

# TeSys U LUFP1

## FIPIO / Modbus RTU Gateway

### User's Manual

03/2009



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

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

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

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

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

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

© 2009 Schneider Electric. All rights reserved.

# Table of Contents

<b>Safety Information .....</b>	<b>4</b>
<b>1. Introduction.....</b>	<b>6</b>
1.1. Introduction to the User's Manual .....	6
1.2. Introduction to the LUFPP1 Gateway .....	8
1.3. Terminology.....	8
1.4. Introduction to the Communication "System" Architecture.....	9
1.5. Principle of LUFPP1 Gateway Configuration and Operation..	10
<b>2. Hardware Implementation of the LUFPP1 Gateway .....</b>	<b>12</b>
2.1. On Receipt .....	12
2.2. Introduction to the LUFPP7 Gateway .....	12
2.3. Mounting the Gateway on a DIN Rail .....	13
2.4. Powering the Gateway .....	14
2.5. Connecting the Gateway to the Modbus Network .....	14
2.5.1. Examples of Modbus Connection .....	15
2.5.2. Pin Outs .....	17
2.5.3. Wiring Recommendations for the Modbus Network.....	18
2.6. Connecting the LUFPP7 gateway to the Profibus-DP Network.....	20
2.6.1. Pin Outs .....	20
2.6.2. Wiring Recommendations for the FIPIO Network .....	21
2.7. Configuring the FIPIO Communication Features .....	23
2.7.1. Encoding the Gateway Address.....	23
<b>3. Signaling .....</b>	<b>25</b>
<b>4. Software Implementation of the Gateway .....</b>	<b>26</b>
4.1. Introduction.....	26
4.1.1. System Architecture .....	26
4.1.2. Configuring the Motor Starters .....	27
4.1.3. Modbus Cycle Time .....	27
4.1.4. Managing Degraded Modes With the Gateway Default Configuration .....	27
4.2. Gateway Configuration under PL7 PRO .....	31
4.2.1. WorldFIP Mode .....	31
4.2.2. Adding the LUFPP1 Gateway under PL7 PRO .....	31
4.2.3. Configuring and Adjusting the LUFPP1 Gateway .....	33
4.2.4. Selecting and Adding the Profibus-DP Master Station ..	34
4.2.5. Adjustment Parameter Values, Standard Configuration	35
4.2.6. Deleting One or More TeSys U Motor Starters from the Configuration .....	36
4.2.7. Configuring Gateway Inputs / Outputs .....	38
4.2.8. Description of Services Assigned to Gateway I / O .....	40
4.2.9. Validating and Saving the FIPIO Network Configuration .....	41
4.2.10. Assigning Symbols to the Gateway's FIPIO Objects ..	41
4.2.11. Checking the Operational Status of the Gateway .....	42
4.2.12. Modifying the Gateway Parameters in ONLINE Mode.	43
4.2.13. Debugging and Using the Gateway Configuration.....	43
4.2.14. Developing a FIPIO Application .....	43
<b>5. FIPIO Objects Available for Programming.....</b>	<b>44</b>
5.1. Periodic Command and Control Words.....	45
5.1.1. Standard Configuration with 8 TeSys U Motor Starters	45
5.1.2. Configuration under PL7 PRO with Profile FED C32 P	46
5.1.3. Configuration under ABC-LUFPP Config Tool with Profile FED C32.....	47
5.2. List of Active Slaves (LAS) Service .....	48
5.3. Indexed Periodic Variables (PKW) Service .....	49
5.3.1. Introduction to the PKW Service .....	49
5.3.2. PKW Service: Command and Response .....	50
5.3.3. Using the Indexed Periodic Variables (PKW) Service ..	51
5.3.4. Sample Uses of PKW Service .....	53
5.3.5. LUFPP1 Gateway Internal Registers .....	55
5.4. FIPIO Diagnostic Objects.....	58
5.4.1. System Words %SW128 to %SW135.....	59
5.4.2. Channel Status (%MWp.2.c0.0.2) .....	59
5.4.3. Module Status (%MWp.2.c0.MOD.2).....	59
5.5. Status of Explicit Exchanges.....	60
5.6. Parameters Specific to Profile FED C32 P .....	60
5.6.1. Configuration Parameters (%KWp.2.c0.0 to %KWp.2.c0.0.29) .....	61
5.6.2. Adjustment Parameters (%MWp.2.c0.0.20 to %MWp.2.c0.0.49).....	61
<b>6. Advanced Implementation of the Gateway .....</b>	<b>62</b>
6.1. Introduction to Advanced Implementation of the Gateway ..	62
6.2. Implementation under PL7 PRO .....	64
<b>7. Using ABC-LUFPP Config Tool .....</b>	<b>67</b>
7.1. Connecting the Gateway to the Configuration PC .....	67
7.1.1. Pin-Outs.....	67
7.1.2. RS-232 Link Protocol .....	68
7.2. Installing ABC-LUFPP Config Tool.....	69
7.3. Connecting to / Disconnecting from the Gateway .....	69
7.4. Importing the Gateway Configuration .....	70
7.5. Transferring a Configuration to the Gateway .....	71
7.5.1. Transferring the Internal Configuration (Profile FED C32 P) .....	71
7.6. Monitoring the Contents of the Gateway's Memory .....	71
<b>Appendix A:.....</b>	<b>108</b>
<b>Appendix B:.....</b>	<b>114</b>
<b>Appendix C .....</b>	<b>127</b>
<b>Appendix D .....</b>	<b>130</b>
<b>Appendix E .....</b>	<b>136</b>
<b>Index.....</b>	<b>139</b>
<b>Glossary.....</b>	<b>140</b>

# Safety Information

---

## NOTICE

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



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



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

## **DANGER**

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

## **WARNING**

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

## **CAUTION**

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

## **CAUTION**

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

## PLEASE NOTE

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

# About the book

---

**Validity Note** This document applies to all V2 gateways.

## Related Documents

Title of Documentation	Reference Number
AnyBus Communicator – User Manual	ABC_User_Manual.pdf (SDN-7061-059)
Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control	NEMA ICS 1.1 (latest edition)
Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems	NEMA ICS 7.1 (latest edition)
Modbus User Guide	TSX DG MDB E
Modicon Modbus Protocol Reference Guide	PI-MBUS-300 Rev. J

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

**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)

# 1. Introduction

---

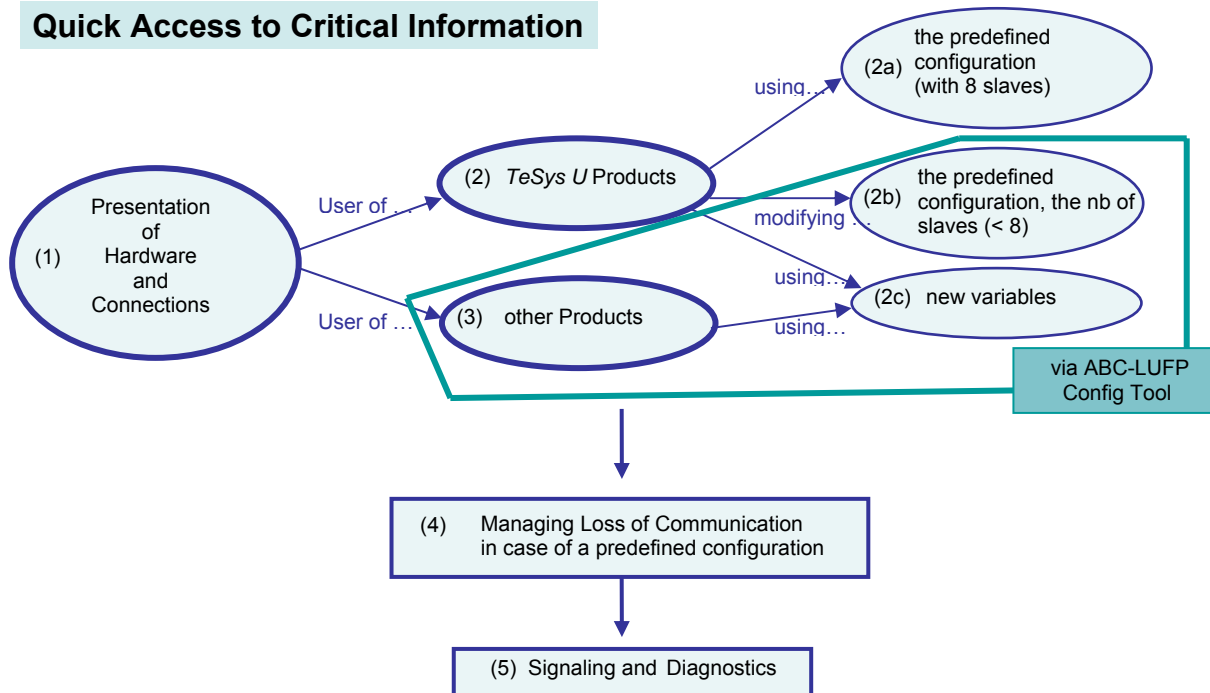
## 1.1. Introduction to the User's Manual

The LUF1 Gateway User's Manual is organised into a series of chapters, which can in turn be grouped together into three parts as follows:

- ♦ **Part I:** covers all the elements required in order to set up the gateway and operate it on a FIPIO network.
  - Chapter 1:** describes the gateway, the user manual that comes with it and the terms used in it.
  - Chapter 2:** gives an introduction to the gateway and describes all the items used when setting it up, both inside the gateway (coding wheels) and outside (cables and connectors).
  - Chapter 3:** describes the six LEDs on the front of the gateway.
  - Chapter 4:** describes the successive steps for setting the gateway up with a PLC using FIPIO. You need to use PL7 PRO to configure the gateway and thus interface a FIPIO master with Modbus slaves. The example used in this implementation enables the FIPIO master to dialog with 8 TeSys U motor starters.
  - Chapter 5:** describes all the FIPIO objects associated with the LUF1 gateway and which you can use with PL7 PRO.
  
- ♦ **Part II:** is dedicated to a third-party application, "ABC-LUF1 Config Tool", which enables the gateway to be configured in greater detail than as set out in Part I. This part therefore supersedes chapt. 4.
  - Chapter 6:** describes an alternative method for configuring the LUF1 gateway. Rather than using PL7 PRO, the gateway is configured with third-party software known as "ABC-LUF1 Config Tool", allowing for more detailed configuration of the LUF1 gateway. The example used in this implementation is the same as in Chapter 4.
  - Chapter 7:** describes how to use ABC-LUF1 Config Tool to create or modify a configuration for the gateway, but only in the context of an Advanced Implementation of the Gateway. It also presents the various software functions (add or delete a Modbus slave, add or modify a Modbus command, etc.) and the changes to be made regarding advanced gateway implementation operations with PL7 PRO (Chapter 6).
  
- ♦ **Part III:** contains the full set of Appendices which supplement the first two parts.
  - Appendix A:** Technical Characteristics describes the technical aspects of both the gateway and the networks it is interfaced with, namely the FIPIO and Modbus RTU networks.
  - Appendix B:** describes the configuration and adjustment parameters for the LUF1 gateway.
  - Appendix C:** Standard Configuration describes the main features of the configuration used in the Advanced Implementation of the Gateway (Chapter 6). However, it does not go into ABC-LUF1 Config Tool in detail.
  - Appendix D:** Sample Use under PL7 PRO provides an example of how the LUF1 gateway is used, based on the configuration described in the Software Implementation of the Gateway (Chapter 4). This example employs the command and control registers for 8 TeSys U motor starters and uses the gateway's indexed periodic variables (PKW) service and list of active slaves (LAS) service.
  - Appendix E:** Modbus Commands describes the contents of the Modbus command frames supported by the LUF1 gateway.

# 1. Introduction

## Quick Access to Critical Information



### (1) Presentation of Hardware and Connections

- **See Chapter 2**
- powering,
  - mounting,
  - Modbus connecting,
  - Profibus connecting,
  - Transmission speed and address selecting

### (2) User of TeSys U Products (LUFPP1 as a FED C32 P device)

(2a) / (2b) / (2c) Software implementation of the gateway

- **See Chapter 4 and 5**

#### (2b) reducing the number of slaves

- **See Chapter 7**
- Using ABC-LUFP Config Tool:
- install (7.2),
  - connect (7.1),
  - remove slaves (7.7)

#### (2c) access to new variables

- **See Chapter 7**
- Using ABC-LUFP Config Tool to access other registers than standard 704 (Command) and 455 (Status)
- with the same request:
- replace a register with another (for instance 455 with 458)
  - expand the size (the number of registers)
- with a supplementary request:
- add-up extra commands
  - other operations (6.8 to 6.13)

### (3) User of other Generic Modbus Products

- **See Chapter 7 (7.6 to 7.11, 7.11.2)**
- Select between:
- adapting the predefined configuration provided with the gateway, if close enough to that you wish (1 register to read and 1 to write, 1 register address to change), or
  - building up your own configuration from scratch (see ABC User Manual)

### (4) Loss of Communication

- **See Chapter 4.1.4 and Chapter 7.11.2.2**
- The variables described are:
- **Reconnect time** (unit = 10ms, default value = 10s)
  - **Retries** (default value = 3)
  - **Timeout time** (unit = 10ms, default value = 1s)

### (5) Signaling of faults and status, Diagnostics

- **See Chapter 3**
- Signaling defaults and gateway status by LEDs on the front
- **See Chapter 5.4**
- Description of the standard FIPIO diagnostics objects
- ABC-LUFP Config Tool includes the following features:
- monitoring of the gateway's input, output, and general memory areas
  - activation / deactivation of one or more Modbus slaves
  - diagnostics of the sub-network (Modbus) with statistics counters
- **See Chapter 7.5 and 7.12.3**

# 1. Introduction

## 1.2. Introduction to the LUF1 Gateway

The LUF1 gateway allows a master located on a FIPIO network to enter into a dialogue with the slaves on a Modbus RTU network. This is a generic protocol converter operating in a way which is transparent to the user.

This gateway allows you to interface many products marketed by *Schneider Electric* with a FIPIO network. These include TeSys U motor starters, Altivar drivers and Altistart soft start- soft stop units.

## 1.3. Terminology

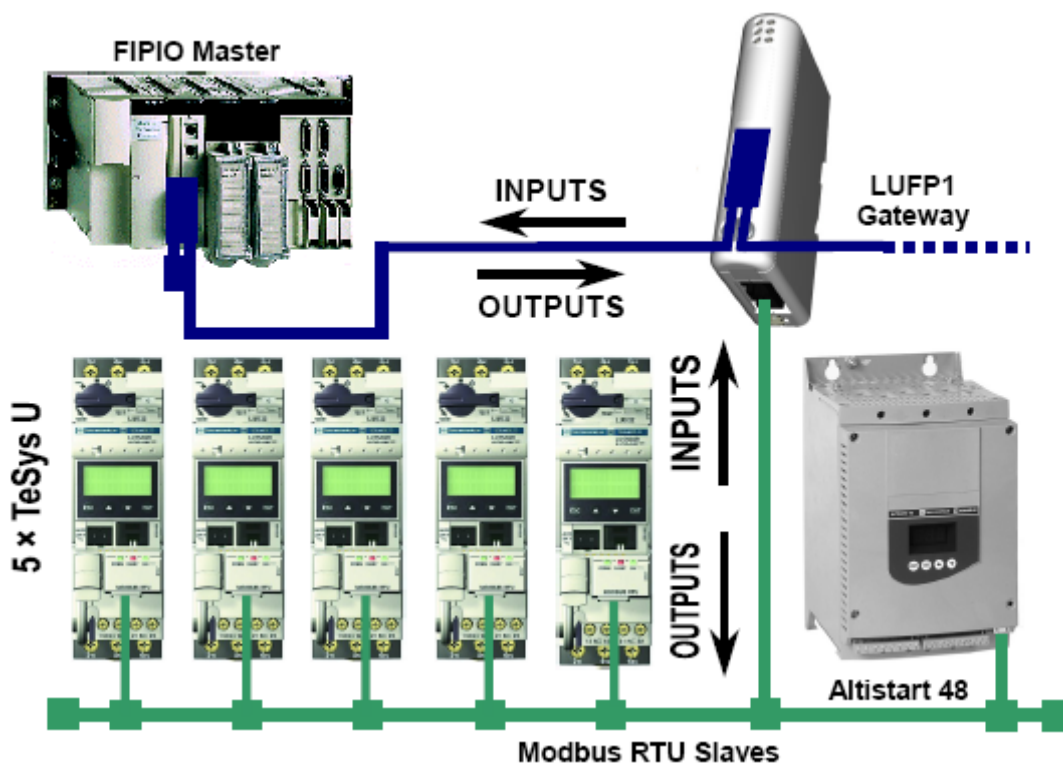
Throughout this document, the term “user” refers to any person or persons who may need to handle or use the gateway.

The term “RTU”, which refers to the Modbus RTU communication protocol, will be omitted most of the time. As a result, the simple term “Modbus” will be used to refer to the Modbus RTU communication protocol.

As it is still the case with all communication systems, the terms “input” and “output” are somewhat ambiguous. To avoid any confusion, we use a single convention throughout this document. So the notions of “input” and “output” are always as seen from the PLC, or the FIPIO master.

Hence, an “output” is a command signal sent to a Modbus slave, whereas an “input” is a monitoring signal generated by this same Modbus slave.

The diagram below shows the flows of “inputs” and “outputs” exchanged between a FIPIO master and Modbus RTU slaves via the LUF1 gateway:



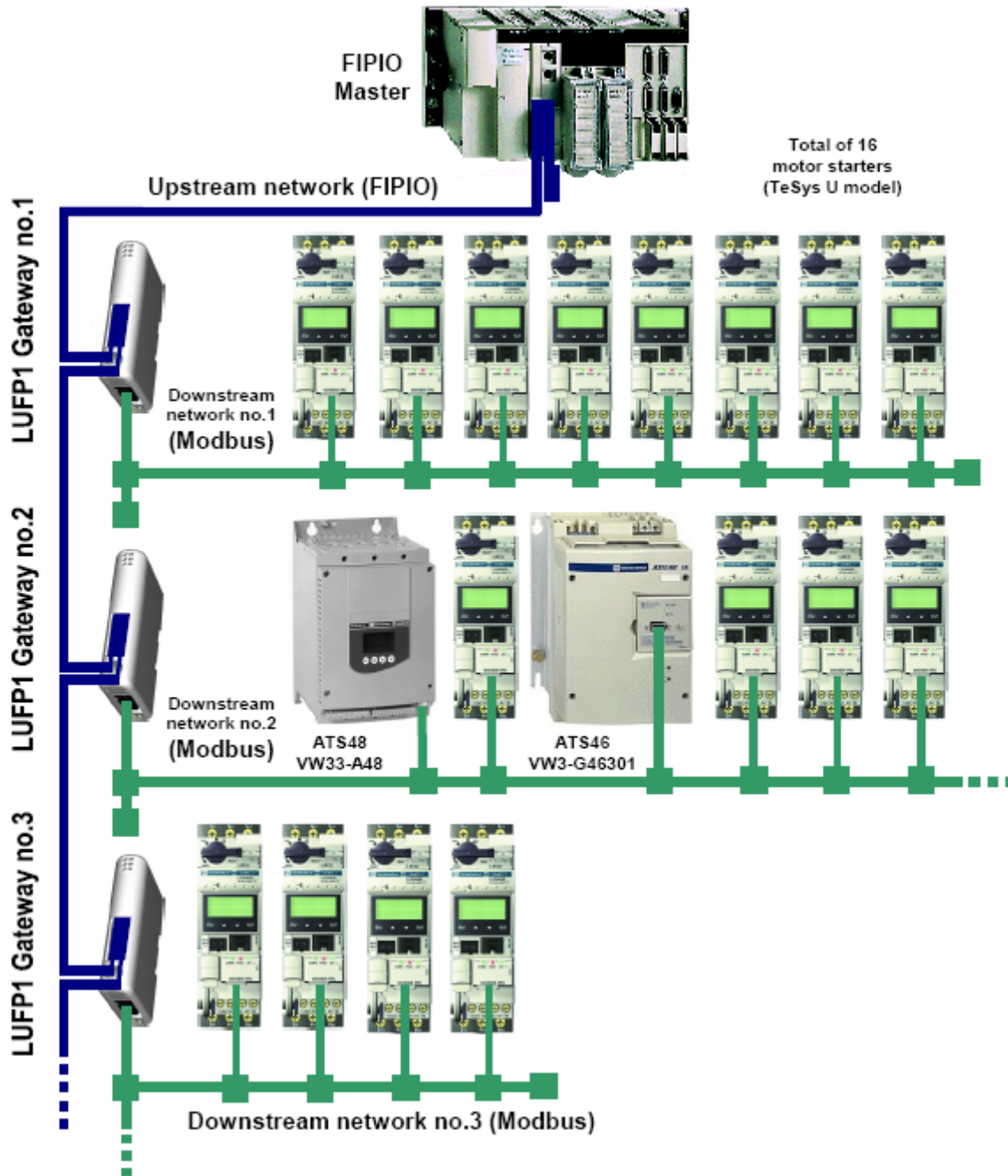
**NOTE:** For more explanation about specific terms, refer to the Glossary at the end of this guide.



# 1. Introduction

## 1.4. Introduction to the Communication “System” Architecture

Each LUF1 FIPIO / Modbus RTU gateway allows one PLC on the FIPIO network to command, control and configure up to 8 Modbus slaves. If there are more than 8 Modbus slaves, you will need to use an appropriate number of LUF1 gateways. Likewise, if the gateway needs to exchange more than 26 words with the Modbus slaves (sum of the read and write operations); the Modbus slaves will have to be shared over more than one gateway.



# 1. Introduction

---

The LUFFP1 gateway behaves both as a FIPIO slave on the upstream network and as a Modbus RTU master on the downstream network.

See 7 Using ABC-LUFFP **Config Tool** if you would like to read about the technical communication characteristics of the LUFFP1 gateway.

In the context of the Software Implementation of the Gateway, the data exchanges (input and output words) between the gateway and the Modbus slaves are all periodic. Collectively, these Modbus exchanges form the gateway's "Modbus scanner"; they are configured using PL7 PRO (by configuring and adjusting the standard profile FED C32 P). Every item of data exchanged in this way is made available to the FIPIO master, which can access it periodically and implicitly. The only aperiodic exchanges that can be performed with the LUFFP1 gateway on the FIPIO network are explicit exchanges relating to the adjustment parameters and gateway status parameters.

The gateway has a configuration and adjustment service called PKW, which offers read-write access to any Modbus slave parameter via the periodic input and output of the LUFFP1 gateways. This service is aperiodic on the Modbus network and periodic on the FIPIO network. It can be used for the following tasks:

- Retrieving or updating the parameters on some or all of the Modbus slaves;
- Reading the value of one or more data for which a low refresh rate is sufficient;
- Reading the value of a data item when an event is signalled by a periodic variable.

One of the gateway input words informs the FIPIO master of the presence or absence of each Modbus slave. The input word in question is the LAS service.

The diagram illustrates the distribution of several slaves throughout three Modbus RTU downstream networks, each one being interfaced with the FIPIO master PLC using a LUFFP1 gateway.

## 1.5. Principle of LUFFP1 Gateway Configuration and Operation

The gateway is part of a family of products (referred to as LUFFP●) designed to meet generic needs for connection between two networks using different communication protocols.

