) of the border of the graphic object.	
Border Color	BorderColor	Color of the border.	
Shape	Shape	Shape (circular, rectangular, etc.) of the indicator light.	
Unit ID	UnitId	Address of the Modbus slave (not used for symbols).	1 to 255
Polling rate [ms]	Rate	Scanning value.	

The HTML code with the parameters in the window above is as follows:

```
<p><APPLET
codebase="/classes"
archive="widgets-lite.jar"
code="com.schneiderautomation.factorycast.gateway.indicatorLight.LiveIndicatorLightApplet"
width="180" height="160" >
  <PARAM name="progressbar" value="true">
  <PARAM name="progresscolor" value="#000000">
  <PARAM name="debug" value="0">
  <PARAM name="BACKGRND" value="LT_GRAY">
  <PARAM name="address" value="2">
  <PARAM name="datatype" value="REGISTER">
  <PARAM name="label" value="label">
  <PARAM name="offWord" value="off">
  <PARAM name="onWord" value="on">
  <PARAM name="offWordBackground" value="LT_GRAY">
  <PARAM name="onWordBackground" value="GREEN">
  <PARAM name="offWordForeground" value="DK_GRAY">
  <PARAM name="onWordForeground" value="ORANGE">
  <PARAM name="flashInterval" value="1000">
  <PARAM name="inputInverted" value="False">
  <PARAM name="borderWidth" value="25">
  <PARAM name="borderColor" value="BLUE">
  <PARAM name="shape" value="ROUND_RECTANGLE">
  <PARAM name="rate" value="1000">
  <PARAM name="unitId" value="255">
</APPLET>
```

## Rotary Slider

A rotary slider gives an analog representation of the value of a direct address of a Modbus device. On a circular dial, it draws a position proportional to the value of the address and represents a percentage of its range in physical units. The size of the circular dial (cycle in degrees of a circle) and the color of the button can be configured. Using the mouse, the user can change the value of the rotary slider and trigger the sending of a new value to the equipment.

The screenshot shows a dialog box titled "Properties [Rotary Slider]". It contains the following fields and values:

- Address: 1
- Data Type: REGISTER
- Label: label
- Major Scale Divisions: 65535
- Minor Scale Divisions: 0
- Scale Precision: 0
- Dial Degrees Sweep: 180
- Maximum EU Value: 100
- Minimum EU Value: 0
- Maximum PLC Value: 65535
- Minimum PLC Value: 0
- Border Width: 25
- Unit ID: 255
- Polling rate [ms]: 1000
- Read Only: FALSE

Buttons for "OK" and "Cancel" are located at the bottom right of the dialog.

The properties of the slider are as follows:

Property	HTML code	Description	Limits
Address	Address	Modbus address of the data type (1 to 65535) or symbol name.	See Note 1 (see page 529)
Data type	Datatype	Modbus address data type (not used for symbols).	See Note 2 (see page 529)
Label	Label	Label to be displayed as part of the graphic object.	See Note 5 (see page 529)
Major scale graduation	MajorTics	Number of major graduations (marked) in the scale.	0 to 100
Minor scale graduation	MinorTics	Number of minor graduations (not marked) in the scale.	0 to 100
Scale precision	Precision	Number of decimal places to be shown for the scale graduations (set to -1 to use a general exponential format).	-1 to 6
Dial Degrees Sweep	DegSweep	Portion of circular dial to be used to draw the scale.	60 to 300
Maximum EU Value	Maximum	Maximum value, in physical units, of the direct address for scaling.	

Property	HTML code	Description	Limits
Minimum EU Value	Minimum	Minimum value, in physical units, of the direct address for scaling.	
Maximum PLC Value	MaxValue	Gross maximum value (without scale) of the direct address in the PLC.	See Note 3 (see page 529)
Minimum PLC Value	MinValue	Gross minimum value (without scale) of the direct address in the PLC.	See Note 3 (see page 529)
Border Width	BorderWidth	Width (in pixels) of the border of the graphic object.	
Unit ID	UnitId	Address of the Modbus slave (not used for symbols).	1 to 255
Polling rate [ms]	Rate	Scanning value.	
Read Only	ReadOnly	True = read value, False = read/write value.	