The software elements common to all these gateways (a configuration tool known as "ABC-LUFFP Config Tool" and the embedded Modbus software) cohabit with the specific features of the network upstream of each of them (FIPIO in the case of the LUFFP1 gateway) generically. This is one of the reasons why the interfacing between the upstream network and the Modbus network is carried out entirely via the gateway's physical memory. In the case of the LUFFP1 gateway, however, using a standard FIPIO profile (FED C32 P) in PL7 PRO makes it easier to configure the gateway, dispensing with the need to use ABC-LUFFP Config Tool.

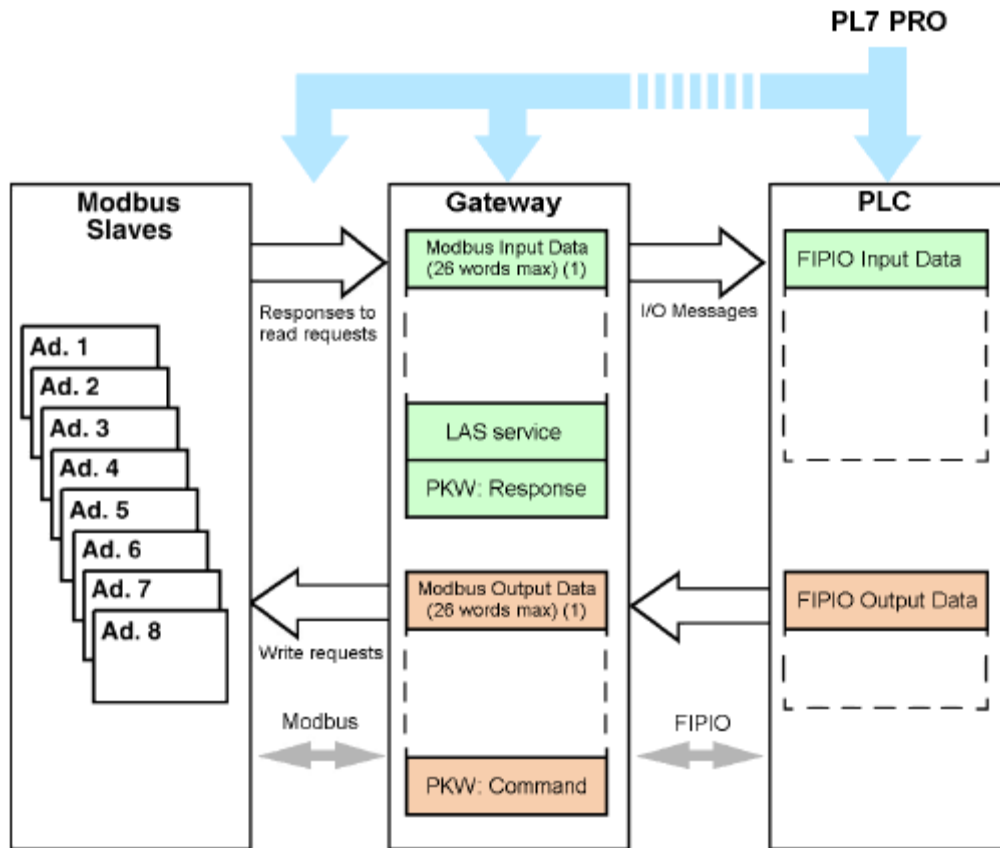
- ⇒ Exchanges between the gateway (acting as the Modbus master) and the Modbus slaves are configured by the gateway, using the configuration and adjustment parameters of profile FED C32 P entered by the user in PL7 PRO. Thanks to this configuration, the gateway creates links between part of the contents of the corresponding Modbus frames and the contents of its physical memory (input memory for the contents of the Modbus responses and output memory for the contents of the queries).
- ⇒ The periodic exchanges between the FIPIO master PLC and the LUFFP1 gateway are limited to transferring the contents of the gateway's input memory to the %IW inputs on the PLC and transferring the PLC's %QW outputs to the gateway's output memory.
- ⇒ Each LUFFP1 gateway is delivered so that it must be configured using PL7 PRO instead of ABC-LUFFP Config Tool. But this configuration is blank because it will be configured by the master PLC itself when the FIPIO network is initialised. Hence, it is absolutely necessary to use PL7 PRO to configure the FED C32 P device that corresponds to the LUFFP1 gateway (see chapter 4.2).

# 1. Introduction

The FIPIO network is totally separate from the Modbus network. The frames on a network are not directly “translated” by the gateway to generate frames on the other network. Instead, the exchanges between the contents of the gateway’s memory and the Modbus slaves make up a system independent of the one entrusted with managing the exchanges between the gateway memory and the FIPIO master.

The example which follows illustrates the independent management of each of the two networks:

— **Managing Gateway ↔ Modbus slaves exchanges (FED C32 P)** —



(1) The sum of Input Data and Output Data is limited to 26 words max.

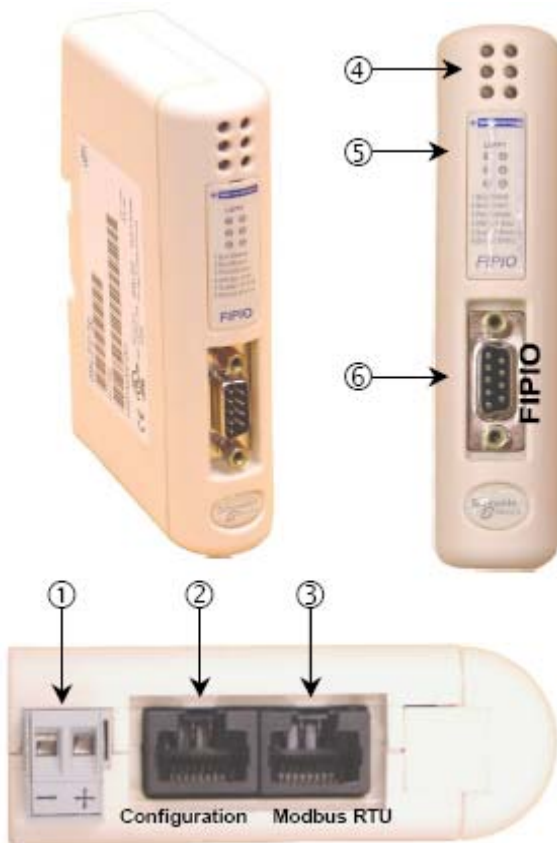
## 2. Hardware Implementation of the LUFPP1 Gateway

### 2.1. On Receipt

After opening the packaging, check that you have an LUFPP1 FIPIO / Modbus RTU gateway equipped with a detachable power connector.

### 2.2. Introduction to the LUFPP7 Gateway

The cables and other accessories for connecting to FIPIO and Modbus networks need to be ordered separately.



#### Legend:

- ① Detachable power connector for the gateway ( $\approx 24V \pm 10\%$ ).
- ② Female RJ45 connector to a PC running ABC-LUFPP Config Tool configuration software.
- ③ Female RJ45 connector for the downstream Modbus RTU network.
- ④ Six diagnostic LEDs.
- ⑤ Removable cover for the coding wheels used to configure the gateway, shown and described in chapter 2.7. The label describing the LEDs is stuck onto this cover.
- ⑥ Female FIPIO connector.

## 2. Hardware Implementation of the LUFPP1 Gateway

The LUFPP1 enables communications between a FIPIO network and Modbus devices for the purpose of industrial automation and control. As with any component used in an industrial control system, the designer must evaluate the potential hazards arising from use of the LUFPP1 in the application.

### **⚠ WARNING**

#### **LOSS OF CONTROL**

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

**Failure to follow this instruction can result in death, serious injury, or equipment damage.**

<sup>a</sup> For additional information, refer to NEMA ICS 1.1 (latest edition), "Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control" and to NEMA ICS 7.1 (latest edition), "Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems".

### 2.3. Mounting the Gateway on a DIN Rail

#### Mounting the gateway



Start by fitting the rear base of the gateway to the upper part of the rail, pushing downwards (1) to compress the gateway's spring. Then push the gateway against the DIN rail (2) until the base of the gateway box fits onto the rail.

#### Removing the gateway



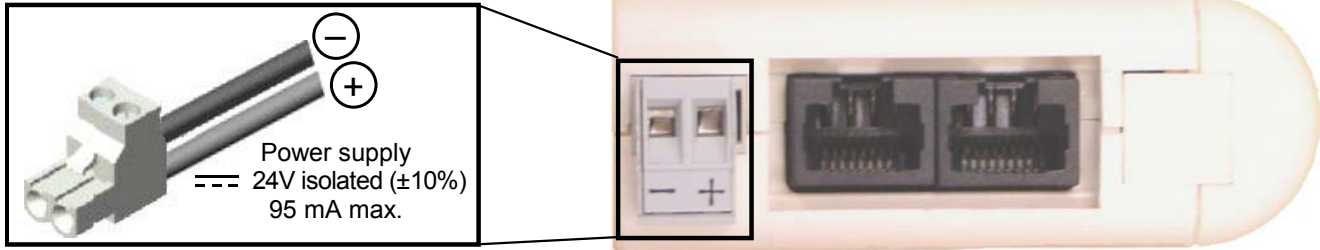
Start by pushing the gateway downwards (1) to compress the gateway's spring. Then pull the bottom of the gateway box forwards (2) until the box comes away from the rail.

**NOTE:** The spring is also used to ground the gateway (Protective Earth).

## 2. Hardware Implementation of the LUFPP1 Gateway

### 2.4. Powering the Gateway

FIPIO / Modbus RTU gateway – View from underneath



### **⚠ WARNING**

#### **RISK OF UNINTENDED EQUIPMENT OPERATION**

Do not use the 24 VDC power available from the FIPIO network cabling to operate the LUFPP Gateways, as the negative terminal (—) of this power is not necessarily at the installation earth ground potential. Use of an ungrounded power supply may cause the LUFPP devices to operate in an unexpected manner.

To ensure reliable operation, the LUFPP Gateways require a separate power supply where the negative terminal (—) is connected to the installation earth ground.

**Failure to follow this instruction can result in death, serious injury, or equipment damage.**

Recommendations:

- Use 60/75 or 75xC copper (CU) wire only.
- The terminal tightening torque must be between 5-7 lbs-in (0.5-0.8 Nm).

### 2.5. Connecting the Gateway to the Modbus Network

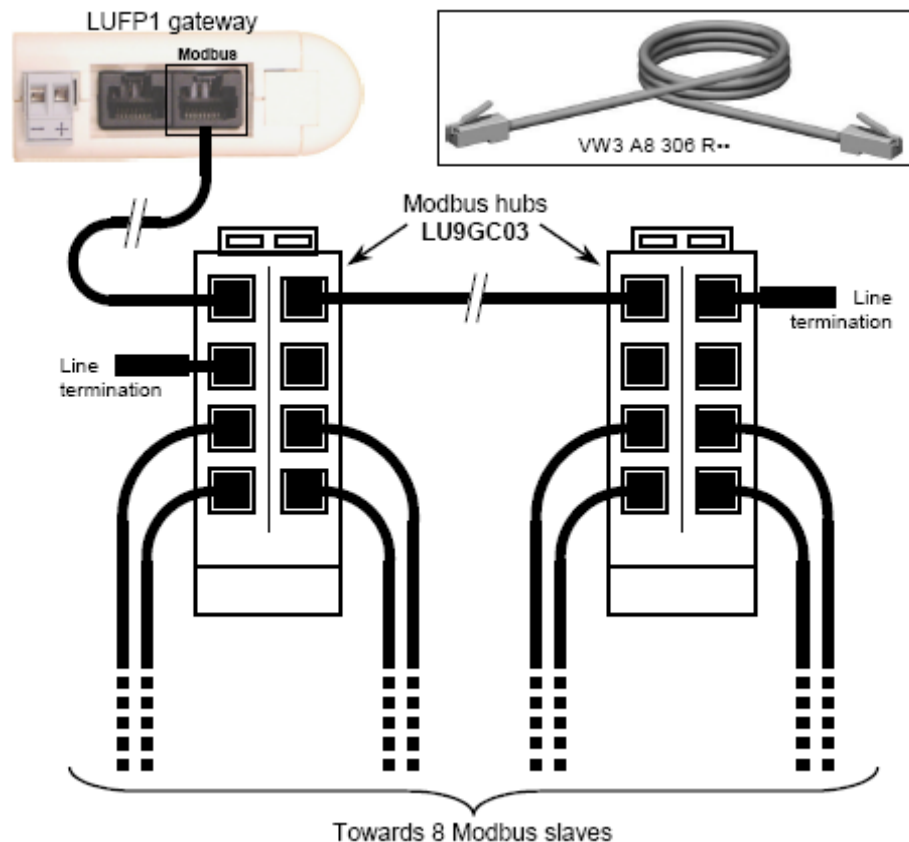
Three typical examples of Modbus connection for the gateway and its slaves are shown below. There are many other possible Modbus connections, but they are not covered in this document.

## 2. Hardware Implementation of the LUF P1 Gateway

### 2.5.1. Examples of Modbus Connection

- **“Star” topology:** This topology uses LU9GC03 Modbus hubs, which have 8 female RJ45 connectors. These hubs should be placed close to the Modbus slaves to which they are connected using VW3 A8 306 R\*\* cables. On the other hand, the nature of the cable connecting the LUF P1 gateway to one of these hubs will depend on the network architecture, so long as there is a male RJ45 connector at each end. If necessary, one or two line terminations may be directly connected to the hubs.

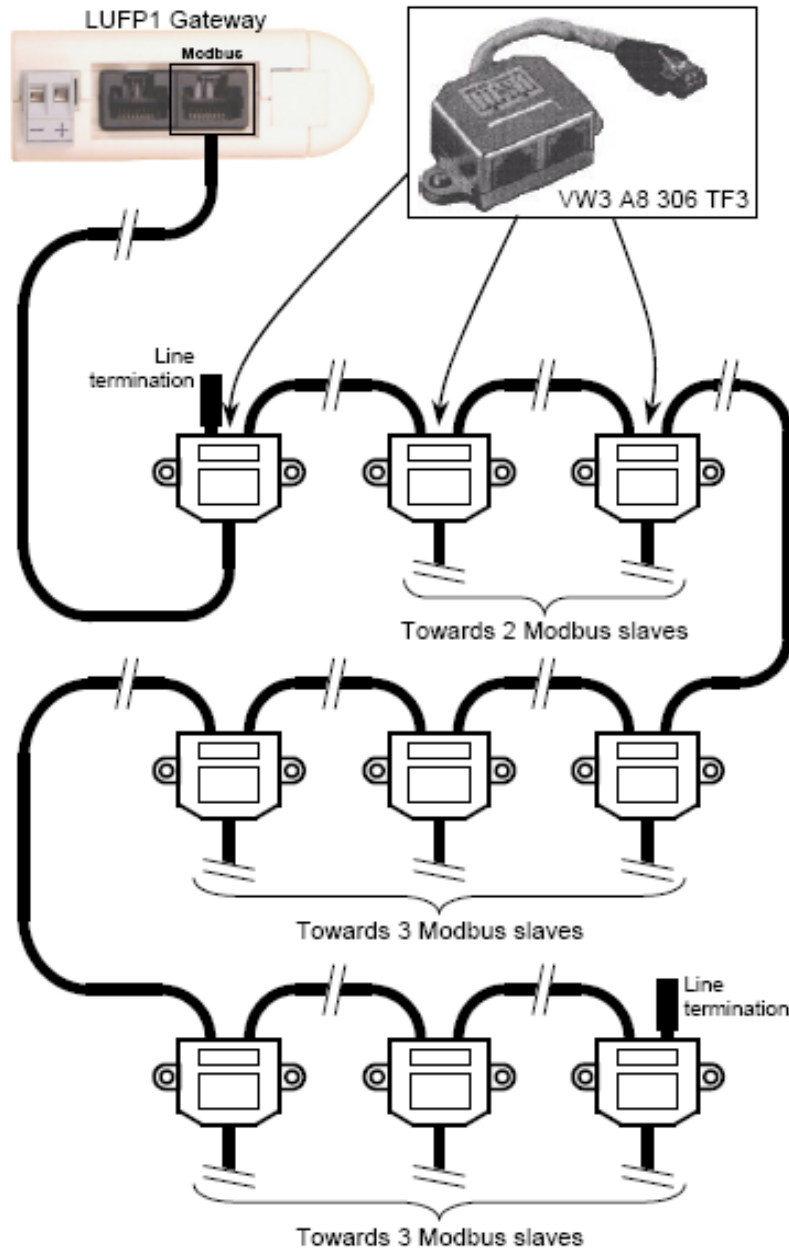
The connections are shown below:



## 2. Hardware Implementation of the LUF1 Gateway

• **“Bus” topology with VW3 A8 306 TF3 drop boxes:** This topology uses VW3 A8 306 TF3 drop boxes to connect each of the Modbus slaves to the main section of the Modbus network. Each box should be placed in the immediate vicinity of the Modbus slave it is associated with. The cable for the main section of the Modbus network must have male RJ45 connectors (like the VW3 A8 306 R cable used for the “star” topology). The lead between the drop box and the slave or the Modbus gateway is an integral part of this box.

The connections are shown below:

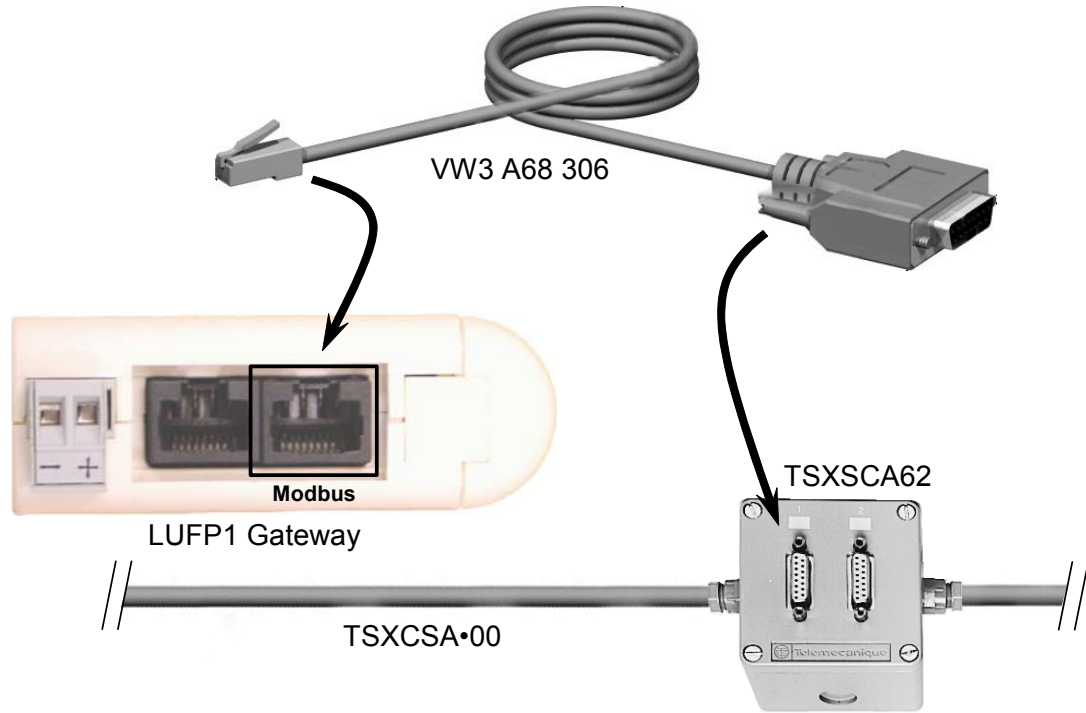




## 2. Hardware Implementation of the LUFPP1 Gateway

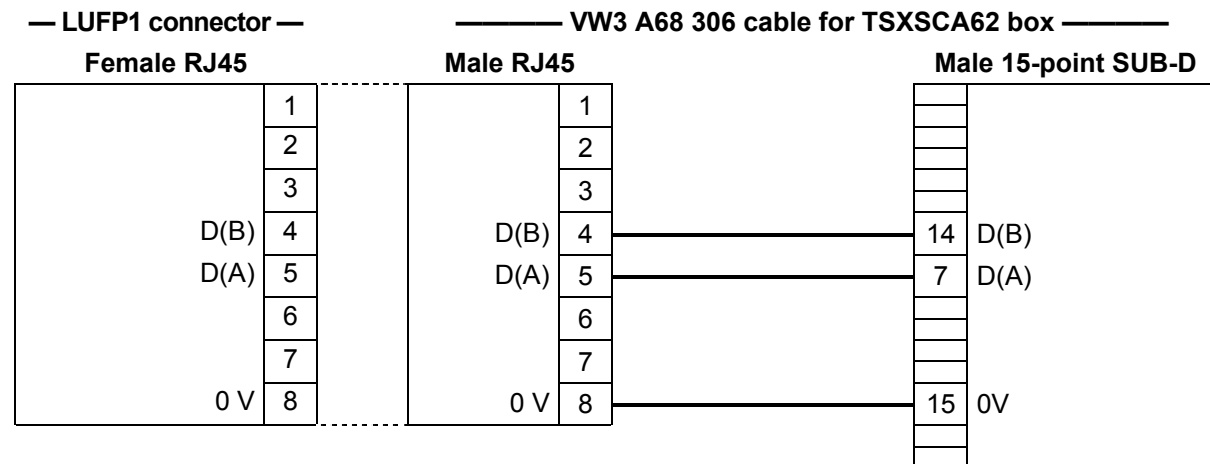
**“Bus” topology with tap boxes:** This topology is similar to the previous one, except that it uses TSXSCA62 subscriber connectors and / or TSXCA50 subscriber connectors. We recommend using a VW3 A68 306 connection cable and the TSXCSA•00 Modbus cables. Connect the RJ45 connector on the VW3 A68 306 cable to the Modbus connector on the LUFPP1 gateway.

The connections are shown below:



### 2.5.2. Pin Outs

In addition to the pin out for the connector on the gateway, the one on the VW3 A68 306 cable is also shown below, as it is the only Modbus cable which does not exclusively use RJ45 connections.



## 2. Hardware Implementation of the LUF1 Gateway

### 2.5.3. Wiring Recommendations for the Modbus Network

- Use a shielded cable with 2 pairs of twisted conductors,
- connect the reference potentials to one another,
- maximum length of line: 1,000 meters (3280 ft)
- maximum length of drop line / tap-off: 20 meters (65.5 ft)
- do not connect more than 9 stations to a bus (slaves and one LUF1 gateway),

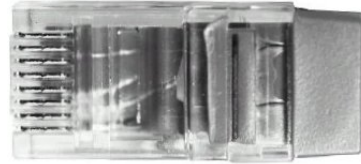
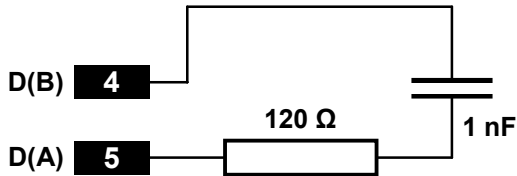
### WARNING

#### RISK OF UNINTENDED EQUIPMENT OPERATION

Do not connect more than 9 stations to the Modbus fieldbus (gateway and 8 slaves). While the gateway may appear to operate correctly with more than 9 devices, it is likely one or more devices will only communicate intermittently, leading to unpredictable system behavior.

**Failure to follow this instruction can result in death, serious injury, or equipment damage.**

- cable routing: keep the bus cable away from power cables (at least 30 cm (0.98 ft)), make crossings at right angles if necessary, and connect the cable shielding to the earth on each unit,
- adapt the line at both ends using a line terminator (see diagram and VW3 A8 306 RC termination below).



— Line termination recommended at both ends of the line —

— VW3 A8 306 RC line termination —

### WARNING

#### MODBUS TERMINATION USING THE RESISTANCE-ONLY METHOD

Use only RC (Resistance-Capacitance) Modbus cable terminations with the LUF1 Gateway. The LUF1 gateways are designed to support client equipment that will not function correctly without using RC-type Modbus cable termination.

**Failure to follow this instruction can result in death, serious injury, or equipment damage.**

## 2. Hardware Implementation of the LUF1 Gateway

---

To make it easier to connect the units using the topologies described in chapter 2.5.1, various accessories are available in the *Schneider Electric* catalogue:

### 1) Hubs, junctions, and line terminations:

- LU9GC03 hub ..... This passive box has 8 female RJ45 connectors. Each of these connectors can be connected to a Modbus slave, to a Modbus master, to another Modbus hub, or to a line termination.  
("bus" topology with LU9GC03 splitter boxes)
  
- VW3 A8 306 TF3 T-junction box ..... This passive box includes a short lead with a male RJ45 connector allowing it to be connected directly to a Modbus slave, without having to use a different cable. It is fitted with 2 female RJ45 connectors for the connection of two Modbus cables of the VW3 A8 306 R•• type.  
("bus" topology with VW3 A8 306 TF3 T-junction boxes)
  
- 2-way TSXSCA62 subscriber connector ..... This passive box has a printed circuit fitted with screw terminals and allows the connection of 2 subscribers to the bus (2 female 15 point SUB-D connectors). It includes the line termination when the connector is located at the end. It is fitted with 2 screw terminals for the connection of two double twisted pair Modbus cables.  
("bus" topology with SCA junction boxes)
  
- TSXCA50 SCA junction box ..... This passive box allows a Modbus unit to be connected to a screw terminal. It includes the line termination when the connector is located at the end. It is fitted with 2 screw terminals for the connection of two double twisted pair Modbus cables.  
("bus" topology with SCA junction boxes)
  
- VW3 A8 306 RC double termination ..... Each of these two red passive boxes is a male RJ45 connector 3 cm (1.18 in) long containing an RC line termination (see diagram and illustration above). Only the abbreviation "RC" is shown on these boxes.  
(all topologies)

### 2) Cables:

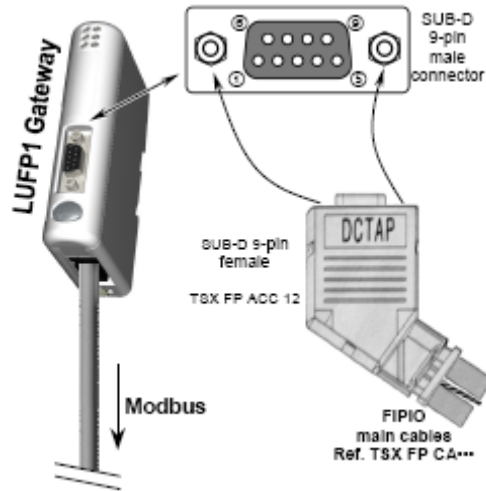
- VW3 A8 306 R•• Modbus cable ..... Shielded cable with a male RJ45 connector at each end.  
("bus" topology with SCA junction boxes)
  
- VW3 A68 306 Modbus cable ..... Shielded cable with a male RJ45 connector and a male 15 point SUB-D connector. It is used to connect a Modbus subscriber (slave or master) to a TSXSCA62 or TSXCA50 box.  
("bus" topology with SCA junction boxes)
  
- Shielded double twisted pair Modbus cable ..... Bare cable (without connectors) used to make up the main section of the Modbus network. There are three items available: TSXCSA100 (100 m or 328 ft), TSXCSA200 (200 m or 656 ft), and TSXCSA500 (500 m or 1640 ft).  
("bus" topology with branch boxes)

# 2. Hardware Implementation of the LUF1 Gateway

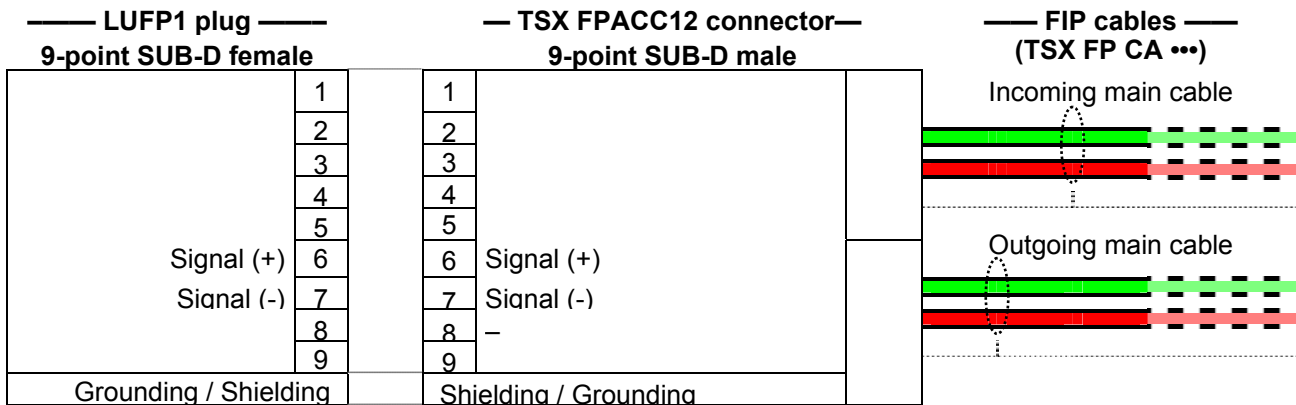
## 2.6. Connecting the LUF1 gateway to the Profibus-DP Network

Connect the SUB-D 9-point male plug on the FIPIO connector to the FIPIO plug on the LUF1 gateway.

Connections are illustrated here:



### 2.6.1. Pin Outs



**NOTE:** If the two main cables are replaced by a TSX FP CC ... tap-off cable, the **red** and **orange** wires correspond to the signal (+), while the **green** and **black** wires correspond to the signal (-).

## 2. Hardware Implementation of the LUF1 Gateway

---

### 2.6.2. Wiring Recommendations for the FIPIO Network

- Use TSX FP ACC 12 connectors (or TSX FP ACC 2 in the case of TSX 17-20 micro-PLCs) to connect the gateway to the FIPIO network.
- Use a TSX FP CA ••• main FIPIO cable to link up the connectors and a TSX FP CC ••• FIPIO tap-off cable if you need to connect up to a tap box (TSX FP ACC 14, TSX FP ACC 3, TSX FP ACC 4 or TSX FP ACC 10). Both of these shielded cables are available in lengths of 100, 200 or 500 meters.
- Use two line terminations to close off the FIPIO network at both ends of each electrical segment. The TSX FP ACC 7 kit includes two FIP terminations.
- Connect the reference potentials to one another.
- The transmission rate is fixed at 1 Mbit / s.
- Use TSX FP ACC 6 repeaters to connect up two electrical segments on the same FIPIO network. The maximum length of an electrical segment is 1,000 m. Follow this rule:  
$$[ (Number\ of\ repeaters \div 2) + Sum\ of\ lengths\ (in\ km) ] < 22$$
The longest FIPIO network that can be implemented under this rule would be 15 km long and would require 14 repeaters ( $14 \div 2 + 15 = 22$ ).
- Do not connect more than 32 master or slave stations per segment (not counting repeaters).
- Cable routing: keep the bus away from power cables (at least 30 cm), make crossings at right angles if necessary and connect the cable shielding to the earth on each unit.

For more details about connections, see the **FIPIO Bus / FIPWAY Network Reference Manual** (ref. TSX DR FIP F for the French version, TSX DR FIP E for the English version, TSX DR FIP G for the German version and TSX DR FIP S for the Spanish version). This manual also contains details on the operating characteristics and implementation of a FIPIO field bus.

In addition, the manual **Electromagnetic Compatibility of Industrial Field Buses and Networks** (ref. TSX DGKBL F for the French version, ref. TSX DGKBL E for the English version and ref. TSX DGKBL G for the German version) contains valuable rules and precautionary measures for wiring up a FIPIO field bus.

A number of accessories are available from the *Schneider Electric* catalogue to facilitate the connection of stations on a FIPIO network:

#### 1) Connectors, tap boxes and line terminations:

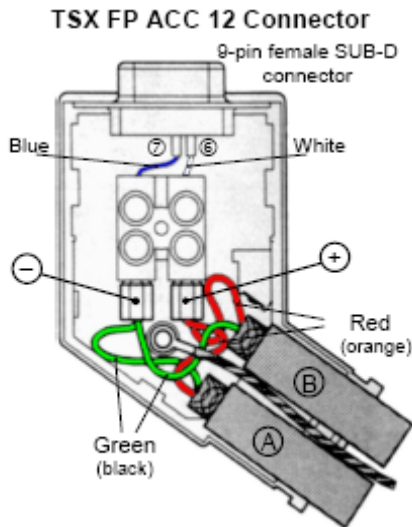
- TSX FP ACC 12 connector ..... Isolating connector for equipment with a 9-pin female SUB-D connector. Used for daisy-chain or parallel cabling (see illustration on next page).
- TSX FP ACC 2 connector ..... Female connector for the TSX FPG 10• coupler (TSX 17-20 micro-PLCs). Used for daisy-chain or parallel cabling.
- TSX FP ACC 14 box ..... Isolating tap box for bus connections to the main cable.
- TSX FP ACC 4 box .....  
TSX FP ACC 10 box ..... Watertight tap boxes for bus connections to the main cable. They also support one 9-pin female SUB-D connector (for the PCMCIA card cable TSX FP CG 010 / 030).
- TSX FP ACC 3 box ..... Isolated tap box for bus connections to the main cable. Also supports two 9-pin female SUB-D connectors (*as above*).
- TSX FP ACC 7 terminations ..... Set of two line terminations for adapting the impedance of both ends of an electrical segment. You will need one set per electrical segment.
- TSX FP ACC 6 repeater..... Electrical repeater for joining two segments, each with a maximum length of 1,000 m.

## 2. Hardware Implementation of the LUF1 Gateway

### 2) Cables:

- TSX FP CA... main cable.....Shielded twisted 150  $\Omega$  cable with a single pair of conductors: red (+) and green (-). In lengths of 100, 200 or 500 m. This cable is the main trunk of the FIPIO network and is therefore used to link up connectors and tap boxes.
- TSX FP CC... tap link cable.....Shielded twisted 150  $\Omega$  cable with two pairs of conductors: red or orange (+), and green or black (-). In lengths of 100, 200 or 500 m. This cable is used to connect a tap box to a distant connector.

**NOTE:** When calculating the length of a FIPIO segment, remember to multiply the length of the cable by two.



The illustration opposite shows the wiring in the TSX FP ACC 12 connector when connecting up to the main cable (two sections of TSX FP CA... cable).

Replace one of the two cables by one of the two lineterminations supplied in the TSX FP ACC 7 kit, if the connector is at the end of an electrical segment.

Replace the two cables (A) and (B) by a TSX FP CC... cable (in (A)) if the connection point is a derivation rather than a daisy-chain link; if so, the **orange** wire corresponds to the (+) signal of the second pair of conductors and the **black** wire corresponds to the (-) signal.

### 2.6.3. FIPIO Bus Commissioning Procedure

We recommend commissioning the devices in sequence, one after the other. For a detailed description of the initial commissioning of an application on FIPIO, see the *FIPIO Bus / FIPWAY Network Reference Manual*.

## 2. Hardware Implementation of the LUFP1 Gateway

### 2.7. Configuring the FIPIO Communication Features

This configuration must be carried out when the gateway is powered off.

#### CAUTION

##### OPENING LUFP• COVER WITH POWER ON

The power supply of the gateway must be turned off before opening the cover. Once the cover has been removed, make sure you touch neither the electrical circuits nor the electronic components, as this may damage the device.

**Failure to follow this instruction can result in injury or equipment damage.**

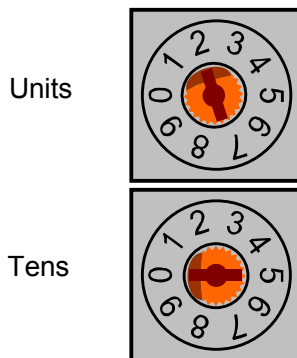
This task is limited to configuring the gateway's FIPIO address, as the communication speed on the FIPIO network is fixed at 1 Mbit/s.

The two coding wheels used for configuring the gateway's address are hidden behind the gateway cover ⑤ (see illustration in chapter 2.2). To remove this cover, insert the end of a small flat screwdriver between the top of the hood and the gateway box and pull it out.

#### 2.7.1. Encoding the Gateway Address

The LUFP1 gateway is identified on the Profibus-DP bus by its address, ranging from 1 to 99.

On FIPIO, address 0 is reserved for the PLC that manages the bus. Address 63 is reserved for the programming or operation terminal. All the other addresses, up to physical address 127, can be used.



The gateway's FIPIO address depends on the position of the two coding wheels represented on the left, in their factory-setting positions (default address = 2).

This address is the sum of the decimal values given by the angular positions of the bottom coding wheel (tens) and the top coding wheel (units).

## 2. Hardware Implementation of the LUF1 Gateway

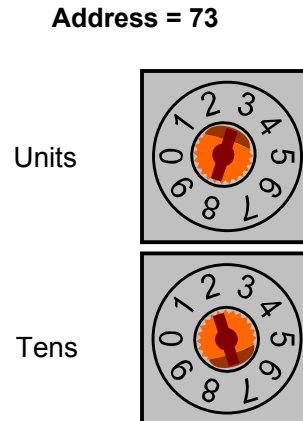
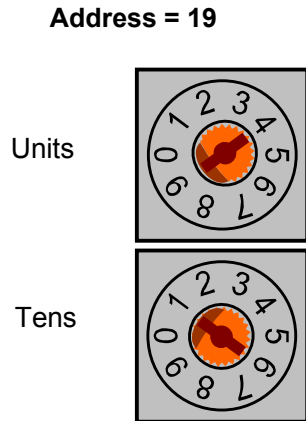
---

Any change to the gateway's address shall be taken into account only once it is powered on again.

**NOTE:**

- No two devices on the same FIPIO bus can ever have the same address. Simultaneous, fast (2 Hz) and continuing flashing of LEDs **1**, **2**, **3** and **4** means that the device cannot connect to the FIPIO bus because the address is already used by another device.
- This flashing signal also occurs when address 00 is configured using the gateway's coding wheels.

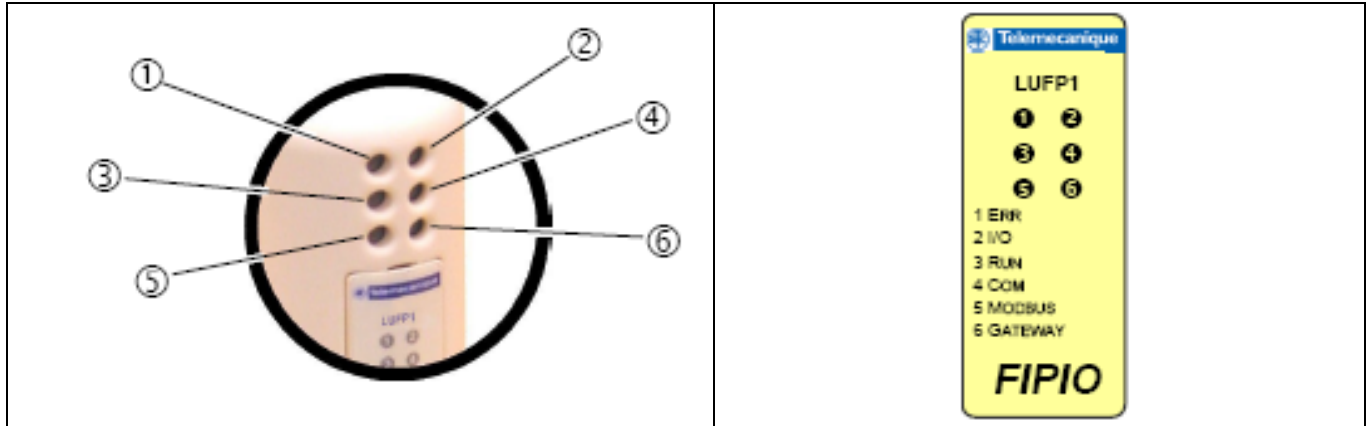
**Examples:**





### 3. Signaling

The gateway's 6 LEDs and the descriptive label on the removable cover which hides its two coding wheels (gateway address) allow you to diagnose the status of the gateway:



LED	LED → Gateway state
❶ ERR	Off: Gateway on-line (exchanges possible)
	Flashing (red): Gateway off-line (exchanges impossible) ↳ Check, in PL7 PRO, that a FIPIO station has been declared for the gateway address. Check that the gateway is correctly connected to the FIPIO network.
❸ RUN	Off: No power, or critical fault
	Flashing green: Duplicate address (see Note, below) Green: Gateway operational
❺ MODBUS	Off: No power
	Flashing (green): No Modbus communications
	Green: Modbus communications OK with all Modbus slaves
	Red: Loss of communication with one or more Modbus slaves

LED	LED → Gateway state
❷ I / O	Off: No fault
	Red: Internal fault ↳ Replace the gateway
	Flashing (red): Timing failure ↳ Fault mainly due to FIPIO network and / or FIPIO master
❹ COM	Off: Total absence of communication with bus arbitrator
	Flashing yellow (5 Hz): Exchanges in progress with bus arbitrator
❻ GATEWAY	Off: No power
	Green: Initialization and configuration of the gateway
	Flashing (green): Gateway in operation
	Flashing (red / green): Configuration absent / invalid ↳ Use ABC-LUFPI Config Tool to load a valid configuration

**NOTE:**

- If LED ❸ does not flash from red to green and if LEDs ❶, ❷, ❸ and ❹ all flash at the same time at a frequency of 2 Hz, this means that the physical gateway address is already used by another FIPIO device, or that it has been set at 0 with the gateway's coding wheels.  
Try to resolve this address conflict by viewing the FIPIO network configuration in PL7 PRO. If necessary, modify the addresses in the configuration. Also check the physical address configured on the gateway using the coding wheels (see Section 2.7.1).
- If both identically-addressed devices are reconnected to FIPIO network simultaneously, then the four LEDs mentioned above will not flash. In this case, only LED ❶ will flash, and then only discontinuously and in an irregular manner. This happens, for example, when you disconnect and then reconnect the bus arbitrator.

# 4. Software Implementation of the Gateway

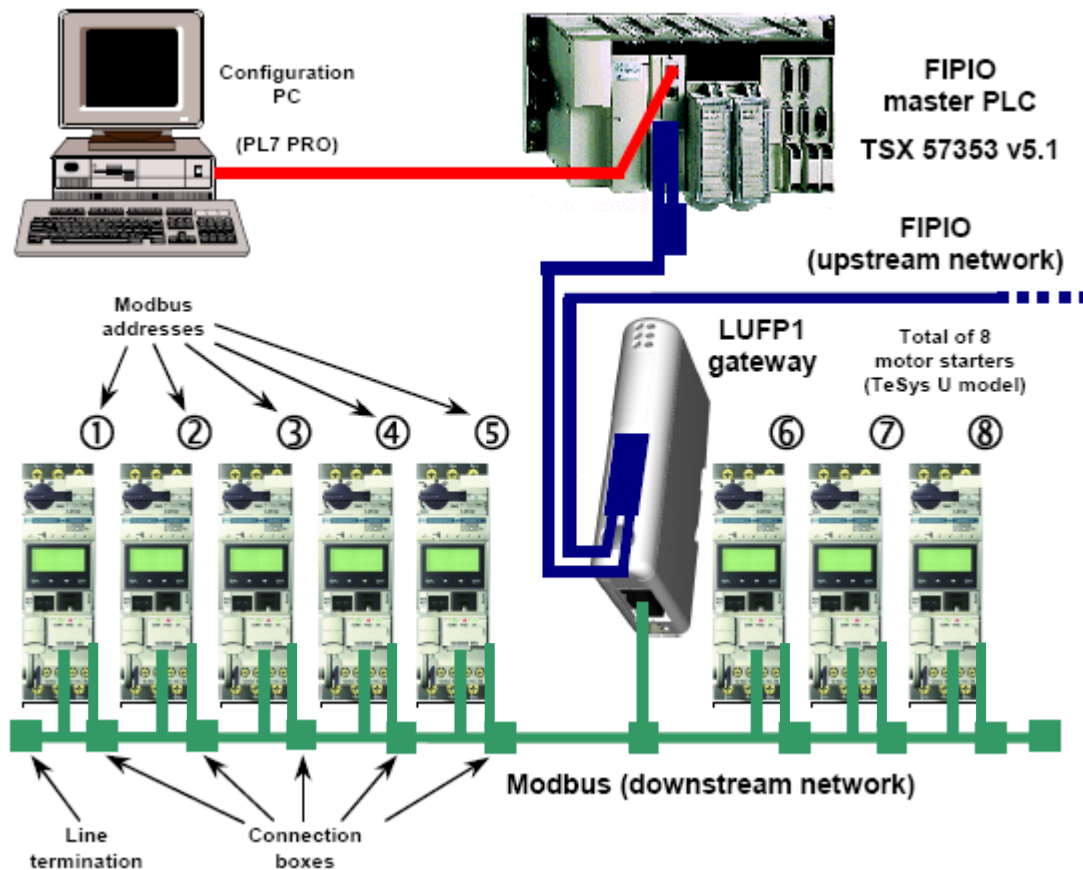
## 4.1. Introduction

This chapter presents the implementation of the LUF1 gateway in PL7 PRO, with a Schneider Electric “Premium” PLC (e.g. PLC TSX 57353 v5.1) acting as master and FIPIO bus arbitrator via the FIPIO link built into its processor card.

**NOTE:** The various configuration possibilities for Modbus exchanges on the gateway are not covered in this section, as the aim here is to describe a standard configuration with 8 TeSys U motor starters. This configuration is described with a view to implementing the gateway using PL7 PRO. The full range of gateway configuration possibilities is set out in Appendix B

### 4.1.1. System Architecture

The configuration described in this LUF1 gateway implementation enables the device to command and control eight TeSys U motor starters:



See Chapter 2 for details on implementing the hardware for the standard configuration.

## 4. Software Implementation of the Gateway

### 4.1.2. Configuring the Motor Starters

Each motor starter should be configured as follows:

Protocol:	Modbus RTU slave	Start bits	1
Modbus address	1 to 8	Parity	None
Bitrate	19,200 bits/s	Parity bit	0
Data bits	8	Stop bits	1

When using a TeSys U motor starter with a Modbus communication module (LULC03● module), the configuration parameters for the RS485 connection are automatically detected, only the Modbus address of the motor starter needs to be configured.

**NOTE:** The gateway's Modbus slaves cannot be configured with PL7 PRO; the configuration and adjustment of the LUF1 gateway are limited to the gateway and to the management of generic Modbus exchanges.

### 4.1.3. Modbus Cycle Time

The LUF1 gateway's default configuration sets a cycle time of 300 ms on Modbus commands for each of the 8 TeSys U motor starters.

### 4.1.4. Managing Degraded Modes With the Gateway Default Configuration

The degraded-mode management of the LUF1 gateway standard configuration is described below. In this instance, we are using a Premium PLC with a FIPIO link built into the processor card.