The HTML code with the parameters in the window above is as follows:

```
<APPLET
codebase="/classes"
archive="widgetslite.jar"
code="com.schneiderautomation.factorycast.gateway.sliders.
LiveRotosliderApplet"
width="180" height="160" >
  <PARAM name="progressbar" value="true">
  <PARAM name="progresscolor" value="#000000">
  <PARAM name="debug" value="0">
  <PARAM name="BACKGRND" value="LT_GRAY">
  <PARAM name="address" value="1">
  <PARAM name="datatype" value="REGISTER">
  <PARAM name="label" value="label">
  <PARAM name="majorTics" value="65535">
  <PARAM name="minorTics" value="0">
  <PARAM name="precision" value="0">
  <PARAM name="degSweep" value="180">
  <PARAM name="maximum" value="100">
  <PARAM name="minimum" value="0">
  <PARAM name="maxValue" value="65535">
  <PARAM name="minValue" value="0">
  <PARAM name="borderWidth" value="25">
  <PARAM name="rate" value="1000">
  <PARAM name="readOnly" value="False">
  <PARAM name="unitID" value="255">
```

## Trend Curves

The trend curve can be used to represent the value of a direct address of a Modbus device or of a module internal register graphically on a curve according to a time interval. This graphic object has no configuration window, therefore the HTML code must be written out.

HTML code	Description	Limits
Address	Modbus address of the data type (1 to 65535) or symbol name.	See Note 1 (see page 529)
Datatype	Modbus address data type (not used for symbols).	See Note 2 (see page 529)
Label	Label to be displayed as part of the graphic object.	See Note 5 (see page 529)
Pens	Color of the curve (black by default).	
Griddivisions	Number of intervals on the grid.	
Chartbackground	Curve background color.	
Gridcolor	Color of the interval grid.	
Scalecolor	Color of the text for the X-axis and Y-axis information.	
Updateinterval	Time in seconds between each interval.	
Timedivisions	Number of time intervals.	
Maximum	Maximum value, in physical units, of the direct address for scaling.	
Minimum	Minimum value, in physical units, of the direct address for scaling.	
MaxValue	Gross maximum value (without scale) of the direct address in the PLC.	See Note 3 (see page 529)
MinValue	Gross minimum value (without scale) of the direct address in the PLC.	See Note 3 (see page 529)
BorderWidth	Width (in pixels) of the border of the graphic object.	
UnitId	Address of the Modbus slave (not used for symbols).	1 to 255
Rate	Scanning value.	



The HTML code with the parameters in the window above is as follows:

```
<APPLET
codebase="/classes"
code="com.schneiderautomation.factorycast.gateway.chart.Li
veChartApplet"
archive="widgetslite.jar" width="685" height="363">
<param name="progressbar" value="true">
<param name="progresscolor" value="#000000">
<param name="pens" value="p1=black,p2=blue,p3=green">
<param name="address" value="5,6,7">
<param name="minimum" value="0">
<param name="maximum" value="100">
<param name="minValue" value="0">
<param name="maxValue" value="100">
<param name="griddivisions" value="5">
<param name="gridcolor" value="blue">
<param name="updateinterval" value="10">
<param name="timedivisions" value="10">
<PARAM name="unitId" value="255">
<param name="chartbackgrnd" value="white">
</applet>
```