Observed behavior		Event			
		Premium PLC: CPU stop or failure	Disconnection of the upstream FIPIO network	Failure of the LUF1 gateway	Disconnection of the downstream Modbus network
Outputs	Reset	Yes		Depending on the configuration of the TeSys U motor starters (1)	
	Hold	---			
	Stop refreshing			---	
Inputs	Reset	---	---	Yes (2)	
	Hold	---	Yes (3)	---	

(1) The desired behavior with regard to outputs should be directly configured on each TeSys U motor starter.

(2) Use the List of Active Slaves (LAS) Service described in Section 5.2.

(3) Use the FIPIO Diagnostic Objects (Section 5.4) to detect the underlying event.

Please refer to Appendix B if you would like to change the way that degraded modes for **all** Modbus commands are managed.

#### 4.1.4.1. Description of the Gateway Degraded Mode Options

**%MW\p.2.c\0.0.23 (bits 0-7): Effects on Modbus exchanges...**

This parameter affects the data sent to **all** Modbus slaves if there is no communication from the FIPIO master.

## 4. Software Implementation of the Gateway

---

This parameter can take 3 values:

- 0x00 ..... Freeze: All data sent retains its current values.
- 0x01 ..... No scanning: No more Modbus queries are transmitted.
- 0x02 ..... Clear: All data sent to the Modbus slaves is reset to 0.

The default behavior for the gateway's standard configuration described here is "Clear". Hence, the TeSys U Command (704) and Status (455) registers continue to be refreshed, but:

- The corresponding gateway's output memory (TeSys U Command registers) is forced to 0,
- And the corresponding gateway's input memory (TeSys U Status registers) is normally refreshed.

### **%MW\p.2.c\0.0.21 (bits 0-7): Modbus timeout time (unit: 100 ms)**

This parameter defines the time the gateway will wait for a response before it either retries sending the same request, or it disconnects the slave and declares it missing.

The same Modbus timeout time is applied to *all* Modbus commands.

In the gateway's standard configuration, described here, this time is equal to 300 ms (3 × 100 ms).

### **%MW\p.2.c\0.0.21 (bits 8-15): Retries**

This parameter determines the number of re-transmissions carried out by the gateway if there is no response from a Modbus slave.

The same number of retries is used for *all* Modbus commands.

In the gateway's standard configuration, described here, this parameter is set to 3.

### **%MW\p.2.c\0.0.22 (bits 0-7): Reconnection time (unit: 1 s)**

This parameter defines the amount of time the gateway will wait before it again tries to communicate with a Modbus slave that was previously declared missing.

The same reconnection time is applied to *all* Modbus slaves.

In the gateway's standard configuration, described here, this time is equal to 10 s (10 × 1 s).

## **WARNING**

### **RISK OF UNINTENDED EQUIPMENT OPERATION**

During the reconnect time, you cannot control a slave (read/write) via the bus. Depending on the slave characteristics and the watchdog configuration, the slave can keep the same status or take a fallback position.

To avoid an unintended equipment operation, you must know the possible status of a slave and adapt the timeout and reconnect time values according to the request sending rate.

**Failure to follow this instruction can result in death, serious injury, or equipment damage.**

### **%MW\p.2.c\0.0.23 (bits 8-15): Effects on FIPIO exchanges...**

This parameter affects the data sent to the FIPIO master if there is no response from a slave or if a slave is absent. It is defined for *all* Modbus slaves, but only the data related to an absent slave is affected.

This parameter can have 2 values:

- 0x00 ..... No action: All data sent to the FIPIO master retains its current values.
- 0x02 ..... Clear: All data sent to the FIPIO master are reset to 0.

The default behavior for the gateway's standard configuration described here is "Clear". Hence, any TeSys U status register is forced to 0 if the corresponding TeSys U motor starter is missing.

#### **4.1.4.2. Degraded Mode Description**

This description takes into account the following elements, but only applies to the standard configuration described here:

- The PLC processor
- The FIPIO master

## 4. Software Implementation of the Gateway

---

- The LUFP1 gateway
- The Tesys U starters-controllers.

### PLC processor stopped or on failure

<b>PLC processor response</b>
<p>Outputs:</p> <p>Software error: outputs reset to default state or hold their present state depending on configuration.</p> <p>Hardware error: (EEPROM or hardware failure), output state will be undetermined.</p> <p>Inputs:</p> <p>PLC stops responding to inputs in any error state.</p>
<b>FIPIO master response</b>
<p>Depending on FIPIO master configuration:</p> <p>the FIPIO master stops to communicate with the LUFP1 gateway, or</p> <p>the FIPIO master forces outputs to 0, and refreshes inputs, or</p> <p>the FIPIO master holds outputs in their last position, and refreshes inputs.</p>
<b>LUFP1 gateway response</b>
<p>If the FIPIO master stops communicating with the LUFP1 gateway (case of a Premium PLC), the periodic Modbus exchanges continue to run:</p> <p>all output data is reset to 0 and sent to the Modbus slaves,</p> <p>the input memory continues to be refreshed with the Modbus slaves' response data.</p>
<p>If the FIPIO master forces outputs to 0 and refreshes the inputs, the periodic Modbus exchanges continue to run:</p> <p>all output data is reset to 0 and sent to the Modbus slaves,</p> <p>the input memory continues to be refreshed with the Modbus slaves' response data..</p>
<p>If the FIPIO master forces holds outputs in their last state and refreshes the inputs, the periodic Modbus exchanges continue to run:</p> <p>all output data is reset to 0 and sent to the Modbus slaves,</p> <p>the input memory continues to be refreshed with the Modbus slaves' response data.</p>
<b>Tesys U response</b>
<p>If the FIPIO master stops communicating with the LUFP1 gateway (such as in the case of a Premium PLC) or forces outputs to 0, the periodic Modbus exchanges continue to run:</p> <p>Command registers (704) are forced to 0 and the motors are stopped,</p> <p>Status registers (455) are transmitted to the gateway.</p>
<p>If the FIPIO master keeps outputs in their last state, the periodic Modbus exchanges continue to run:</p> <p>Command registers (704) keep their last values and the motors rest in the same state,</p> <p>Status registers (455) are transmitted to the gateway.</p>

### FIPIO master stopped or on failure

<b>PLC processor response</b>
<p>The PLC processor provides some error and/or diagnostic objects to the application in case of FIPIO master stop or failure (input/output not valid).</p> <p>Refer to the PLC user manual to have their description.</p> <p>This information must be managed in the PLC application.</p>
<b>FIPIO master response</b>
<p>If the Profibus master is stopped (command coming from the application):</p> <p>the master stops to communicate with the LUFP1 gateway.</p>
<b>LUFP1 gateway response</b>
<p>Periodic Modbus exchanges continue to run:</p> <p>all output data is reset to 0 and sent to the Modbus slaves,</p>

## 4. Software Implementation of the Gateway

---

the input memory continues to be refreshed with the Modbus slaves' response data.

### **Tesys U response**

Periodic Modbus exchanges continue to run:  
Command registers (704) are forced to 0 and the motors are stopped,  
Status registers (455) are transmitted to the gateway.

### **LUFP1 gateways disconnected on FIPIO side**

#### **PLC processor response**

The PLC processor provides some error and diagnostic objects coming from the FIPIO master in case of slave disconnection from the application:  
Refer to the PLC user manual to have their description.  
This information must be managed in the PLC application.

#### **FIPIO master response**

The FIPIO master provides the processor with some error and diagnostic objects in case of FIPIO slave disconnection.

#### **LUFP1 gateway response**

Periodic Modbus exchanges continue to run:  
all output data is reset to 0 and sent to the Modbus slaves,  
the input memory continues to be refreshed with the data of the Modbus slaves' responses.

### **Tesys U response**

Periodic Modbus exchanges continue to run:  
Command registers (704) are forced to 0 and the motors are stopped,  
Status registers (455) are transmitted to the gateway.

### **LUFP1 gateways failure**

#### **PLC processor response**

The PLC processor provides some error and diagnostic objects coming from the FIPIO master in case of slave failure to the application.  
Refer to the PLC user manual to have their description.  
This information must be managed in the PLC application

#### **FIPIO master response**

The FIPIO master provides the processor with some error and diagnostic objects in case of FIPIO slave failure.

#### **LUFP1 gateway response**

In case of a failure, the gateway stops to communicate with the FIPIO master and the Modbus slaves.

### **Tesys U response**

Depending on the Tesys U configuration:  
If the starters-controllers do not receive any requests, they will:  
stop the motor,  
keep the same state, or  
run the motor.  
Refer to the Tesys U user manuals to adjust these fallback positions.

## 4. Software Implementation of the Gateway

---

### LUF1 gateways disconnected on Modbus side or Tesys U failure

<b>PLC processor response</b>
The processor has access to the List of Active Slaves (LAS) from the FIPIO master input table. This word must be managed in the PLC application in order to detect if a Modbus slave is missing.
<b>FIPIO master response</b>
The FIPIO master continues communicating with the PLC processor and the LUF1 gateway.
<b>LUF1 gateway response</b>
With the gateway's standard configuration described here: Modbus timeout time = 300 ms, Retries = 3, Reconnection time = 10 sec, <i>and</i> Effects on FIPIO exchanges... = Clear. After sending a request to a slave, if there is no response after 300 ms, the gateway will send it again three more times before providing information about the missing slave in the List of Active Slave (LAS). If, despite the three retries, there is no answer from the slave, the input data sent to the FIPIO master (Read requests) is reset to 0, but only for the data received from the missing slave. The gateway will try to reconnect the missing slave by communicating with it again every 10 seconds.
<b>Tesys U response</b>
If the LUF1 gateway is disconnected on Modbus side: The starters-controllers do not receive any requests, depending on their configuration, they will: stop the motor, keep the same state, or run the motor. Refer to the Tesys U user manuals to adjust the fallback position.
In case of a Tesys U failure: No response is sent to the gateway, the motor state will be undetermined. This case <b>must</b> be managed in the PLC application.

### 4.2. Gateway Configuration under PL7 PRO

The following sections describe the configuration steps in PL7 PRO that you will need to go through to ensure that the gateway is correctly recognized by the FIPIO master PLC and that it can command and control the 8 motor starters in the standard configuration described earlier.

#### 4.2.1. WorldFIP Mode

The LUF1 gateway supports the WorldFIP mode, configured via the FIPIO bus properties in PL7.

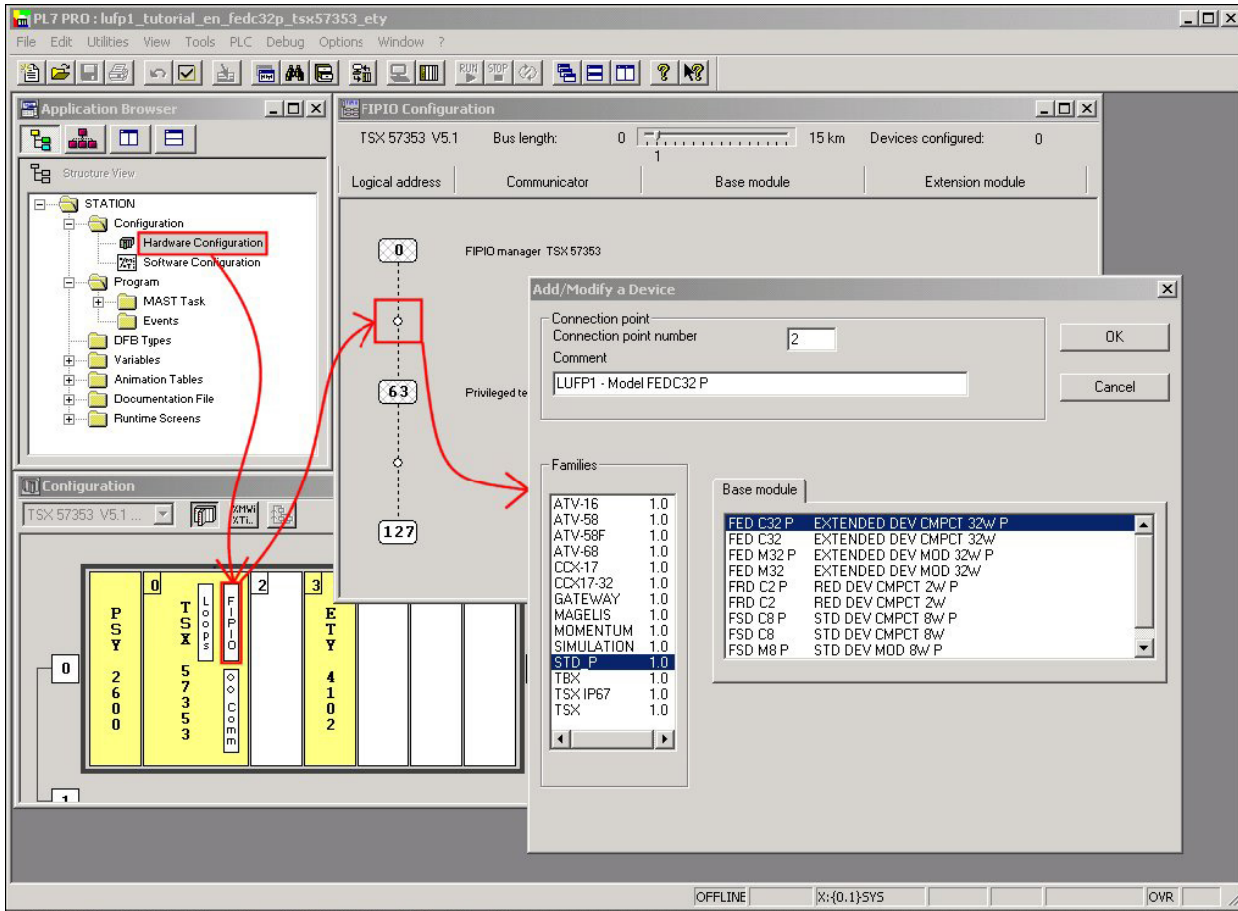
#### 4.2.2. Adding the LUF1 Gateway under PL7 PRO

Open the hardware configuration of your application and edit the FIPIO network configuration by double-clicking **FiPio** on the PLC processor card (TSX 57353).

Add the FIPIO device corresponding to the LUF1 gateway by double-clicking the first free connection point. In the window that appears, select the family "STD\_P" and the basic module for that family, "FED C32 P".

You also need to enter the physical address of the gateway on the FIPIO bus in the field "Connection point number". Entering a "Comment" is optional, but it does make it easier to pick out the gateway from among the other devices on the same FIPIO bus.

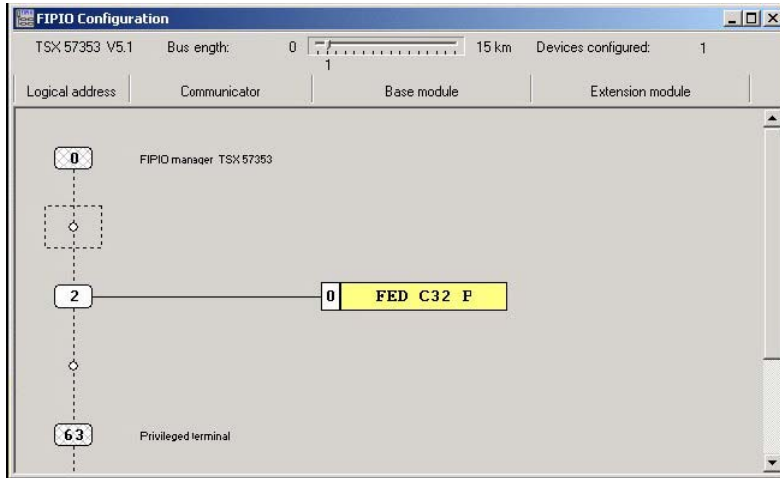
# 4. Software Implementation of the Gateway





# 4. Software Implementation of the Gateway

## 4.2.3. Configuring and Adjusting the LUFP1 Gateway

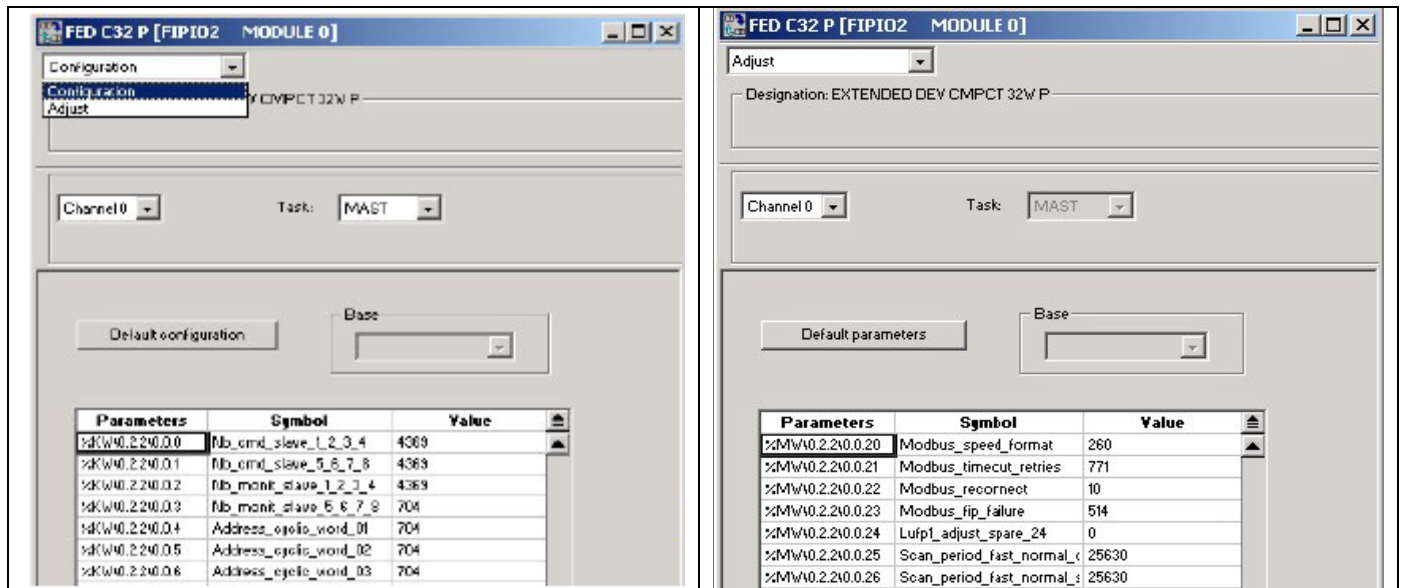


Once the gateway has been added to the FIPIO network (see opposite), double-click the basic module

to edit its configuration and adjustment parameters.

In the window that appears, enter the values given on the next two pages. These correspond to the configuration described in Section 4.1.1.

**NOTE:** Make sure you configure the “Bus Length” correctly (see Section 2.6.2).



**NOTE:** The “FED C32 P [FIPIO2 MODULE 0]” windows shown above present “Symbols” which have been added manually using the PL7 PRO variables editor. By default, no symbol is attributed to the gateway’s configuration parameters (%KW) or adjustment parameters (%MW).

## 4. Software Implementation of the Gateway

All the configuration and adjustment parameters described in the following sections are an integral part of the PLC application. The FIPIO master PLC will transmit these parameters to the gateway implicitly via the FIPIO bus whenever either of the two following events occurs:

- The PLC goes into RUN mode.
- The gateway is disconnected from and reconnected to the FIPIO network.

### NOTE:

- The gateway configuration possibilities offered by the FED C32 P model are limited to the gateway itself; it cannot be used to configure the Modbus slaves.
- If you want to configure the Modbus slaves, you will need to use a different method (local configuration or remote configuration with a software tool such as PowerSuite, for example).

### 4.2.4. Selecting and Adding the Profibus-DP Master Station

Parameter	Value	Description
%KWip.2.ci0.0.0	4369 0x1111	Number of periodic command words for slaves n°1 to n°4: Bits 0- 3: Number of periodic command words for slave n°1 = 1 Bits 4- 7: Number of periodic command words for slave n°2 = 1 Bits 8-11: Number of periodic command words for slave n°3 = 1 Bits 12-15: Number of periodic command words for slave n°4 = 1
%KWip.2.ci0.0.1	4369 0x1111	Number of periodic command words for slaves n°5 to n°8: Bits 0- 3: Number of periodic command words for slave n°5 = 1 Bits 4- 7: Number of periodic command words for slave n°6 = 1 Bits 8-11: Number of periodic command words for slave n°7 = 1 Bits 12-15: Number of periodic command words for slave n°8 = 1
%KWip.2.ci0.0.2	4369 0x1111	Number of periodic control words for slaves n°1 to n°4: Bits 0- 3: Number of periodic control words for slave n°1 = 1 Bits 4- 7: Number of periodic control words for slave n°2 = 1 Bits 8-11: Number of periodic control words for slave n°3 = 1 Bits 12-15: Number of periodic control words for slave n°4 = 1
%KWip.2.ci0.0.3	4369 0x1111	Number of periodic control words for slaves n°5 to n°8: Bits 0- 3: Number of periodic control words for slave n°5 = 1 Bits 4- 7: Number of periodic control words for slave n°6 = 1 Bits 8-11: Number of periodic control words for slave n°7 = 1 Bits 12-15: Number of periodic control words for slave n°8 = 1
%KWip.2.ci0.0.4	704 0x02C0	Address of periodic word n° 1: Address of the unique periodic command word for slave n°1 (704 → command word for a TeSys U motor starter)
%KWip.2.ci0.0.5	704 0x02C0	Address of periodic word n° 2: As above, but for slave n°2
%KWip.2.ci0.0.6	704 0x02C0	Address of periodic word n° 3: As above, but for slave n°3
%KWip.2.ci0.0.7	704 0x02C0	Address of periodic word n° 4: As above, but for slave n°4
%KWip.2.ci0.0.8	704 0x02C0	Address of periodic word n° 5: As above, but for slave n°5
%KWip.2.ci0.0.9	704 0x02C0	Address of periodic word n° 6: As above, but for slave n°6
%KWip.2.ci0.0.10	704 0x02C0	Address of periodic word n° 7: As above, but for slave n°7
%KWip.2.ci0.0.11	704 0x02C0	Address of periodic word n° 8: As above, but for slave n°8
%KWip.2.ci0.0.12	455 0x01C7	Address of periodic word n° 9: Address of the unique periodic control word for slave n°1 (455 → status word for a TeSys U motor starter)
%KWip.2.ci0.0.13	455 0x01C7	Address of periodic word n°10: As above, but for slave n°2
%KWip.2.ci0.0.14	455 0x01C7	Address of periodic word n°11: As above, but for slave n°3
%KWip.2.ci0.0.15	455 0x01C7	Address of periodic word n°12: As above, but for slave n°4
%KWip.2.ci0.0.16	455 0x01C7	Address of periodic word n°13: As above, but for slave n°5
%KWip.2.ci0.0.17	455 0x01C7	Address of periodic word n°14: As above, but for slave n°6
%KWip.2.ci0.0.18	455 0x01C7	Address of periodic word n°15: As above, but for slave n°7
%KWip.2.ci0.0.19	455 0x01C7	Address of periodic word n°16: As above, but for slave n°8
%KWip.2.ci0.0.20	0 0x0000	Address of periodic word n°17: Not used (bits at 0)
.....	.....	.....

## 4. Software Implementation of the Gateway

Parameter	Value	Description
%KWIp.2.cI0.0.29	0 0x0000	Address of periodic word n <sup>26</sup> : Not used (bits at 0)

**Syntax:** p = processor position (0 or 1);  
c = connection point number (gateway address).

For modifications to these adjustment parameters, see Appendix B.

However, during the Software Implementation of the Gateway, you will probably find it useful to know how to delete one or more TeSys U motor starters. This operation is therefore described in Section 4.2.6.

### 4.2.5. Adjustment Parameter Values, Standard Configuration

Parameter	Value	Description
%MWIp.2.cI0.0.20	260 0x0104	Bits 0- 7: Modbus speed = 19,200 bits/s (LSB byte = 4) Bit 8: Modbus format = 8 data bits (bit at 1) Bit 9: Modbus format = 1 stop bit (bit at 0) Bits 10-11: Modbus format = No parity (bits at 0) Bits 12-15: Not used (bits at 0)
%MWIp.2.cI0.0.21	771 0x0303	Bits 0- 7: Modbus response timeout = 300 ms (3 * 100 ms) Bits 8-15: Number of frame retransmissions over the Modbus network = 3
%MWIp.2.cI0.0.22	10 0x000A	Bits 0- 7: Time taken by gateway to reconnect Modbus slave following a break in communication = 10 s (10 * 1 s) Bits 8-15: Not used (bits at 0)
%MWIp.2.cI0.0.23	514 0x0202	Bits 0- 7: Behavior on losing FIPIO network = Reset periodic command word values sent to the Modbus slaves (LSB byte = 2) Bits 8-15: Behavior on losing Modbus network = Reset periodic control word values sent by the Modbus slaves (LSB byte = 2)
%MWIp.2.cI0.0.24	0 0x0000	Not used (bits at 0)
%MWIp.2.cI0.0.25	25 630 0x641E	Bits 0- 7: Cycle time of fast periodic commands = 300 ms (30 * 10 ms) Bits 8-15: Cycle time of normal periodic commands = 1 s (100 * 10 ms)
%MWIp.2.cI0.0.26	25 630 0x641E	Bits 0- 7: Cycle time of fast periodic controls = 300 ms (30 * 10 ms) Bits 8-15: Cycle time of normal periodic controls = 1 s (100 * 10 ms)
%MWIp.2.cI0.0.27	0 0x0000	Cycle time settings for periodic commands n <sup>1</sup> to 16: Bit 0: Periodic command n <sup>1</sup> = fast (bit at 0) Bits 1 to 7: As above, but for commands n <sup>2</sup> to 8, in that order Bits 8 to 15: Not used (bits at 0)
%MWIp.2.cI0.0.28	0 0x0000	Cycle time settings for periodic commands n <sup>17</sup> to 26: Not used
%MWIp.2.cI0.0.29	0 0x0000	Cycle time settings for periodic controls n <sup>1</sup> to 16: Bit 0: Periodic control n <sup>1</sup> = fast (bit at 0) Bits 1 to 7: As above, but for controls n <sup>2</sup> to 8, in that order Bits 8 to 15: Not used (bits at 0)
%MWIp.2.cI0.0.30	0 0x0000	Cycle time settings for periodic controls n <sup>17</sup> to 26: Not used
%MWIp.2.cI0.0.31	513 0x0201	<b>Modbus addresses:</b> Slave n <sup>1</sup> (bits 0-7) = 1 — Slave n <sup>2</sup> (bits 8-15) = 2
%MWIp.2.cI0.0.32	1 027 0x0403	<b>Modbus addresses:</b> Slave n <sup>3</sup> (bits 0-7) = 3 — Slave n <sup>4</sup> (bits 8-15) = 4
%MWIp.2.cI0.0.33	1 541 0x0605	<b>Modbus addresses:</b> Slave n <sup>5</sup> (bits 0-7) = 5 — Slave n <sup>6</sup> (bits 8-15) = 6
%MWIp.2.cI0.0.34	2 055 0x0807	<b>Modbus addresses:</b> Slave n <sup>7</sup> (bits 0-7) = 7 — Slave n <sup>8</sup> (bits 8-15) = 8
%MWIp.2.cI0.0.35	0 0x0000	Not used (bits at 0)
.....	.....	.....
%MWIp.2.cI0.0.49	0 0x0000	Not used (bits at 0)

**Syntax:** p = processor position (0 or 1);  
c = connection point number (gateway address).

## 4. Software Implementation of the Gateway

---

### **WARNING**

#### **RISK OF UNINTENDED EQUIPMENT OPERATION**

Do not confuse the order of configuration of the Modbus slaves with their Modbus addresses.

In the configuration shown in parameters %MW\p.2.c\0.0.31 to %MW\p.2.c\0.0.34 the order numbers of the Modbus slaves (1 to 8) are identical to the addresses of the corresponding TeSys U motor starters (Modbus addresses 1 to 8).

**Failure to follow this instruction can result in death, serious injury, or equipment damage.**

For modifications to these adjustment parameters, see Appendix B.

However, during the Software Implementation of the Gateway, you will probably find it useful to know how to delete one or more TeSys U motor starters. This operation is therefore described in Section 4.2.6.

#### **4.2.6. Deleting One or More TeSys U Motor Starters from the Configuration**

We will just look at one modification operation on the standard 8 TeSys U motor starter configuration: the deletion of one or more motor starters. The 8 motor starters were configured to represent the typical *maximal* configuration (a maximum of 8 Modbus slaves).

It is best to delete any slaves absent from your configuration in order to avoid any degradation in performance that might be caused by the physical absence of motor starters declared in the LUFP1 gateway configuration.

Make the following changes to the Configuration Parameter Values, Standard Config (Section 4.2.4) and to the Adjustment Parameter Values, Standard Configuration (Section 4.2.5):

- ① For each absent TeSys U motor starter, reset to zero the number of periodic command words (%KW\p.2.c\0.0.0 and %KW\p.2.c\0.0.1) *and* the number of periodic control words (%KW\p.2.c\0.0.2 and %KW\p.2.c\0.0.3) on a single slave, starting with slave n°8 and working back towards slave n°1.
- ② For each absent TeSys U motor starter, *two* of the addresses of periodic words n°1 to 16 (%KW\p.2.c\0.0.4 to %KW\p.2.c\0.0.19) will become “Not used” parameters, starting with n°16 and working back towards n°1. These two words comprise a command word / control word pair, which means that you have to delete the last command word, shift the control words along by one parameter, and then delete the last control word.
- ③ Check the mappings between Modbus slaves n°1 to 8 and the addresses of the TeSys U motor starters in your configuration (%MW\p.2.c\0.0.31 to %MW\p.2.c\0.0.34). For each absent motor starter, one of the slaves will take the value 0x00 as its Modbus address, starting with slave n°8 and working back to n°1.

## 4. Software Implementation of the Gateway

**Example:** Deletion of the motor starters corresponding to Modbus addresses 1, 3, 4 and 8. The starters at Modbus addresses 2, 5, 6 and 7 will remain present. The new configuration and settings for the gateway are as follows:

Parameter	Value	Description
%KW\p.2.c\0.0.0	4369 0x1111	Number of periodic command words for each slave from n°1 to n°4
%KW\p.2.c\0.0.1	0 0x0000	Not used
%KW\p.2.c\0.0.2	4369 0x1111	Number of periodic control words for each slave from n°1 to n°4
%KW\p.2.c\0.0.3	0 0x0000	Not used
%KW\p.2.c\0.0.4	704 0x02C0	Address of periodic word n° 1: Address of the command word for slave n°1
%KW\p.2.c\0.0.5	704 0x02C0	Address of periodic word n° 2: As above, but for slave n°2
%KW\p.2.c\0.0.6	704 0x02C0	Address of periodic word n° 3: As above, but for slave n°3
%KW\p.2.c\0.0.7	704 0x02C0	Address of periodic word n° 4: As above, but for slave n°4
%KW\p.2.c\0.0.8	455 0x01C7	Address of periodic word n° 5: Address of the control word for slave n°1
%KW\p.2.c\0.0.9	455 0x01C7	Address of periodic word n° 6: As above, but for slave n°2
%KW\p.2.c\0.0.10	455 0x01C7	Address of periodic word n° 7: As above, but for slave n°3
%KW\p.2.c\0.0.11	455 0x01C7	Address of periodic word n° 8: As above, but for slave n°4
%KW\p.2.c\0.0.12	0 0x0000	Address of periodic word n° 9: Not used
.....	.....	.....
%KW\p.2.c\0.0.29	0 0x0000	Address of periodic word n°26: Not used

Parameter	Value	Description
%MW\p.2.c\0.0.20	260 0x0104	Modbus format and speed
%MW\p.2.c\0.0.21	771 0x0303	Modbus timeout and number of retransmissions
%MW\p.2.c\0.0.22	10 0x000A	Reconnection time for a Modbus slave
%MW\p.2.c\0.0.23	514 0x0202	Behavior on losing FIPIO network and Modbus network
%MW\p.2.c\0.0.24	0 0x0000	Not used
%MW\p.2.c\0.0.25	25 630 0x641E	Cycle time of fast / normal periodic commands
%MW\p.2.c\0.0.26	25 630 0x641E	Cycle time of fast / normal fast controls
%MW\p.2.c\0.0.27	0 0x0000	Bits 0 to 3: Periodic commands n°1 to 4 = fast (bits at 0) Bits 4 to 15: Not used
%MW\p.2.c\0.0.28	0 0x0000	Not used
%MW\p.2.c\0.0.29	0 0x0000	Bits 0 to 3: Periodic controls n°1 to 4 = fast (bits at 0) Bits 4 to 15: Not used
%MW\p.2.c\0.0.30	0 0x0000	Not used
%MW\p.2.c\0.0.31	1 282 0x0502	<b>Modbus addresses:</b> Slave n°1 (bits 0-7) = 2 — Slave n°2 (bits 8-15) = 5
%MW\p.2.c\0.0.32	1 798 0x0706	<b>Modbus addresses:</b> Slave n°3 (bits 0-7) = 6 — Slave n°4 (bits 8-15) = 7
%MW\p.2.c\0.0.33	0 0x0000	<b>Modbus addresses:</b> Slaves n°5 and 6 absent
%MW\p.2.c\0.0.34	0 0x0000	<b>Modbus addresses:</b> Slaves n°7 and 8 absent
%MW\p.2.c\0.0.35	0 0x0000	Not used
.....	.....	.....
%MW\p.2.c\0.0.49	0 0x0000	Not used

## 4. Software Implementation of the Gateway

---

### 4.2.7. Configuring Gateway Inputs / Outputs

The settings described in the previous sections enable the gateway to establish a Modbus scanner made up of periodic write commands (“commands”) and read instructions (“controls”), which are sent out to the Modbus slaves.

Each control word corresponds to a Modbus periodic read instruction, and each command word to a Modbus periodic write command. Each word also corresponds to a gateway input or output. All of these inputs and outputs are defined by the gateway by means of its configuration parameters (%KW): the periodic control words correspond to the inputs, and the periodic command words to the outputs.

The gateway establishes the following correspondences between its inputs / outputs and the values of its periodic command / control words:

Inputs	Periodic control words	Outputs	Periodic command words
%IWp.2.c\0.0	First control word from slave n°1	%QWp.2.c\0.0	First command word for slave n°1
⋮	.....	⋮	.....
⋮	Last control word from slave n°1	⋮	Last command word for slave n°1
⋮	First control word from slave n°2	⋮	First command word for slave n°2
⋮	.....	⋮	.....
⋮	Last control word from slave n°2	⋮	Last command word for slave n°2
⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮
⋮	First control word from slave n°8	⋮	First command word for slave n°8
⋮	.....	⋮	.....
⋮	Last control word from slave n°8	⋮	Last command word for slave n°8
⋮	⋮	⋮	⋮
%IWp.2.c\0.0.25	⋮	%QWp.2.c\0.0.25	⋮

## 4. Software Implementation of the Gateway

For the standard configuration presented here, the mapping between the gateway inputs and the PLC inputs is as follows:

Service	PLC input	Description	
		Bit 15.....Bit 8	Bit 7.....Bit 0
Periodic communications — Control (monitoring) of TeSys U motor starters	%IW\p.2.c\0.0	Value of status register for motor starter ①	
	%IW\p.2.c\0.0.1	Value of status register for motor starter ②	
	%IW\p.2.c\0.0.2	Value of status register for motor starter ③	
	%IW\p.2.c\0.0.3	Value of status register for motor starter ④	
	%IW\p.2.c\0.0.4	Value of status register for motor starter ⑤	
	%IW\p.2.c\0.0.5	Value of status register for motor starter ⑥	
	%IW\p.2.c\0.0.6	Value of status register for motor starter ⑦	
Periodic communications — Free locations	%IW\p.2.c\0.0.8 ..... %IW\p.2.c\0.0.25	Not used (18 words)	
	%IW\p.2.c\0.0.26	Reserved (1 word)	
List of active slaves (LAS) service	%IW\p.2.c\0.0.27	List of active Modbus slaves	
Aperiodic communications — Indexed periodic variables (PKW) service – RESPONSE	%IW\p.2.c\0.0.28	PKE – Address of the written / read datum	
	%IW\p.2.c\0.0.29	DN – Address of the responding slave	R/W/N – Read / Write OK / Error
	%IW\p.2.c\0.0.30	PWE – Value written / read (1st word)	
	%IW\p.2.c\0.0.31	PWE – Value written / read (2nd word)	

Likewise, the PLC outputs correspond to the gateway outputs as follows:

Service	PLC output	Description	
		Bit 15.....Bit 8	Bit 7.....Bit 0
Periodic communications — Command of TeSys U motor starters	%QW\p.2.c\0.0	Value of command register for motor starter ①	
	%QW\p.2.c\0.0.1	Value of command register for motor starter ②	
	%QW\p.2.c\0.0.2	Value of command register for motor starter ③	
	%QW\p.2.c\0.0.3	Value of command register for motor starter ④	
	%QW\p.2.c\0.0.4	Value of command register for motor starter ⑤	
	%QW\p.2.c\0.0.5	Value of command register for motor starter ⑥	
	%QW\p.2.c\0.0.6	Value of command register for motor starter ⑦	
Periodic communications — Free locations	%QW\p.2.c\0.0.8 ...to... %QW\p.2.c\0.0.25	Not used (18 words)	
	%QW\p.2.c\0.0.26 %QW\p.2.c\0.0.27	Reserved (2 words)	
Aperiodic communications — Indexed periodic variables (PKW) service – COMMAND	%QW\p.2.c\0.0.28	PKE – Address of the datum to be read / written	
	%QW\p.2.c\0.0.29	DN – Address of the queried slave	R/W – Read / write command
	%QW\p.2.c\0.0.30	PWE – Value to be written (1st word)	
	%QW\p.2.c\0.0.31	PWE – Value to be written (2nd word)	

## 4. Software Implementation of the Gateway

---

### 4.2.8. Description of Services Assigned to Gateway I / O

On the FIPIO network, all the inputs / outputs that correspond to the gateway are periodic; the same is not true of the Modbus network. There are therefore two types of service: periodic services and aperiodic services.

All of these services correspond to FIPIO objects. For a more detailed description, see Chapter 5.

**Periodic communications (inputs):** For the configuration described here, the value of each of the first 8 gateway input words corresponds to the value of the status register of a TeSys U motor starter (register located at address 455).

**Periodic communications (outputs):** Likewise, the value of each of the first 8 gateway output words corresponds to a value to be sent to the command register of a TeSys U motor starter (register located at address 704).

The FIPIO objects corresponding to the gateway's periodic Modbus communications are described in Section 5.1.

See also Command and Control of the 8 TeSys U Motor Starters in Appendix D for an example of how these "periodic communications" services are used with the type of configuration adopted in this gateway implementation (8 TeSys U devices).

If you delete any motor starters from the configuration, you will need to adjust the two tables on the previous page accordingly.

**Aperiodic communications:** The aperiodic communications service operates in the same way as the "indexed periodic variables", or PKW, service found on a number of other *Schneider Electric* products, such as some ATV drives. Hence the service continues to use the name "PKW".

The FIPIO objects corresponding to the gateway's aperiodic Modbus communications are described in Section 5.3.

See also Using the Indexed Periodic Variables (PKW) Service in Appendix D for an example of the simplified use of the gateway's "aperiodic communications" service.



**List of active slaves (LAS):** This service is specific to the gateway, and does not generate any communications over the Modbus network. That is why it is named differently from the other gateway inputs / outputs (periodic or aperiodic communications). Each bit (from 0 to 7) of the input word indicates the absence or presence of a Modbus slave (slaves n°1 to 8, in that order).

The only FIPIO object corresponding to the list of active slaves service is described in Section 5.2.



# 4. Software Implementation of the Gateway

## 4.2.9. Validating and Saving the FIPIO Network Configuration

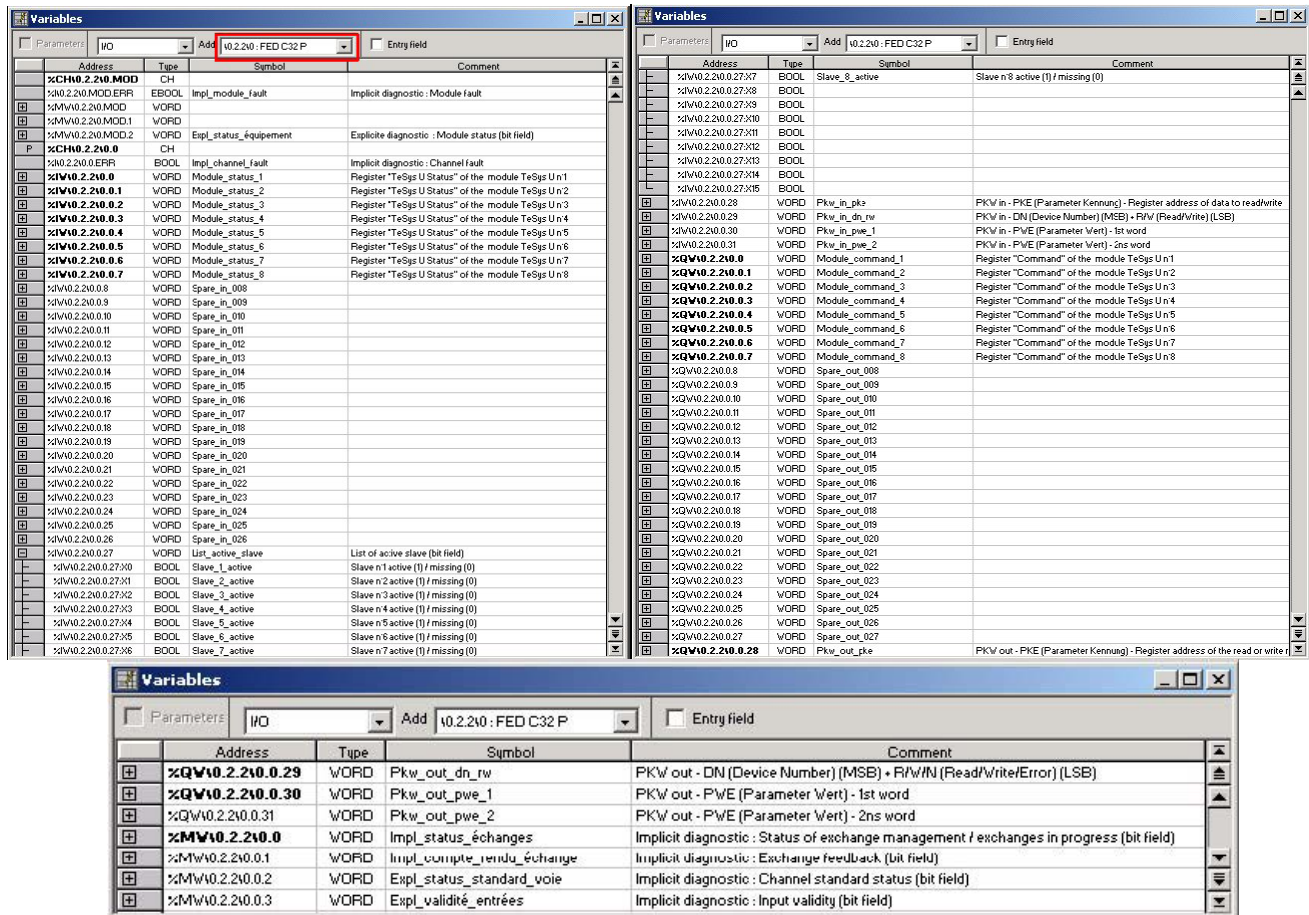
- ➔ Confirm the FED C32 P profile settings and configuration by clicking the  button.
- ➔ Close the FED C32 P profile settings and configuration window.
- ➔ In the FIPIO network configuration window, click  again.
- ➔ Close this window, and the PLC hardware configuration window.
- ➔ Save the PL7 PRO application by giving it a name.

## 4.2.10. Assigning Symbols to the Gateway's FIPIO Objects

It is advisable to assign symbols to the gateway's inputs and outputs: this makes it easier to understand the PLC application that will be managing them. Edit I / O variables using PL7 PRO, and select the address "0.2.2\0: FED C32 P" if, for example, the gateway is located at address 2.

You can also assign symbols to the configuration and adjustment parameters described in the previous sections, as well as to the gateway's FIPIO diagnostic objects, but this is not as useful as it is for the inputs and outputs.

Once defined, these symbols are used in the various FED C32 P profile configuration and monitoring windows.





The screenshots show the 'Variables' window in PL7 PRO, displaying a list of variables for the address '0.2.2\0: FED C32 P'. The variables are organized into columns: Address, Type, Symbol, and Comment. The first screenshot shows variables like %CH0.2.2\0.MOD, %Mw0.2.2\0.MOD, and %Mw0.2.2\0.0. The second screenshot shows variables like %QV0.2.2\0.0, %QV0.2.2\0.0.1, and %QV0.2.2\0.0.2. The third screenshot is a zoomed-in view of the bottom of the list, showing variables like %QV0.2.2\0.0.29, %QV0.2.2\0.0.30, and %Mw0.2.2\0.0.