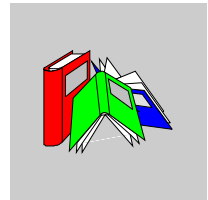
## Notes

1.	Modbus address of the data type (1 to 65535) or symbol name configured by Web Designer for TSX ETG 30**. You must set the ??? parameter to "PERSISTENT" mode.	
2.	The various values of the Data Type property for a direct Modbus address have the following meanings:	
	Data type	Meaning
	COIL	output bit (Boolean)
	DISCRETE INPUT	input bit (Boolean)
	REGISTER	16-bit signed integer
	INT32	32-bit signed integer
	INT32SWAP	32-bit signed integer with least significant and most significant words inverted
	INPUT REGISTER	16-bit signed integer for analog input
3.	The limits of the Maximum PLC Value and Minimum PLC Value properties are the PLC natural limits of the configured Data type property.	
4.	For a push button you must specify at least one value. If several values are entered, they will be assigned to an address table starting with the direct address indicated.	
5.	For the applet to display a numerical data value instead of a label, specify parameters in the HTML code as follows: name = "label" value = "\$data\$".	



---

# Glossary



---

## A

### applet

Software component that runs in the context of another program, for example a Web browser.

### ASCII

#### **American Standard Code for Information Interchange.**

This is an American code (but now an international standard) which allows all alphanumeric characters used in English, punctuation marks, some graphics characters and various commands to be defined with 7 bits.

### AT commands

Also called **Hayes Commands**: Set of commands for various telephone-line manipulations, dialing and hanging up for instance.

## B

### BIT

Contraction of Binary Digit.

This is the binary unit of information content, which can represent two separate values (or states): 0 or 1.

A field of 8 bits constitutes 1 **byte**.

**BOOTP**

**Bootstrap Protocol:** Protocol for booting diskless terminals or stations by centralized management of network parameters.

**C**

**CF card**

**Compact Flash card:** Type of data storage device, used in portable electronic devices.

**communication error**

Error detected by the module when the periodic exchanges with the PLC stop.

**configuration**

The configuration comprises all the data that defines the device (invariable) and that is necessary to the operation of the module.

**CPU**

**Central Processing Unit.**

The microprocessor. This comprises the entire control unit and the arithmetic unit. The purpose of the control unit is to extract the execution instruction from the central memory along with the data needed to execute this instruction, to establish electrical connections in the arithmetic and logic unit and to start the processing of this data in the unit. **ROM** or **RAM** memories are sometimes included on the same chip, and sometimes even I/O interfaces or buffers.

**CRC**

**Cyclic Redundancy Check:** Type of hash function used to produce a checksum – a small, fixed number of bits – against a block of data, such as a packet of network traffic or a block of a computer file.

**CTS**

**Clear To Send:** Acknowledge RTS and allow DTE to transmit.

## D

### DCD

**Data Carrier Detect:** Indicates that a good carrier is being received from the remote modem

### DHCP

**Dynamic Host Configuration Protocol:** Protocol allowing a station connected to the network to obtain its configuration dynamically.

### DIN

**Deutsches Institut für Normung:** German standards institute.

### DNS

**Domain Name System:** It stores and associates many types of information with domain names, but most importantly, it translates domain names (computer hostnames) to IP addresses.

### driver

Program which informs the operating system of the presence and characteristics of a peripheral.

### DSR

**Data Set Ready:** Indicates to the computer that it is ready to receive data

### DTE

**Data Terminal Equipment:** End instrument that converts user information into signals for transmission, or reconverts the received signals into user information.

### DTR

**Data Terminal Ready:** Indicates that the computer is ready to receive data.

## F

### **FactoryCast HMI**

Active Web server that executes HMI functions integrated in a PLC module. With the active Web server, you do not need to use polling to update the HMI/SCADA database.

### **FDR**

**Faulty Device Replacement:** Automatic configuration recovery service provided by the module.

### **firewall**

Information technology (IT) security device which is configured to permit, deny or proxy data connections set and configured by the organization's security policy.

### **Flash memory**

Form of non-volatile computer memory that can be electrically erased and reprogrammed.

### **FTP/TFTP**

**File Transfer Protocol/Trivial File Transfer Protocol:** Network file transfer protocol.

## G

### **GND**

**Ground Signal:** Common return for the other connections

### **GPRS**

**General Packet Radio Service:** A radio technology for GSM networks that adds packet-switching protocols and shorter set-up time for ISP connections.