Address	Type	Symbol	Comment
%CH0.2.2\0.MOD	CH		
%Mw0.2.2\0.MOD	WORD	Impl_module_fault	Implicit diagnostic : Module fault
%Mw0.2.2\0.0	WORD	Impl_status_équipement	Explicite diagnostic : Module status (bit field)
%Mw0.2.2\0.0.1	WORD	Impl_channel_fault	Implicit diagnostic : Channel fault
%Mw0.2.2\0.0.2	WORD	Module_status_1	Register "TeSys U Status" of the module TeSys Un1
%Mw0.2.2\0.0.3	WORD	Module_status_2	Register "TeSys U Status" of the module TeSys Un2
%Mw0.2.2\0.0.4	WORD	Module_status_3	Register "TeSys U Status" of the module TeSys Un3
%Mw0.2.2\0.0.5	WORD	Module_status_4	Register "TeSys U Status" of the module TeSys Un4
%Mw0.2.2\0.0.6	WORD	Module_status_5	Register "TeSys U Status" of the module TeSys Un5
%Mw0.2.2\0.0.7	WORD	Module_status_6	Register "TeSys U Status" of the module TeSys Un6
%Mw0.2.2\0.0.8	WORD	Module_status_7	Register "TeSys U Status" of the module TeSys Un7
%Mw0.2.2\0.0.9	WORD	Module_status_8	Register "TeSys U Status" of the module TeSys Un8
%Mw0.2.2\0.0.10	WORD	Spare_in_008	
%Mw0.2.2\0.0.11	WORD	Spare_in_009	
%Mw0.2.2\0.0.12	WORD	Spare_in_010	
%Mw0.2.2\0.0.13	WORD	Spare_in_011	
%Mw0.2.2\0.0.14	WORD	Spare_in_012	
%Mw0.2.2\0.0.15	WORD	Spare_in_013	
%Mw0.2.2\0.0.16	WORD	Spare_in_014	
%Mw0.2.2\0.0.17	WORD	Spare_in_015	
%Mw0.2.2\0.0.18	WORD	Spare_in_016	
%Mw0.2.2\0.0.19	WORD	Spare_in_017	
%Mw0.2.2\0.0.20	WORD	Spare_in_018	
%Mw0.2.2\0.0.21	WORD	Spare_in_019	
%Mw0.2.2\0.0.22	WORD	Spare_in_020	
%Mw0.2.2\0.0.23	WORD	Spare_in_021	
%Mw0.2.2\0.0.24	WORD	Spare_in_022	
%Mw0.2.2\0.0.25	WORD	Spare_in_023	
%Mw0.2.2\0.0.26	WORD	Spare_in_024	
%Mw0.2.2\0.0.27	WORD	List_active_slave	List of active slave (bit field)
%QV0.2.2\0.0.27	WORD	Slave_1_active	Slave n1 active (1) / missing (0)
%QV0.2.2\0.0.28	WORD	Slave_2_active	Slave n2 active (1) / missing (0)
%QV0.2.2\0.0.29	WORD	Slave_3_active	Slave n3 active (1) / missing (0)
%QV0.2.2\0.0.30	WORD	Slave_4_active	Slave n4 active (1) / missing (0)
%QV0.2.2\0.0.31	WORD	Slave_5_active	Slave n5 active (1) / missing (0)
%QV0.2.2\0.0.32	WORD	Slave_6_active	Slave n6 active (1) / missing (0)
%QV0.2.2\0.0.33	WORD	Slave_7_active	Slave n7 active (1) / missing (0)
%QV0.2.2\0.0.29	WORD	Pkw_out_dn_rw	PKW out - DN (Device Number) (MSB) + R/W (Read/Write) (LSB)
%QV0.2.2\0.0.30	WORD	Pkw_out_pwe_1	PKW out - PwE (Parameter Wert) - 1st word
%QV0.2.2\0.0.31	WORD	Pkw_out_pwe_2	PKW out - PwE (Parameter Wert) - 2ns word
%Mw0.2.2\0.0	WORD	Impl_status_échanges	Implicit diagnostic : Status of exchange management / exchanges in progress (bit field)
%Mw0.2.2\0.0.1	WORD	Impl_compte_rendu_échange	Implicit diagnostic : Exchange feedback (bit field)
%Mw0.2.2\0.0.2	WORD	Impl_status_standard_voie	Implicit diagnostic : Channel standard status (bit field)
%Mw0.2.2\0.0.3	WORD	Impl_validité_entrées	Implicit diagnostic : Input validity (bit field)

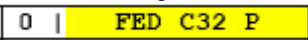
# 4. Software Implementation of the Gateway

## 4.2.11. Checking the Operational Status of the Gateway

After validating any changes made, you must send the new configuration and adjustment parameters for profile FED C32 P to the corresponding gateway by downloading the PL7 PRO application to the PLC. Then check that the gateway has been duly recognized on FIPIO by the PLC, by viewing its status in the FIPIO network configuration window.

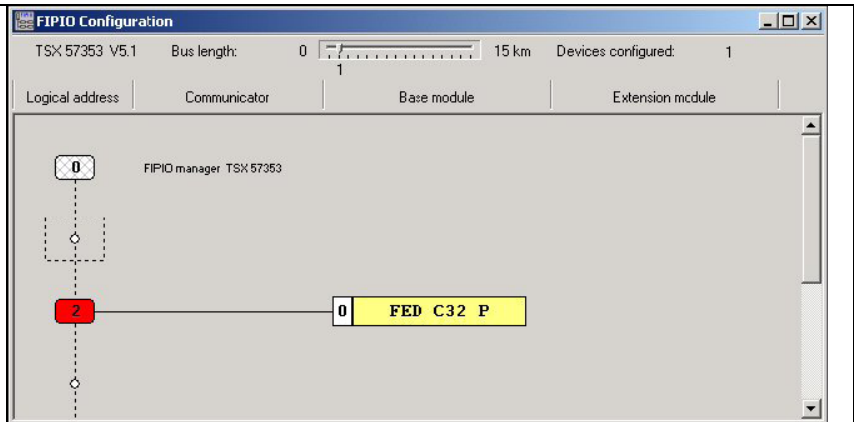
- ➔ Transfer the application from the PC to the PLC by selecting the “Transfer program...” command in the “AP” menu (or by clicking the  button) and selecting “PC -> PLC”.
- ➔ Switch from **OFFLINE** to **ONLINE** mode by selecting “Connect” in the “AP” menu (or by clicking ).
- ➔ Initialize and start the PLC application using the “Init...” and “Run...” commands in the “AP” menu.
- ➔ Open the PLC hardware configuration and the FIPIO configuration and check that the gateway connection point is valid (colored white).

In the example shown here, the connection point is signaling an error (and is therefore in red). If this happens, open the “debug” window by double-clicking the basic module



In this window, click on the two red “DIAG...” buttons.

The table below sets out the procedure to be followed in the two most likely instances:



Module diagnosis			Channel diagnosis			Causes and corrective measures
Internal faults	External faults	Other faults	Internal faults	External faults	Other faults	
—	Functional fault	—	—	—	Application fault	Incorrect configuration / adjustment parameters ↳ Switch to OFFLINE mode, modify the gateway's parameter values, save the changes, and reload the application onto the FIPIO master PLC.
—	Functional fault	—	—	—	Hardware configuration fault	Gateway configured in FED C32 ↳ Update the internal gateway configuration using the ABC-LUFP Config Tool configuration tool (see Section 7.5.1).

**NOTE:** Neither of these configuration errors will be signaled by the gateway (its LEDs will indicate normal operation). It is therefore essential that you perform the foregoing checks.

# 4. Software Implementation of the Gateway

## 4.2.12. Modifying the Gateway Parameters in ONLINE Mode

The modification of one or more configuration and / or adjustment parameters in **ONLINE** mode will momentarily bring the gateway to a halt (for complete reconfiguration of the Modbus scanner). This interrupts the periodic Modbus communications, but for such a brief period of time that the Modbus slaves are unlikely to detect it, even if they have very short communication timeouts.

## 4.2.13. Debugging and Using the Gateway Configuration

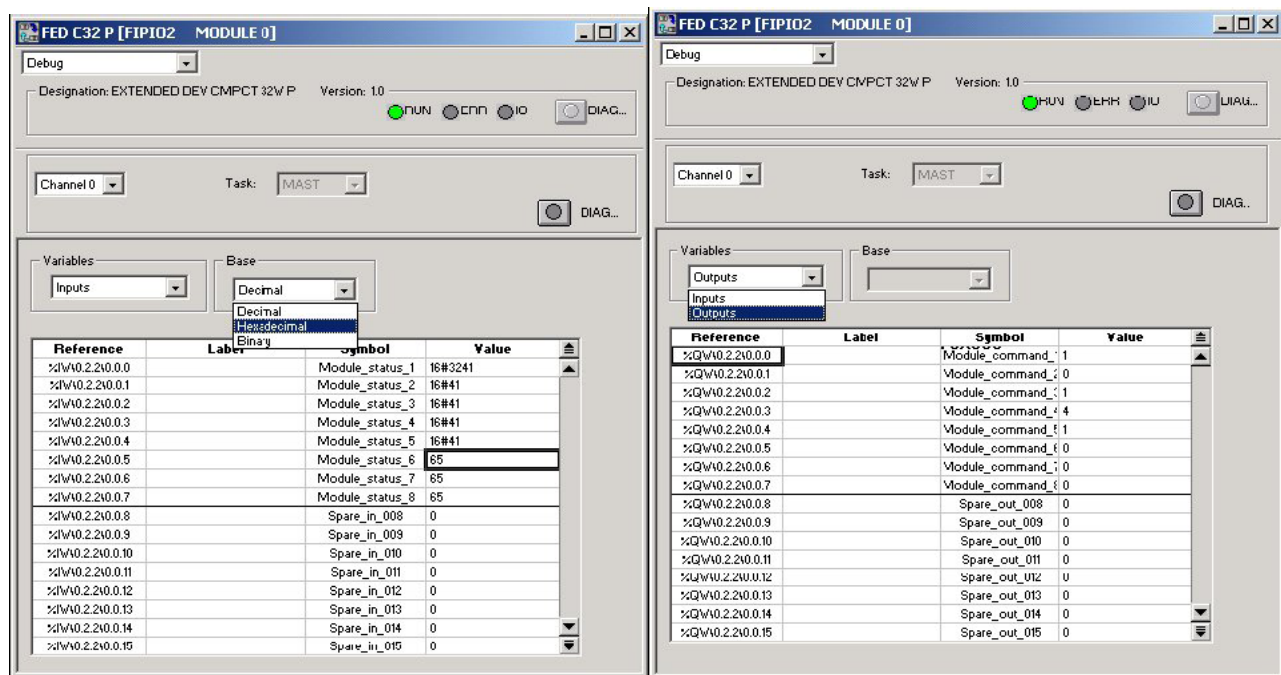
While in **ONLINE** mode, open the gateway debug window by double-clicking the basic module



This window displays the gateway diagnostics (see previous page), and the values of the gateway inputs or outputs (selected using the scroll bar in the “Variables” pane).

The display mode for the selected value can be modified (in the “Base” pane) to make it easier to read if, for example, it consists of two independent bytes or a register of 16 independent bits.

The values of the outputs can also be modified, so that you can check—in the absence of a PLC program that updates the values periodically—that the periodic commands have been transmitted to the gateway, first of all, and subsequently to the TeSys U motor starters.



## 4.2.14. Developing a FIPIO Application

The FIPIO master PLC used in the example is a TSX 57353 v5.1, marketed by Schneider Electric. A sample PLC application, developed under PL7 PRO, is presented in Appendix D: Sample Use under PL7 PRO. This example uses the PLC, the gateway and the 8 TeSys U motor starters shown in the Software Implementation of the Gateway.

## 5. FIPIO Objects Available for Programming

This section presents all the language objects associated with standard profiles FED C32 and FED C32 P (depending on the chosen configuration method—PL7 PRO or ABC-LUFP Config Tool) for FIPIO communication with the LUFP1 gateway.

All the FIPIO objects described in the following sections (e.g. output word %QW\p.2.c\0.0.7) obey the following syntax:

- p = processor position (0 or 1);
- c = connection point number (gateway address).

The table below sets out all the FIPIO objects available for use with the gateway. They are then described in detail in the rest of the section.

FIPIO Object	Designation	Format	Access (1)
%Ip.2.c\0.0.ERR	Channel fault (if bit at 1)	1 bit	IMPLICIT read
%Ip.2.c\0.MOD.ERR	Module fault (if bit at 1)	1 bit	
%IW\p.2.c\0.0 → \0.0.25	Periodic input variables (control of Modbus slaves)	26 words	IMPLICIT read
%IW\p.2.c\0.0.26	Reserved (word = 0x0000)	1 word	
%IW\p.2.c\0.0.27	List of active slaves (LAS) service	1 word	
%IW\p.2.c\0.0.28 → \0.0.31	Indexed periodic variables (PKW) service: Response	4 words	
%QW\p.2.c\0.0 → \0.0.25	Periodic output variables (command of Modbus slaves)	26 words	IMPLICIT write
%QW\p.2.c\0.0.26 → \0.0.27	Reserved (words not used by gateway)	2 words	
%QW\p.2.c\0.0.28 → \0.0.31	Indexed periodic variables (PKW) service: Command	4 words	
%MW\p.2.c\0.0	Explicit exchanges: Exchange in progress	1 word	IMPLICIT read
%MW\p.2.c\0.0.1	Explicit exchanges: Report on last exchange	1 word	
%MW\p.2.c\0.MOD.2	Module status (gateway diagnosis)	1 word	EXPLICIT read
%MW\p.2.c\0.0.2	Channel status (gateway and communication diagnosis)	1 word	
%MW\p.2.c\0.0.3	Reserved (word = 0x0000)	1 word	EXPLICIT read
%MW\p.2.c\0.0.4 → \0.0.11	Reserved (words = 0xFFFF)	8 words	
%MW\p.2.c\0.0.12 → \0.0.19	Reserved (words = 0x0000)	8 words	EXPLICIT write
%MW\p.2.c\0.0.20 → \0.0.49	Adjustment parameters for profile FED C32 P (2)	30 words	EXPLICIT write
%KW\p.2.c\0.0 → \0.0.29	Configuration parameters for profile FED C32 P (2)	30 words	
%SW128 → %SW135	Faulty FIPIO connection point (if corresponding bit at 0)	8 words	System

- (1) IMPLICIT exchanges are performed by the PLC *without any additional programming*; these are the periodic FIPIO exchanges. EXPLICIT exchanges call for the *programming of communication functions* like READ\_PARAM, READ\_STS, etc; these are aperiodic FIPIO exchanges for passing diagnostic variables, downloading complete configurations, and so on.
- (2) If the gateway is configured using ABC-LUFP Config Tool (see Chapter 6), the profile FED C32 must be used. The configuration and adjustment parameters then become internal to the gateway and no longer need to be transmitted by the FIPIO PLC, and the associated %MW and %KW objects cease to exist.

# 5. FIPIO Objects Available for Programming

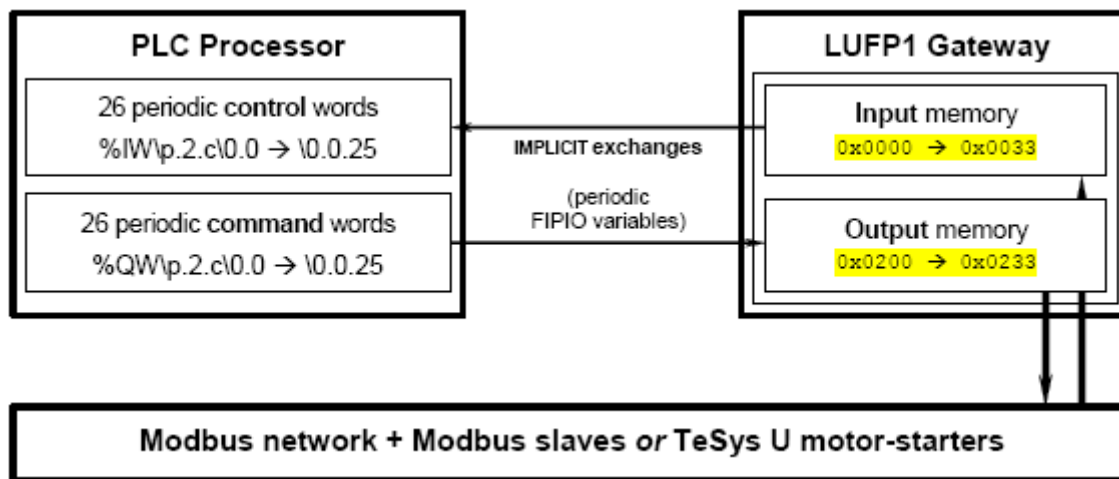
## 5.1. Periodic Command and Control Words

Unlike the other FIPIO objects for the LUFPI gateway, the periodic command words (%QW\p.2.c\0.0 to %QW\p.2.c\0.0.25) and periodic control words (%IW\p.2.c\0.0 to %IW\p.2.c\0.0.25) are fully configurable by the user.

Their organizational structure is presented in three different ways, depending on how the gateway is used:

- Use of the standard configuration with 8 TeSys U motor starters;
- Configuration with PL7 PRO, using the standard profile FED C32 P;
- External configuration (with ABC-LUFP Config Tool), using the standard profile FED C32.

These objects are updated implicitly by the FIPIO master (by periodic FIPIO exchanges), at the start of the PLC cycle for the control words, and at the end of the PLC cycle for the command words.



### 5.1.1. Standard Configuration with 8 TeSys U Motor Starters

The periodic command and control words described below correspond to the configuration described in Chapter 4.

#### Inputs / Controls

FIPIO object	Description
%IW\p.2.c\0.0	Status of motor starter ①
%IW\p.2.c\0.0.1	Status of motor starter ②
%IW\p.2.c\0.0.2	Status of motor starter ③
%IW\p.2.c\0.0.3	Status of motor starter ④
%IW\p.2.c\0.0.4	Status of motor starter ⑤
%IW\p.2.c\0.0.5	Status of motor starter ⑥
%IW\p.2.c\0.0.6	Status of motor starter ⑦
%IW\p.2.c\0.0.7	Status of motor starter ⑧
%IW\p.2.c\0.0.8	Not used (18 words)
.....	
%IW\p.2.c\0.0.25	
%IW\p.2.c\0.0.26	Reserved (1 word)

#### Outputs / Commands

FIPIO object	Description
%QW\p.2.c\0.0	Command of motor starter ①
%QW\p.2.c\0.0.1	Command of motor starter ②
%QW\p.2.c\0.0.2	Command of motor starter ③
%QW\p.2.c\0.0.3	Command of motor starter ④
%QW\p.2.c\0.0.4	Command of motor starter ⑤
%QW\p.2.c\0.0.5	Command of motor starter ⑥
%QW\p.2.c\0.0.6	Command of motor starter ⑦
%QW\p.2.c\0.0.7	Command of motor starter ⑧
%QW\p.2.c\0.0.8	Not used (18 words)
.....	
%QW\p.2.c\0.0.25	
%QW\p.2.c\0.0.26	Reserved (2 words)
%QW\p.2.c\0.0.27	

## 5. FIPIO Objects Available for Programming

---

The input assigned to a motor starter gives the value of the status register for that motor starter (address 704 in the motor starter memory). The gateway handles the periodic Modbus communications with the motor starter in order to update the value of the FIPIO object that corresponds to the input.

The output value assigned to a motor starter is transmitted to that motor starter in order to update the value of its command register (address 455 in the motor starter memory). The gateway handles the periodic Modbus communications with the motor starter in order to transmit the value of the FIPIO object that corresponds to the output.

Transmission is periodic, but writing a *new value* means that the gateway will transmit it as soon as possible.

**NOTE:** If you delete any motor starters from the configuration (see Section 4.2.6), you will need to adapt the two tables above accordingly. If, for example, you delete 3 motor starters (it has to be the last 3, whichever Modbus addresses are used), the statuses and commands for motor starters □, □ and □ will become the words “Not used”.

### 5.1.2. Configuration under PL7 PRO with Profile FED C32 P

The actual contents of the periodic command and control words described below will depend entirely on the values you entered for the gateway’s configuration and adjustment parameters. These parameters are described in Chapter 1.

**Periodic control words**

FIPIO object	Description
%IW\p.2.c\0.0	Periodic control word n° 1
%IW\p.2.c\0.0.1	Periodic control word n° 2
.....	.....
%IW\p.2.c\0.0.25	Periodic control word n°26 (1)
%IW\p.2.c\0.0.26	Reserved word

**Periodic command words**

FIPIO object	Description
%QW\p.2.c\0.0	Periodic command word n° 1
%QW\p.2.c\0.0.1	Periodic command word n° 2
.....	.....
%QW\p.2.c\0.0.25	Periodic command word n°26 (1)
%QW\p.2.c\0.0.26	Reserved word
%QW\p.2.c\0.0.27	Reserved word

(1) You can only configure a total of 26 periodic words. If you configure 20 periodic control words, for example, you can then only configure a maximum of 6 periodic command words.

Periodic words (control and command) are organized in increasing order of the Modbus slaves (from n°1 to n°8). Within this schema, the words assigned to each slave are, in turn, organized in increasing order.

*Example:* In the next example, the gateway is configured to control and command 4 Modbus slaves. This example exploits the fact that the number of words is not necessarily the same for each slave.

## 5. FIPIO Objects Available for Programming

FIPIO object	Description (example)
%IW\p.2.c\0.0	Slave n°1: Control n°1
%IW\p.2.c\0.0.1	Slave n°1: Control n°2
%IW\p.2.c\0.0.2	Slave n°1: Control n°3
%IW\p.2.c\0.0.3	Slave n°2: Control n°1
%IW\p.2.c\0.0.4	Slave n°2: Control n°2
%IW\p.2.c\0.0.5	Slave n°3: Control n°1
%IW\p.2.c\0.0.6	Slave n°4: Control n°1
%IW\p.2.c\0.0.7	Slave n°4: Control n°2
%IW\p.2.c\0.0.8	Not used (18 words)
..... %IW\p.2.c\0.0.25	
%IW\p.2.c\0.0.26	Reserved

FIPIO object	Description (example)
%QW\p.2.c\0.0	Slave n°1: Command n°1
%QW\p.2.c\0.0.1	Slave n°2: Command n°1
%QW\p.2.c\0.0.2	Slave n°2: Command n°2
%QW\p.2.c\0.0.3	Slave n°2: Command n°3
%QW\p.2.c\0.0.4	Slave n°3: Command n°1
%QW\p.2.c\0.0.5	Slave n°3: Command n°2
%QW\p.2.c\0.0.6	Slave n°4: Command n°1
%QW\p.2.c\0.0.7	Not used (19 words)
..... %QW\p.2.c\0.0.25	
%QW\p.2.c\0.0.26	Reserved
%QW\p.2.c\0.0.27	Reserved

### 5.1.3. Configuration under ABC-LUFP Config Tool with Profile FED C32

The actual contents of the periodic command and control words described below will depend entirely on the Modbus data / gateway memory associations you configured in ABC-LUFP Config Tool. These associations involve the “Data” or “Preset Data” elements in the Modbus query and response frames. See Chapter 7, for the various possibilities offered by this gateway configuration tool, especially in conjunction with profile FED C32.