### **GSM**

**Global System for Mobile Communications** is the most popular standard for mobile telephones in the world.

## H

### HMI

**Human Machine Interface:** The aggregate of means by which people (the users) interact with a particular machine, device, computer program or other complex tool (the system).

### HTML

**Hyper Text Markup Language** is the predominant markup language for the creation of web pages. It provides a means to describe the structure of text-based information in a document and to supplement that text with interactive forms, embedded images, and other objects.

### HTTP

**Hyper Text Transfer Protocol:** Network transfer protocol for documents written in hypertext (links).

## I

### IP

**Internet Protocol:** Data-oriented protocol used for communicating data across a packet-switched internetwork (i.e. the Internet).

### ISO

International Standards Organization. The ISO code is the most widely used. Formats, symbols, transmission rules are all covered by ISO standards. AFNOR is a member of ISO.

### ISP

**Internet Service Provider:** Business or organization that sells to consumers access to the Internet and related services.

## M

### MIB

**Management Information Base:** Database used by the SNMP protocol for network management and containing information on data transmission, station or router components, etc.

- MIB II: standard MIB
- Schneider Electric MIB: private MIB

## N

### NAT

**Network Address Translation:** is the translation of an Internet Protocol address (IP address) used within one network to a different IP address known within another network.

### NTP

**Network Time Protocol:** Protocol for synchronizing the clocks of computer systems over packet-switched, variable-latency data networks.

## P

### PAP

**Password Authentication Protocol:** Password identification protocol used for remote modem connections.

### PL7

Schneider Electric PLC programming software.

### PLC

**Programmable Logic Controller:** It is a small computer used for automation of industrial processes, such as control of machinery on factory assembly lines.

### PPP

**Point-to-Point Protocol:** Point-to-point communication protocol used for modem connections.



**Premium**

Family of Schneider Electric PLCs.

**PSTN/RTC**

**Public Switched Telephone Network/Réseau Téléphonique Commuté:** The network of the world's public circuit-switched telephone networks.

**Q****Quantum**

Family of Schneider Electric PLCs.

**R****RAS**

**Remote Access Server:** Hardware and software to enable the remote access to tools or information that typically reside on a network of IT devices.

**RD**

**Received Data:** Data sent from DCE to DTE.

**RGB**

Additive model in which red, green, and blue (often used in additive light models) are combined in various ways to reproduce other colors.

**RI**

**Ring Indicator:** Detects a ring signal from the telephone line.

**RS232**

Serial communication standard that in particular defines the following operating voltage:

- A signal of +3 to +25V indicates a logic 0
- A signal of -3V to -25V indicates a logic 1

Between +3V and -3V the signal is regarded as invalid.

RS 232 connections are relatively sensitive to interference. The standard recommends not exceeding a distance of 15 meters and a speed of 20,000 baud (bps) maximum.

**RS485**

Serial connection standard operates at +/-5V differential. The connection uses the same wire for transmission and receipt. Their "3-status" outputs allow them to switch to listening mode when transmission is completed.

**RTS**

**Request To Send:** Asserted (set to 0) by DTE to prepare DCE to receive data.

**RTU**

**Remote Terminal Unit:** Device which interfaces objects in the physical world to a distributed control system or SCADA system by transmitting telemetry data to the system and/or altering the state of connected objects based on control messages received from the system.

**RUN**

Function used to start execution of the application program in the PLC.

**S**

**SCADA**

**Supervisory Control And Data Acquisition:** Software that, interfacing with a programmable logic controller, gathers and analyzes information used to monitor and control commercial equipment.

**SMTP**

**Simple Mail Transfer Protocol:** Application protocol used to transmit messages via the Internet and direct them to a mailbox.

**SNMP**

**Simple Network Management Protocol:** Network management protocol for controlling a network remotely by polling the stations for their status and modifying their configuration, performing security tests and viewing information relating to data transmission. It can also be used to manage software and databases remotely.