#### Mapping of FIPIO PLC INPUTS to LUFP1 gateway memory

FIPIO object	Memory
%IW\p.2.c\0.0	0x0000-0x0001
%IW\p.2.c\0.0.1	0x0002-0x0003
%IW\p.2.c\0.0.2	0x0004-0x0005
%IW\p.2.c\0.0.3	0x0006-0x0007
%IW\p.2.c\0.0.4	0x0008-0x0009
%IW\p.2.c\0.0.5	0x000A-0x000B
%IW\p.2.c\0.0.6	0x000C-0x000D
%IW\p.2.c\0.0.7	0x000E-0x000F
%IW\p.2.c\0.0.8	0x0010-0x0011

FIPIO object	Memory
%IW\p.2.c\0.0.9	0x0012-0x0013
%IW\p.2.c\0.0.10	0x0014-0x0015
%IW\p.2.c\0.0.11	0x0016-0x0017
%IW\p.2.c\0.0.12	0x0018-0x0019
%IW\p.2.c\0.0.13	0x001A-0x001B
%IW\p.2.c\0.0.14	0x001C-0x001D
%IW\p.2.c\0.0.15	0x001E-0x001F
%IW\p.2.c\0.0.16	0x0020-0x0021
%IW\p.2.c\0.0.17	0x0022-0x0023

FIPIO object	Memory
%IW\p.2.c\0.0.18	0x0024-0x0025
%IW\p.2.c\0.0.19	0x0026-0x0027
%IW\p.2.c\0.0.20	0x0028-0x0029
%IW\p.2.c\0.0.21	0x002A-0x002B
%IW\p.2.c\0.0.22	0x002C-0x002D
%IW\p.2.c\0.0.23	0x002E-0x002F
%IW\p.2.c\0.0.24	0x0030-0x0031
%IW\p.2.c\0.0.25	0x0032-0x0033

#### Mapping of FIPIO PLC OUTPUTS to LUFP1 gateway memory

FIPIO object	Memory
%QW\p.2.c\0.0	0x0200-0x0201
%QW\p.2.c\0.0.1	0x0202-0x0203
%QW\p.2.c\0.0.2	0x0204-0x0205
%QW\p.2.c\0.0.3	0x0206-0x0207
%QW\p.2.c\0.0.4	0x0208-0x0209
%QW\p.2.c\0.0.5	0x020A-0x020B
%QW\p.2.c\0.0.6	0x020C-0x020D
%QW\p.2.c\0.0.7	0x020E-0x020F
%QW\p.2.c\0.0.8	0x0210-0x0211

FIPIO object	Memory
%QW\p.2.c\0.0.9	0x0212-0x0213
%QW\p.2.c\0.0.10	0x0214-0x0215
%QW\p.2.c\0.0.11	0x0216-0x0217
%QW\p.2.c\0.0.12	0x0218-0x0219
%QW\p.2.c\0.0.13	0x021A-0x021B
%QW\p.2.c\0.0.14	0x021C-0x021D
%QW\p.2.c\0.0.15	0x021E-0x021F
%QW\p.2.c\0.0.16	0x0220-0x0221
%QW\p.2.c\0.0.17	0x0222-0x0223

FIPIO object	Memory
%QW\p.2.c\0.0.18	0x0224-0x0225
%QW\p.2.c\0.0.19	0x0226-0x0227
%QW\p.2.c\0.0.20	0x0228-0x0229
%QW\p.2.c\0.0.21	0x022A-0x022B
%QW\p.2.c\0.0.22	0x022C-0x022D
%QW\p.2.c\0.0.23	0x022E-0x022F
%QW\p.2.c\0.0.24	0x0230-0x0231
%QW\p.2.c\0.0.25	0x0232-0x0233

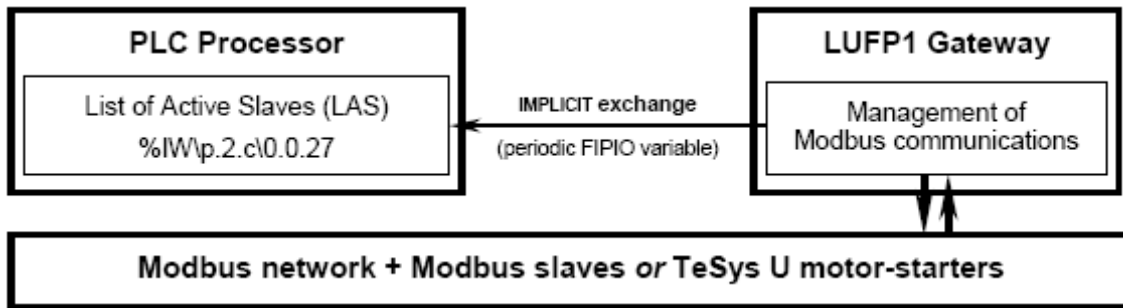
**NOTE:** Unlike with profile FED C32 P, you are not restricted to a maximum total of 26 periodic words. You can use all 26 input words and all 26 output words in the two tables above, organizing them in the gateway memory as you see fit, within the limit of the addresses shown.

You are still limited to 26 Modbus commands, however, but you can read or write more than one word with each of these commands.

# 5. FIPIO Objects Available for Programming

## 5.2. List of Active Slaves (LAS) Service

This service consists of a single implicit input word (%IW\p.2.c\0.0.27) in which each bit (X0 to X7) corresponds to a Modbus slave. Each bit will have the value 1 if the corresponding Modbus slave is present and active on the Modbus network.



Bit	Standard configuration (8 TeSys U devices)	Configuration under PL7 PRO (profile FED C32 P)	Configuration under ABC- LUF1 Config Tool (profile FED C32)
X0	TeSys U motor starter n°1	Modbus slave n°1	1st Modbus slave
X1	TeSys U motor starter n°2	Modbus slave n°2	2nd Modbus slave
X2	TeSys U motor starter n°3	Modbus slave n°3	3rd Modbus slave
X3	TeSys U motor starter n°4	Modbus slave n°4	4th Modbus slave
X4	TeSys U motor starter n°5	Modbus slave n°5	5th Modbus slave
X5	TeSys U motor starter n°6	Modbus slave n°6	6th Modbus slave
X6	TeSys U motor starter n°7	Modbus slave n°7	7th Modbus slave
X7	TeSys U motor starter n°8	Modbus slave n°8	8th Modbus slave
X8-X15	Not used (bits at 0)		

In the event of no response, or an invalid response, from a Modbus slave (after the timeout and subsequent retransmissions of the Modbus command) the corresponding bit switches to 0. It will return to 1 as soon as the gateway receives a valid response from the slave.

If several periodic words (or several Modbus commands) are configured for the same Modbus slave, then it only requires one of the words to generate valid communications for the corresponding LAS bit to be set to 1.

**NOTE:**

- You must configure at least one periodic command or control word per slave (profile FED C32 P) or at least one Modbus command per slave (ABC-LUF1 Config Tool and profile FED C32). If not, the slave will be considered permanently absent (bit at 0).
- If you want to communicate with a Modbus slave using the PKW service only, be aware that the LAS and PKW services are totally disassociated: a PKW response from a Modbus slave will not set to 1 the corresponding bit in word %IW\p.2.c\0.0.27.

**Standard configuration (8 TeSys U devices):** If you delete any motor starters from the configuration (see Section 4.2.6), you will need to adapt the above table accordingly. If, for example, you delete 3 motor starters (it has to be the last 3, whichever Modbus addresses are used), the bits corresponding to TeSys U motor starters n°6, 7 and 8—namely bits X5-X7—will take on the value “Not used (bits at 0)”.



# 5. FIPIO Objects Available for Programming

**Configuration under PL7 PRO (profile FED C32 P):** The absence from the configuration of one or more Modbus slaves entails the same modifications as for the standard configuration of 8 TeSys U devices.

**Configuration under ABC-LUFP Config Tool (profile FED C32):** The order of the slaves in the LAS word corresponds to the declaration order for the various “Nodes” in the “Sub-Network” element, as they appear in ABCLUFP Config Tool. As in the two previous cases, if fewer than 8 Modbus slaves are used, then one or more slaves— starting with the 8th and working back towards the 1st—will be absent from the list.

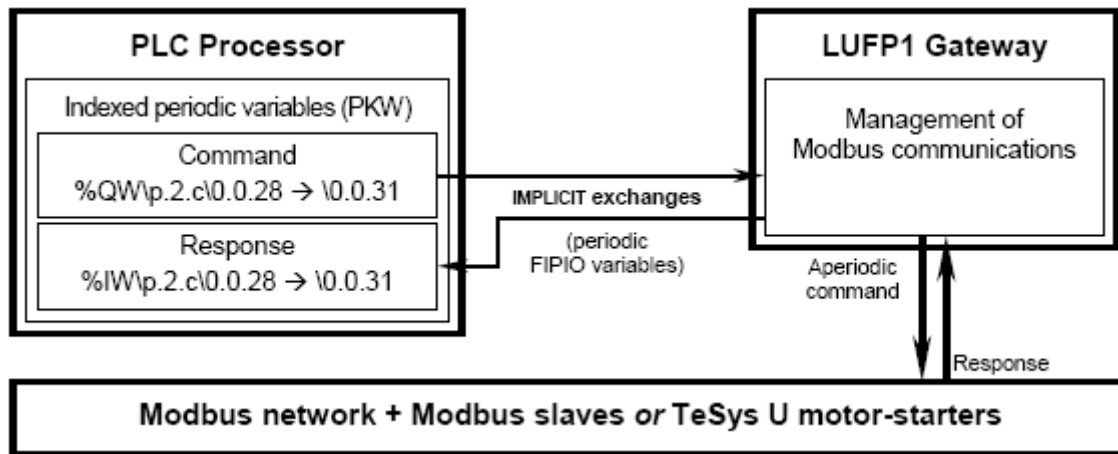
## 5.3. Indexed Periodic Variables (PKW) Service

### 5.3.1. Introduction to the PKW Service

With this service, you can configure and adjust any Modbus slave: it offers read and write access to any parameter of a slave via the periodic inputs and outputs of LUFP1 gateway. It can also be used for access to certain reserved registers on the LUFP1 gateway.

The PKW service is aperiodic on the Modbus network (a command is transmitted whenever one of the %QW outputs associated with the service changes) and periodic on the FIPIO network (implicit exchanges). You can use the service to perform the following tasks:

- Retrieving or updating the parameters on some or all of the Modbus slaves;
- Reading the value of one or more data for which a low refresh rate is sufficient;
- Reading the value of a data item when an event is signaled by a periodic variable.



In the absence of any response from the queried Modbus slave using this service, the gateway retransmits the same command 3 times in succession, waiting for 1 second between transmissions. If you wish, you can modify the values of these two parameters (number of retransmissions and timeout) using ABC-LUFP Config Tool (see Section 7.13.3).



## 5. FIPIO Objects Available for Programming

---

### 5.3.3. Using the Indexed Periodic Variables (PKW) Service

To perform a complete transaction, the FIPIO master must first update the whole of the command (%QW\p.2.c\0.0.28 to %QW\p.2.c\0.0.31), and then wait until the values in the response (%IW\p.2.c\0.0.28 to %IW\p.2.c\0.0.31) match the command (by comparing the PKE, DN and R/W—or, in the case of the response, R/W/N—values).

The gateway stores the response values for as long as the FIPIO master modifies *none* of the command values. If any output word in the PKW command (even the PWE outputs) is modified, a new command is transmitted over the Modbus network, except when R/W = 0x00.

#### NOTE:

- Stick strictly to the values given in the PKW service command table. Otherwise the gateway will send an erroneous response.
- Be careful not to write incorrect values in the outputs corresponding to the PKW service, as this would generate an incoherent command: only limited checks are run on the data used by this service, and it is therefore left to the FIPIO master PLC application to handle them.
- The PKE addresses corresponding to periodic command words (profile FED C32 P) or to registers assigned by Modbus write commands (profile FED C32) must not be used in PKW service write commands, as this could lead to a write conflict in the same registers.
- Such a command could however be envisaged if, for example, you wanted to run the same command on all the Modbus slaves *urgently and simultaneously* (DN = 255). In this case, you will also need to write the same value as the broadcast command PWE into the periodic command words that share the same PKE address, in order to avoid generating conflicts between the values transmitted.
- As the write commands (R/W = 0x57 and 0x58) return no value, you will have to use the appropriate read command (R/W = 0x52 or 0x53) if you want to check that the write operation has been performed correctly.

Below are further details on the most important data in this service:

#### Device Number (DN):

- DN = 0            1st Modbus slave. This is TeSys U motor starter n°1, or Modbus slave n°1 when profile FED C32 P is used. When profile FED C32 is used, it stands for the first node configured in the “Sub-Network” element under ABC-LUFP Config Tool. In either case, the gateway will use the true physical Modbus address every time to query the “1st slave”.
- DN = 1 to 247    Physical address of the queried Modbus slave. The queried slave *must* be one of the Modbus slaves configured under PL7 PRO (profile FED C32 P) or under ABC-LUFP Config Tool (profile FED C32). By querying one or more of the slaves episodically via the PKW service, you can avoid having to configure periodic words (under PL7 PRO) or Modbus commands (under ABC-LUFP Config Tool).
- DN = 254            LUFP1 gateway. With this reserved address you can read the values of certain registers on the gateway (see Section 5.3.5). All the registers on the gateway are in 8-bit format and their values will be returned to bits 0-7 of word %IW\p.2.c\0.0.30.

Some of these registers cannot be written using the PKW service; they are noted as being “Read Only”, or “RO”, and only the 16-bit word read command (R/W = 0x52) can be used with them. All the other gateway's internal registers can be read and written using the PKW service; they are noted as being “Read/Write”, or “R/W”, and the 16-bit word read and write commands (R/W = 0x52 or 0x57) can be used with them.

## 5. FIPIO Objects Available for Programming

---

DN = 255      Broadcast to all Modbus slaves. This DN value must only be used for write operations (R/W/N = 0x4E in the event of a read operation). On the Modbus network, the command generated uses the address 0: this means that all the Modbus slaves must accept the command, although none of them will acknowledge it.

The drawback with the broadcast command is that the gateway has no way of checking whether the Modbus slaves have received the command. Even if there are no slaves present on the Modbus network, the gateway will still return a valid PKW response (R/W/N = OK).

**NOTE:** This feature can be particularly useful if you need to perform an urgent command on *all* the slaves *at the same time*.

DN = ...      Incorrect address. Any value other than those indicated above will elicit an error code from the gateway (R/W/N = 0x4E).

### Read / Write (R/W):

R/W = 0x00 ..... Reset command and response. As long as R/W remains at 0x00, no command will be generated on the Modbus network (even if one of the other PKW service outputs is modified) and *all* the PKW service inputs will remain at 0x0000. Basically, you should use this value if you want to deactivate the PKW service.

**NOTE:** As the PKW service only performs single read and write operations (the active command is only executed once), you could use this specific value of R/W to repeatedly overwrite the active R/W value. This would generate a series of identical commands (interspersed with resets to zero), thereby enabling the gateway to emulate the principle of permanent read and write operations.

R/W = 0x52 ..... Read a 16-bit word. This is also the command to use for reading an 8-bit register from the gateway (DN = 254).

R/W = 0x53 ..... Read 2 consecutive 16-bit words *or* a 32-bit Dword. This command is used mainly for reading the values of two 16-bit registers located at two consecutive addresses on the same Modbus slave. The value located at address PKE will be returned in the input %IW\p.2.c\0.0.30 (1st word of PWE) and the value at address PKE+1 in the input %IW\p.2.c\0.0.31 (2nd word of PWE).

**NOTE:** This command should be preferred to the 16-bit word read command (R/W = 0x52) when using the PKW service to read a large number of consecutive registers on a Modbus slave.

R/W = 0x57 ..... Write a 16-bit word. This command allows you to modify the value of the 16-bit register of which the address is given by the PKE output. Only the 1st word of the PWE output is used (%QW\p.2.c\0.0.30).

R/W = 0x58 ..... Write 2 consecutive 16-bit words *or* a 32-bit Dword. This command is used mainly for modifying the value of two 16-bit registers located at two consecutive addresses on the same Modbus slave. The value of output %QW\p.2.c\0.0.30 (1st word of PWE) is written into address PKE and the value of output %QW\p.2.c\0.0.31 (2nd word of PWE) is written into address PKE+1.

**NOTE:** This command should be preferred to the 16-bit word write command (R/W = 0x57) when using the PKW service to update a large number of consecutive registers on a Modbus slave.

## 5. FIPIO Objects Available for Programming

### Read / Write / Error (R/W/N):

R/W/N = 0x4E ⇒ This specific value enables the gateway to signal a PKW command runtime error to the FIPIO master. In this case, the gateway uses the LSB of the 1st PWE word in the response to return an error code to the FIPIO master:

- 0x04 if no Modbus response, or an erroneous Modbus response, is received.
- 0x01 to 0x08 (including 0x04) if an exception response is received from the Modbus slave. This value will correspond to the exception code (see Appendix E).

R/W/N = 0x•• ⇒ Copies the value of the R/W command register (see above), thereby also acknowledging the command.

### 5.3.4. Sample Uses of PKW Service

The examples below illustrate commands addressed to Modbus slaves consisting mainly of the 8 TeSys U motor starters described in Chapter 4.

- **Sample reading of the value of a motor starter register (16-bit word)**

The 1st fault register (address = 452) is read on TeSys U motor starter n°5 (address = 5). The result of the read operation is 0x0002 (presence of a magnetic fault only). Only the 1st PWE word in the response is used; the 2nd word is forced to 0x0000 by the gateway.

PKW Command			PKW Response		
Output	Value	Meaning	Input	Value	Meaning
%QWp.2.cI0.0.28	0x01C4	PKE = 452	%IWp.2.cI0.0.28	0x01C4	PKE = 452
%QWp.2.cI0.0.29	0x0552	DN = 0x05 R/W = 0x52 (read)	%IWp.2.cI0.0.29	0x0552	DN = 0x05 R/W/N = 0x52 (read OK)
%QWp.2.cI0.0.30	0x0000	PWE = 0x••••	%IWp.2.cI0.0.30	0x0002	PWE = 0x0002 (used)
%QWp.2.cI0.0.31	0x0000	0x••••	%IWp.2.cI0.0.31	0x0000	0x0000 (not used)

- **Sample writing of the value of a motor starter register (16-bit word)**

The 2nd command register (address = 705) is written to TeSys U motor starter n°7 (address = 7) with the value 0x0006 (clear statistics + reset thermal memory).

The result of the write operation is a simple acknowledgement, as no value is returned in the PWE of the response; instead of inserting a value, the gateway forces both words of the PWE to 0x0000.

### PKW Command PKW Response

PKW Command			PKW Response		
Output	Value	Meaning	Input	Value	Meaning
%QWp.2.cI0.0.28	0x02C1	PKE = 705	%IWp.2.cI0.0.28	0x02C1	PKE = 705
%QWp.2.cI0.0.29	0x0757	DN = 0x07 R/W = 0x57 (write)	%IWp.2.cI0.0.29	0x0757	DN = 0x07 R/W/N = 0x57 (write OK)
%QWp.2.cI0.0.30	0x0002	PWE = 0x0002	%IWp.2.cI0.0.30	0x0000	PWE = 0x0000 (not used)
%QWp.2.cI0.0.31	0x0000	0x••••	%IWp.2.cI0.0.31	0x0000	0x0000 (not used)

- **Sample reading of the value of two consecutive motor starter registers (16-bit words)**

The operating duration (addresses = 119 for the MSB and 120 for the LSB) is read on TeSys U motor starter n°8 (address = 8).

The result of the read operation (value of the register at 119 and value of the register at 120) is 0x000D and 0xC9C4 (total operating duration = 0x000DC9C4 = 903,620 seconds, or 251.0055 hours). Both of the response PWE words are therefore used.

## 5. FIPIO Objects Available for Programming

PKW Command			PKW Response		
Output	Value	Meaning	Input	Value	Meaning
%QWp.2.cI0.0.28	0x0077	PKE = 119	%IWp.2.cI0.0.28	0x0077	PKE = 119
%QWp.2.cI0.0.29	0x0853	DN = 0x08 R/W = 0x53 (read32)	%IWp.2.cI0.0.29	0x0853	DN = 0x08 R/W/N = 0x53 (read32 OK)
%QWp.2.cI0.0.30	0x0000	PWE = 0x****	%IWp.2.cI0.0.30	0x000D	PWE = 0x000D (used)
%QWp.2.cI0.0.31	0x0000	0x****	%IWp.2.cI0.0.31	0xC9C4	0xC9C4 (used)

- **Sample writing of the value of two consecutive motor starter registers (16-bit words)**

The timeout and threshold for the thermal reset (addresses = 607 and 608) are written to TeSys U motor starter n°2 (address = 2) with the following values: 90 s (0x005A) and 60% (0x003C).

The result of the write operation is a simple acknowledgement, as no value is returned in the PWE of the response; instead of inserting one or two values, the gateway forces both words of the PWE to 0x0000.

PKW Command			PKW Response		
Output	Value	Meaning	Input	Value	Meaning
%QWp.2.cI0.0.28	0x025F	PKE = 607	%IWp.2.cI0.0.28	0x025F	PKE = 607
%QWp.2.cI0.0.29	0x0258	DN = 0x02 R/W = 0x58 (write32)	%IWp.2.cI0.0.29	0x0258	DN = 0x02 R/W/N = 0x58 (write32 OK)
%QWp.2.cI0.0.30	0x005A	PWE = 0x005A	%IWp.2.cI0.0.30	0x0000	PWE = 0x0000 (not used)
%QWp.2.cI0.0.31	0x003C	0x003C	%IWp.2.cI0.0.31	0x0000	0x0000 (not used)

- **Sample erroneous reading of the value of a motor starter register (16-bit word)**

An attempt is made to read the status register value (address = 455 for a TeSys U motor starter) on a slave whose physical address corresponds to none of the addresses configured for the gateway's Modbus slaves (address = 32 = 0x20).

The response is an error code (R/W/N = 0x4E) and is immediate: the gateway signals to the FIPIO master that the queried slave does not exist (1st word of PWE = error code = 0x0002).

PKW Command			PKW Response		
Output	Value	Meaning	Input	Value	Meaning
%QWp.2.cI0.0.28	0x01C7	PKE = 455	%IWp.2.cI0.0.28	0x01C7	PKE = 455
%QWp.2.cI0.0.29	0x2052	DN = 0x20 R/W = 0x52 (read)	%IWp.2.cI0.0.29	0x204E	DN = 0x20 R/W/N = 0x4E (ERROR)
%QWp.2.cI0.0.30	0x0000	PWE = 0x****	%IWp.2.cI0.0.30	0x0002	PWE = 0x0002 (error code)
%QWp.2.cI0.0.31	0x0000	0x****	%IWp.2.cI0.0.31	0x0000	0x0000 (not used)

- **Sample erroneous writing of the value of a motor starter register (16-bit word)**

An attempt is made to write the status register value (address = 455) to TeSys U motor starter n°2 (address = 2) with the value 0x0001, but the register is in read-only mode.

The response is an error code (R/W/N = 0x4E): the motor starter returns an exception response (1st word of PWE = error code = Modbus exception code = 0x0002 = Illegal Data Address).

PKW Command			PKW Response		
Output	Value	Meaning	Input	Value	Meaning
%QWp.2.cI0.0.28	0x01C7	PKE = 455	%IWp.2.cI0.0.28	0x01C7	PKE = 455
%QWp.2.cI0.0.29	0x0357	DN = 0x03 R/W = 0x57 (write)	%IWp.2.cI0.0.29	0x034E	DN = 0x03 R/W/N = 0x4E (ERROR)
%QWp.2.cI0.0.30	0x0001	PWE = 0x0001	%IWp.2.cI0.0.30	0x0002	PWE = 0x0002 (exception code)
%QWp.2.cI0.0.31	0x0000	0x****	%IWp.2.cI0.0.31	0x0000	0x0000 (not used)

## 5. FIPIO Objects Available for Programming

- **Sample writing of the value of a register (16-bit word) on all the motor starters (broadcast)**

The command register (address = 704) is written to all the TeSys U motor starters (address = broadcast = 255) with the value 0x2000 (Stop and Pause).

**NOTE:** This simultaneously switches off all the TeSys U motor starters, but you still need to insert the value 0x2000 in the periodic output words sent to the motor starters, otherwise they will return to their previous status at the next Modbus polling cycle.

The result of the write operation is a simple acknowledgement, as no value is returned in the PWE of the response; instead of inserting a value, the gateway forces both words of the PWE to 0x0000.

PKW Command			PKW Response		
Output	Value	Meaning	Input	Value	Meaning
%QWp.2.cI0.0.28	0x02C0	PKE = 704	%IWp.2.cI0.0.28	0x02C0	PKE = 704
%QWp.2.cI0.0.29	0xFF57	DN = 0xFF R/W = 0x57 (write)	%IWp.2.cI0.0.29	0xFF57	DN = 0xFF R/W/N = 0x57 (write OK)
%QWp.2.cI0.0.30	0x2000	PWE = 0x2000	%IWp.2.cI0.0.30	0x0000	PWE = 0x0000 (not used)
%QWp.2.cI0.0.31	0x0000	0x****	%IWp.2.cI0.0.31	0x0000	0x0000 (not used)

- **Sample reading of the value of a register (8-bit byte) on the LUF1 gateway**

The last error code from Modbus slave n°1 is read on the gateway (address = 300 and DN = 0xFE).