**SQL**

**Structured Query Language:** Used to query (request data from) a relational database.

**T****TCP**

**Transmission Control Protocol:** Virtual circuit protocol that is one of the core protocols of the Internet protocol suite, often simply referred to as TCP/IP.

**TCP/IP**

The set of communications protocols that implement the protocol stack on which the Internet and most commercial networks run.

**TD**

**Transmitted Data:** Data sent from DTE to DCE.

**Time Out**

**Expiry of a waiting time.** Stops the application or disconnects after a lengthy period of non-use.

**U****UDP**

**User Datagram Protocol:** One of the core protocols of the Internet protocol suite. Using UDP, programs on networked computers can send short messages sometimes known as datagrams to one another.

**URL**

**Uniform Resource Locator:** The global address of documents and other resources on the World Wide Web.

## V

### VPN

**Virtual Private Network:** A private network that is configured within a public network. It uses encryption and other security mechanisms so that only authorized users can access the network and that the data cannot be intercepted.

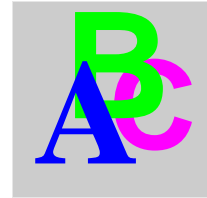
## W

### Web Proxy

A Web proxy server is a server which services the requests of its clients by making requests to other servers. A client connects to the proxy server, requesting a Web page from a different server.

---

# Index



---

## A

- active pages service
  - configuration, *318*
  - presentation, *316*
- addressing, *75*
  - flat, *98*
- applets
  - FrontPage, *496, 504*
  - HTML, *479, 481, 484*

## B

- battery
  - insertion and removal, *43*

## C

- cables, *66*
- calculation service
  - configuration, *288*
  - presentation, *286*
- CF card
  - backup, *52*
  - characteristics, *61*
  - Installation and removal, *51*
  - precautions, *51*
- changing passwords, *79*
- characteristics
  - electrical, *70*
  - environmental, *72*
  - physical description, *19*
  - standards, *71*

- compatibility, *23*
- conditions of use, *72*
- configuration pages, *75*
- connectors, *56*
- custom web pages
  - FrontPage, *490*
  - graphic objects, *509*
  - home page, *475*
  - inserting applets, *478*
  - uploading, *473*
  - uploading pages, *508*
  - using Java applets, *479, 481, 484, 496, 504*

## D

- daisy chain, *256*
- Data Editor
  - direct address, *407*
  - spreadsheets, *404*
  - symbol, *406*
  - templates, *403*
  - variable, *406*
- Data Editor Lite
  - configuration, *412*
- data templates, *403*
- database service
  - configuration, *295*
  - database table, *299*
  - presentation, *294*

- datalogging service
  - configuration, 306
  - presentation, 302
- device explorer
  - configuration, 249
  - presentation, 247
- device service
  - configuration, 270
  - presentation, 259, 267
- DHCP
  - client, 230
  - FDR, 231
  - presentation, 229
- diagnostics, 68
  - service status, 261
- dimensions, 35
- direct addresses
  - write-enabled, 408
- direct cable, 120
- DNS
  - configuration, 236
  - presentation, 234

## E

- email service
  - configuration, 279
  - password protection, 278
  - presentation, 277
- external modem
  - addressing, 153
  - configuration, 157
  - connection priorities, 152
  - opening a connection, 149

## F

- FDR
  - presentation, 229
- FrontPage
  - extension, 490
  - uploading pages, 508
- FTP, 253

## G

- GPRS
  - contract, 163
  - IP publication, 167
  - presentation, 162
- GPRS modem
  - configuration, 165
  - opening a connection, 169
- Graphic Editor
  - applet parameters, 436
  - presentation, 423
  - property sheet, 433
  - security, 435
  - toolbar, 425
  - user functions, 430
- graphic objects
  - extended, 459
  - presentation, 438
- Graphic Viewer, 469
- GSM modem
  - addressing, 153
  - configuration, 157
  - connection priorities, 152
  - opening a connection, 149

## H

- hardware characteristics, 32
- HMI services
  - presentation, 259
- HTTP
  - General Home, 344

## I

- I/O board, 62
  - characteristics, 59, 64
  - wiring, 53
- I/O scanning, 64
- IMEI number, 50
- IP address
  - default, 115, 115
  - dynamic, 162
  - PPP interface, 115

IP control  
  configuration, 174, 355

IP filtering  
  configuration, 123  
  presentation, 351

## J

Java applets, 481, 484, 496, 504

## L

LED indicators, 68

LiveBeanApplet, 479, 481, 496, 504

LiveLabelApplet, 484

local registers, 102  
  input, 110  
  output, 110

## M

Modbus  
  addressing, 96  
  broadcast, 94  
  configuration, 99  
  performance, 100  
  presentation, 93  
  requests, 97  
  server, 96

Modbus UnitID, 191

modem  
  connection priorities, 152  
  StandBy, 224  
  Standby Configuration, 358

module  
  functions, 20  
  services, 20

mounting, 35

## N

naming, 269

NAT, 177  
  NAT On Port, 186  
  NAT On Port Limitations, 190  
  Port 502 Forwarding, 195

network topologies, 84

NTP  
  configuration, 239  
  presentation, 238

## P

PAP, 149

passwords, 79

phone  
  configuration, 216  
  presentation, 356

pocket data editor  
  presentation, 415  
  using, 417

PoE, 254

power cord  
  installation, 41

power over Ethernet, 254

power plug, 41

PPP, 149

PPP security, 174, 213, 355

proxy, 249

PSTN modem  
  addressing, 153  
  configuration, 157  
  connection priorities, 152  
  opening a connection, 149

## R

recipe database service  
  configuration, 330

recipe service  
  configuration, 324  
  presentation, 323

registers  
  internal module registers, 102

retrieving the IP address, 120

## S

- security
  - access, *26, 29*
  - firewall, *27*
  - passwords, *26*
- serial link configuration, *77*
- services
  - service status, *261*
- SIM card, *47*
- SMS service
  - configuration, *279*
  - presentation, *277*
- SNMP
  - configuration, *244*
  - presentation, *241*
- SOAP, *125*
  - documentation, *147*
  - ModbusXMLDA, *128*
  - optimization, *143*
  - SymbolicXMLDA, *138*
- symbols
  - insert, *406*
  - modifying, *408*

## T

- TCP/IP
  - addressing, *112*
  - closing a connection, *220*
  - configuration, *121*
  - connection breakdown, *221*
  - features, *114*
  - IP address, *115*
  - opening a connection, *117*
- topologies
  - remote access, *82*
- TSX ETG 30••
  - overview, *16*

## U

- USB memory key, *61*

## V

- variables
  - insert, *406*
  - modifying, *408*
  - naming, *269*
- VPN
  - client, *201*
  - configuration, *213, 355*
  - presentation, *199*



---

## W

### website

- control home page, *387*
- custom pages, *398*
- data editor, *394*
- data editor lite, *395*
- date and time, *390*
- device explorer, *385*
- diagnostics, *363*
- DNS configuration, *359*
- Ethernet configuration, *350*
- Ethernet statistics, *366*
- FDR Command, *389*
- FDR statistics, *380*
- gateway statistics, *365*
- graphic editor, *396*
- graphic viewer, *397*
- IP filtering presentation, *351*
- Modbus configuration, *349*
- Modbus statistics, *368*
- modem configuration, *354*
- modem statistics, *370, 374, 376, 378*
- monitoring home page, *393*
- NAT Configuration, *357*
- NTP configuration, *360*
- NTP statistics, *382*
- phone presentation, *356*
- Port 502 Forwarding, *391*
- presentation, *341*
- reboot, *388*
- security, *348*
- setup home page, *347*
- SNMP configuration, *361*
- SNMP statistics, *384*
- UnitID Configuration, *353*
- version, *345*

### wiring

- power cord, *41*