If Modbus slave n°1 (TeSys U motor starter n°1 in the case of the standard configuration) has already been disconnected from the Modbus network, the gateway will have assigned it the error code 0x04. In this case, the result of the read operation is 0x••04. As the value of the register read is contained in the LSB of the 1st word in the response PWE, the corresponding MSB is forced to 0x00 by the gateway; likewise, the 2nd word is forced to 0x0000.

PKW Command			PKW Response		
Output	Value	Meaning	Input	Value	Meaning
%QWp.2.cI0.0.28	0x012C	PKE = 300	%IWp.2.cI0.0.28	0x012C	PKE = 300
%QWp.2.cI0.0.29	0xFE52	DN = 0xFE R/W = 0x52 (read)	%IWp.2.cI0.0.29	0xFE52	DN = 0xFE R/W/N = 0x52 (read OK)
%QWp.2.cI0.0.30	0x0000	PWE = 0x****	%IWp.2.cI0.0.30	0x0004	PWE = 0x0004 (LSB used)
%QWp.2.cI0.0.31	0x0000	0x****	%IWp.2.cI0.0.31	0x0000	0x0000 (not used)

### 5.3.5. LUF1 Gateway Internal Registers

The registers presented in this section are internal to the gateway and are all in 8-bit format. The PKW service provides access to them (DN = 254), but only with the aid of a 16-bit word read command. The gateway inserts the result of a read operation in bits 0-7 of the 1st word in the PWE (%QWp.2.cI0.0.30); the rest of the PWE is forced to 0.

The “RO” access right means that the registers are in “read-only” mode, rather than “read / write” (“R/W”). But even when a register is in “R/W”, the gateway can reject a write operation via the PKW service if the value of the PWE in the write command is incorrect. This tends to happen with any value other than 0x0000 in the case of registers 300 to 307 (error code reset).

## 5. FIPIO Objects Available for Programming

### LUFF1 Gateway Internal Registers: Summary

Addresses	Rights	Registers
<b>Adjustment parameters</b> (%MW\p.2.c\0.0.20 to %MW\p.2.c\0.0.49)		
0 to 9	RO	General Modbus configuration for the gateway
10 to 21	R/W	Cycle time settings for command and control words
22 to 29	RO	Physical addresses of Modbus slaves
30 to 59	RO	Registers not used (0x●●)
<b>Configuration parameters</b> (%KW\p.2.c\0.0 to %KW\p.2.c\0.0.29)		
60 to 67	RO	Number of periodic command and control words
68 to 119	R/W	Addresses of each of the command and control words
<b>Registers not used</b>		
120 to 299	RO	Not used (values equal to 0x00)
<b>Error codes relating to Modbus slaves n°1 to 8</b>		
300 to 307	R/W	Error codes relating to Modbus slaves n°1 to 8 (last error)

The adjustment and configuration parameters will only be present when the gateway is configured using PL7 PRO, i.e. in accordance with the standard profile FED C32 P. You should therefore not use addresses 0 to 119 if you configure the gateway with ABC-LUFF Config Tool (profile FED C32 cannot be configured under PL7 PRO).

These parameters are shown below with the sole purpose of listing their addresses and access rights to the PKW service. For more details about them, see Appendix B. To make it easier to read, the table below is based on the FIPIO objects that correspond to the gateway's configuration and adjustment parameters:

### LUFF1 Gateway Internal Registers: Details

FIPIO object	Bits 0 to 7 (LSB)			Bits 8 to 15 (MSB)		
	Add.	Rights	Name	Add.	Rights	Name
%MW\p.2.c\0.0.20	0	RO	Modbus speed	1	RO	Modbus format
%MW\p.2.c\0.0.21	2	RO	Modbus response timeout	3	RO	Number of retransmissions
%MW\p.2.c\0.0.22	4	RO	Modbus reconnection time	5	RO	Not used
%MW\p.2.c\0.0.23	6	RO	Behavior on loss of FIPIO	7	RO	Behavior on loss of Modbus
%MW\p.2.c\0.0.24	8	RO	Not used	9	RO	Not used
%MW\p.2.c\0.0.25	10	R/W	Fast cycle: Commands	11	R/W	Normal cycle: Commands
%MW\p.2.c\0.0.26	12	R/W	Fast cycle: Controls	13	R/W	Normal cycle: Controls
%MW\p.2.c\0.0.27	14	R/W	Speed: Commands 1 to 8	15	R/W	Speed: Commands 9 to 16
%MW\p.2.c\0.0.28	16	R/W	Speed: Commands 17 to 24	17	R/W	Speed: Commands 25 to 26
%MW\p.2.c\0.0.29	18	R/W	Speed: Controls 1 to 8	19	R/W	Speed: Controls 9 to 16
%MW\p.2.c\0.0.30	20	R/W	Speed: Controls 17 to 24	21	R/W	Speed: Controls 25 to 26
%MW\p.2.c\0.0.31	22	RO	Address of Modbus slave n°1	23	RO	Address of Modbus slave n°2
%MW\p.2.c\0.0.32	24	RO	Address of Modbus slave n°3	25	RO	Address of Modbus slave n°4
%MW\p.2.c\0.0.33	26	RO	Address of Modbus slave n°5	27	RO	Address of Modbus slave n°6
%MW\p.2.c\0.0.34	28	RO	Address of Modbus slave n°7	29	RO	Address of Modbus slave n°8
%MW\p.2.c\0.0.35	30	RO	Not used	31	RO	Not used
.....	..	...	.....	..	...	.....
%MW\p.2.c\0.0.49	58	RO	Not used	59	RO	Not used



## 5. FIPIO Objects Available for Programming

FIPIO object	Bits 0 to 7 (LSB)			Bits 8 to 15 (MSB)		
	Add.	Rights	Name	Add.	Rights	Name
%KWIp.2.cI0.0	60	RO	Number of commands: slaves 1-2	61	RO	Number of commands: slaves 3-4
%KWIp.2.cI0.0.1	62	RO	Number of commands: slaves 5-6	63	RO	Number of commands: slaves 7-8
%KWIp.2.cI0.0.2	64	RO	Number of controls: slaves 1-2	65	RO	Number of controls: slaves 3-4
%KWIp.2.cI0.0.3	66	RO	Number of controls: slaves 5-6	67	RO	Number of controls: slaves 7-8
	Bits 8 to 15 (MSB)			Bits 0 to 7 (LSB)		
%KWIp.2.cI0.0.4	68	R/W	Com/con address n° 1 (MSB)	69	R/W	Com/con address n° 1 (LSB)
%KWIp.2.cI0.0.5	70	R/W	Com/con address n° 2 (MSB)	71	R/W	Com/con address n° 2 (LSB)
%KWIp.2.cI0.0.6	72	R/W	Com/con address n° 3 (MSB)	73	R/W	Com/con address n° 3 (LSB)
%KWIp.2.cI0.0.7	74	R/W	Com/con address n° 4 (MSB)	75	R/W	Com/con address n° 4 (LSB)
%KWIp.2.cI0.0.8	76	R/W	Com/con address n° 5 (MSB)	77	R/W	Com/con address n° 5 (LSB)
%KWIp.2.cI0.0.9	78	R/W	Com/con address n° 6 (MSB)	79	R/W	Com/con address n° 6 (LSB)
%KWIp.2.cI0.0.10	80	R/W	Com/con address n° 7 (MSB)	81	R/W	Com/con address n° 7 (LSB)
%KWIp.2.cI0.0.11	82	R/W	Com/con address n° 8 (MSB)	83	R/W	Com/con address n° 8 (LSB)
%KWIp.2.cI0.0.12	84	R/W	Com/con address n° 9 (MSB)	85	R/W	Com/con address n° 9 (LSB)
%KWIp.2.cI0.0.13	86	R/W	Com/con address n°10 (MSB)	87	R/W	Com/con address n°10 (LSB)
%KWIp.2.cI0.0.14	88	R/W	Com/con address n°11 (MSB)	89	R/W	Com/con address n°11 (LSB)
%KWIp.2.cI0.0.15	90	R/W	Com/con address n°12 (MSB)	91	R/W	Com/con address n°12 (LSB)
%KWIp.2.cI0.0.16	92	R/W	Com/con address n°13 (MSB)	93	R/W	Com/con address n°13 (LSB)
%KWIp.2.cI0.0.17	94	R/W	Com/con address n°14 (MSB)	95	R/W	Com/con address n°14 (LSB)
%KWIp.2.cI0.0.18	96	R/W	Com/con address n°15 (MSB)	97	R/W	Com/con address n°15 (LSB)
%KWIp.2.cI0.0.19	98	R/W	Com/con address n°16 (MSB)	99	R/W	Com/con address n°16 (LSB)
%KWIp.2.cI0.0.20	100	R/W	Com/con address n°17 (MSB)	101	R/W	Com/con address n°17 (LSB)
%KWIp.2.cI0.0.21	102	R/W	Com/con address n°18 (MSB)	103	R/W	Com/con address n°18 (LSB)
%KWIp.2.cI0.0.22	104	R/W	Com/con address n°19 (MSB)	105	R/W	Com/con address n°19 (LSB)
%KWIp.2.cI0.0.23	106	R/W	Com/con address n°20 (MSB)	107	R/W	Com/con address n°20 (LSB)
%KWIp.2.cI0.0.24	108	R/W	Com/con address n°21 (MSB)	109	R/W	Com/con address n°21 (LSB)
%KWIp.2.cI0.0.25	110	R/W	Com/con address n°22 (MSB)	111	R/W	Com/con address n°22 (LSB)
%KWIp.2.cI0.0.26	112	R/W	Com/con address n°23 (MSB)	113	R/W	Com/con address n°23 (LSB)
%KWIp.2.cI0.0.27	114	R/W	Com/con address n°24 (MSB)	115	R/W	Com/con address n°24 (LSB)
%KWIp.2.cI0.0.28	116	R/W	Com/con address n°25 (MSB)	117	R/W	Com/con address n°25 (LSB)
%KWIp.2.cI0.0.29	118	R/W	Com/con address n°26 (MSB)	119	R/W	Com/con address n°26 (LSB)
	120	RO	Not used (0x00)	121	RO	Not used (0x00)
	...	...	.....	...	...	.....
	298	RO	Not used (0x00)	299	RO	Not used (0x00)
	300	R/W	Error code: slave n°1	301	R/W	Error code: slave n°2
	302	R/W	Error code: slave n°3	303	R/W	Error code: slave n°4
	304	R/W	Error code: slave n°5	305	R/W	Error code: slave n°6
	306	R/W	Error code: slave n°7	307	R/W	Error code: slave n°8

### CAUTION

#### RISK OF UNPROPER COMMUNICATION

Be careful while modifying the register values at addresses 68 to 119. The FIPIO master can alter the very nature of the information exchanged between the gateway and the slaves.

**Failure to follow this instruction can result in injury or equipment damage.**

**Slave error codes:** Each of the registers at addresses 300 to 307 contains the code of the last error declared by the gateway for the Modbus slave concerned. If the gateway receives an exception response from a Modbus slave (see Modbus Protocol Exception responses in Appendix E) during periodic exchanges, it puts the exception code in the corresponding register (300-307). The value of each of these registers remains the same until either a new error is signaled for the slave concerned, or you reset it to zero using the PKW service (the gateway only accepts write commands where PWE = 0x0000).

## 5. FIPIO Objects Available for Programming

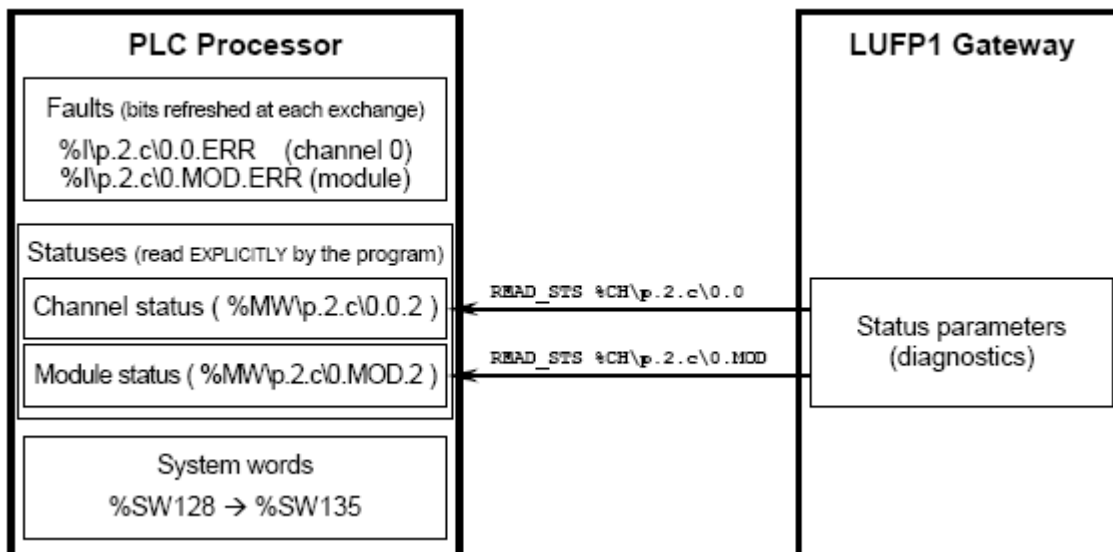
The value of the “error code” is 0x04 if the corresponding slave is declared absent by the gateway (after the Modbus periodic commands for the slave have been retransmitted with timeouts).

**NOTE:** The LAS service is updated at the same time as registers 300 to 307 when this type of event occurs (see Section 5.2). You can therefore use the various bits in the periodic word of this service (%IW\p.2.c\0.0.27) to trigger a read operation on the internal register assigned to the slave declared absent by the LAS service.

### 5.4. FIPIO Diagnostic Objects

We strongly recommend that you adhere to the following five rules while using the other FIPIO objects associated with the LUF1 gateway:

- ① Check system words %SW128 to %SW135  
Each bit in this group of words signals the status of a device connected to the FIPIO bus (addresses 0 to 127). See Section 5.4.1 for a detailed description of these system words.
- ② Check the implicit “channel fault” bit (%I\p.2.c\0.0.ERR) that monitors the connection point  
This bit, normally at 0, switches to 1 in the event of a fault or break in the FIPIO connection. If so, none of the %IW\p.2.c\0.0.\*\* inputs will be valid.
- ③ If there is a “channel fault” (see point ②), do an explicit read on the “Channel status” (%MW\p.2.c\0.0.2)  
This information is updated by the explicit command `READ_STS %CH\p.2.c\0.0;`. See Section 5.4.2 for a description of the “Channel status” bits.
- ④ Check the implicit “module fault” bit (%I\p.2.c\0.MOD.ERR) that monitors the gateway  
As with the “Channel fault”, this bit is normally at 0 but switches to 1 in the event of a fault or break in the FIPIO connection.
- ⑤ If there is a “module fault” (see point ④), do an explicit read on the “Module status” (%MW\p.2.c\0.MOD.2)  
This information is updated by the explicit command `READ_STS %CH\p.2.c\0.MOD;`. See Section 5.4.3 for a description of the “Module status” bits.



See also the PL7 PRO documentation or on-line help (Working with communications > FIPIO bus communications > Programming a FIPIO communication > Managing FIPIO faults) to learn more about the procedures for managing FIPIO faults.

## 5. FIPIO Objects Available for Programming

### 5.4.1. System Words %SW128 to %SW135

Each bit in this group of words signals the status of a device connected to the FIPIO bus (addresses 0 to 127). These bits are normally at 1, but switch to 0 in the event of a fault on the connection point (faulty FIPIO connection point). For a non-configured connection point, the corresponding bit is always at 1.

The table below maps the correspondences between these system word bits and the addresses on the FIPIO bus:

	X0	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X14	X15
%SW128:	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
%SW129:	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
%SW130:	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
%SW131:	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
%SW132:	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
%SW133:	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
%SW134:	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
%SW135:	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127

### 5.4.2. Channel Status (%MWp.2.c\0.0.2)

Each of the “channel status” bits corresponds to a fault generated either by the gateway or by the PLC. A fault is present if the corresponding bit is at 1. The “channel status” is updated by the explicit command

```
READ_STS %CH\p.2.c\0.0;
```

	Bits	Meaning
Channel status managed by the gateway	X0-X3	Reserved (bits at 0)
	X4	<b>Internal fault:</b> The gateway is considered absent. It may be out of supply or absent from the FIPIO network. If it was configured using the standard profile FED C32, it may be that an unauthorized memory address was used under ABC-LUFP Config Tool (see the warnings in sections 7.10.3 and 7.10.4).
	X5	<b>Hardware configuration fault:</b> The standard FIPIO profile selected for configuring the gateway under PL7 PRO was not the right one. This fault is signaled when profile FED C32 is used in place of profile FED C32 P, and vice versa.
	X6	<b>Communication fault with the PLC</b>
	X7	<b>Application fault:</b> The configuration and adjustment parameters defined for profile FED C32 P, which corresponds to the gateway, contain incorrect values.
Channel status managed by the PLC	X8-X15	Reserved (bits at 0)

### 5.4.3. Module Status (%MWp.2.c\0.MOD.2)

Each of the “module status” bits corresponds to a specific fault. A fault is present if the corresponding bit is at 1. The “module status” is updated by the explicit command `READ_STS %CH\p.2.c\0.MOD;`

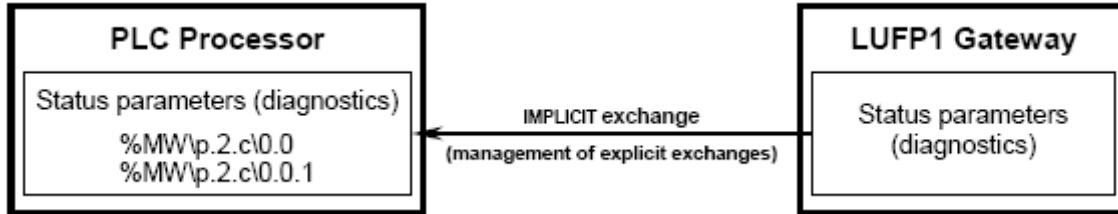
Bit	Meaning
X0	Reserved (bit at 0)
X1	<b>Functional fault:</b> Fault signaled by the gateway; reading “channel status” recommended (see above).
X2-X4	Reserved (bit at 0)

Bit	Meaning
X5	<b>Hardware configuration fault:</b> The standard FIPIO profile configured under PL7 PRO was the wrong one.
X6	<b>Gateway absent:</b> Check for an “internal fault” in the “channel status” (see above).
X7-X15	Reserved (bits at 0)

# 5. FIPIO Objects Available for Programming

## 5.5. Status of Explicit Exchanges

The two implicit words %MW\p.2.c\0.0 (exchange in progress) and %MW\p.2.c\0.0.1 (report) serve to control explicit exchanges. You should use them if you need to program explicit exchanges in the FIPIO master PLC application.



Exchange in progress (%MW\p.2.c\0.0)

Bit	Meaning
X0	Reading status (if bit at 1) Make sure this bit is at 0 before you perform a new explicit exchange
X1	Reserved (bit at 0)
X2	Sending adjustment parameters (if bit at 1)
X3-X14	Reserved (bits at 0)
X15	Sending configuration parameters (if bit at 1)

Report (%MW\p.2.c\0.0.1)

Bit	Meaning
X0	Read status: OK (if bit at 0)
X1	Reserved (bit at 0)
X2	Adjustment parameters received and accepted by gateway (if bit at 0)
X3-X14	Reserved (bits at 0)
X15	Configuration parameters received and accepted by gateway (if bit at 0)

Before carrying out an exchange, check that the relevant bit (%MW\p.2.c\0.0:X\*\*) is at 0 (exchange inactive). The bit will switch to 1 for the duration of the exchange.

When the exchange is complete (the bit switches back to 0), you can check the bit at the other end (%MW\p.2.c\0.0.1:X\*\*) to determine whether the exchange succeeded (bit at 0) or failed (bit at 1).

## 5.6. Parameters Specific to Profile FED C32 P

The gateway configuration and adjustment parameters are only stored in the FIPIO master PLC memory if the gateway was configured (under PL7 PRO) to support configuration and adjustment using standard profile FED C32 P.

**If the gateway is configured with standard profile FED C32, then these parameters do not exist.** Configuration and adjustment must then be carried out independently. In this case, ABC-LUF1 Config Tool will be required: see Chapter 7. The software implementation of the gateway under PL7 PRO, using profile FED C32, is described in Chapter 6. From the viewpoint of PL7 PRO, it then becomes a “locally configured” gateway.

**NOTE:** Some of the configuration and adjustment parameters cannot be modified after the initial setup of the gateway using the PLC. See the tables in section 5.3.5 and in Appendix B.

# 5. FIPIO Objects Available for Programming

## 5.6.1. Configuration Parameters (%KW\p.2.c\0.0 to %KW\p.2.c\0.0.29)

The 30 configuration parameters of profile FED C32 P are transmitted to the gateway implicitly when it connects up to the FIPIO network.

When you modify them in **ONLINE** mode under PL7 PRO and save the changes, communications in channel 0 are paused while the new settings are sent to the gateway.



These parameters are described in Appendix B.

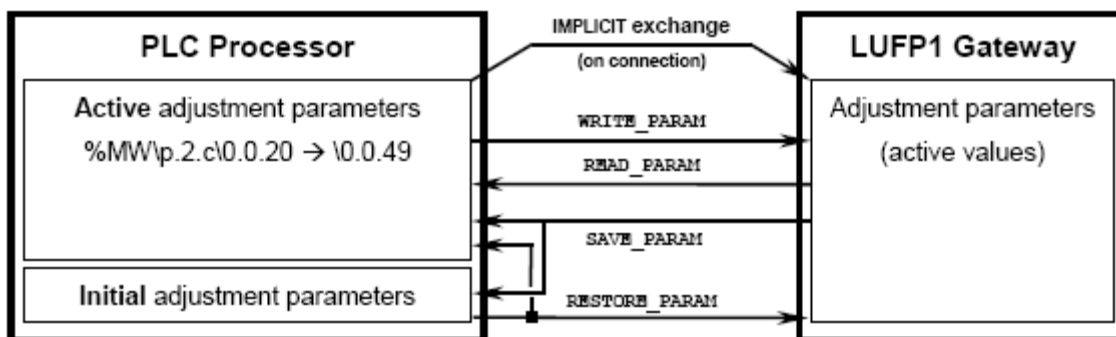
## 5.6.2. Adjustment Parameters (%MW\p.2.c\0.0.20 to %MW\p.2.c\0.0.49)

The 30 adjustment parameters of profile FED C32 P are transmitted to the gateway implicitly when it connects up to the FIPIO network.

When you modify them in **ONLINE** mode under PL7 PRO and save the changes, a command is generated to send the parameters to the gateway.

You can also use the following functions to perform explicit exchanges between the FIPIO PLC and the LUF1 gateway:

- READ\_PARAM %CH\p.2.c\0.0..... Read the adjustment parameters on the gateway.
- WRITE\_PARAM %CH\p.2.c\0.0..... Write adjustment parameters to the gateway.
- SAVE\_PARAM %CH\p.2.c\0.0..... Save the gateway adjustment parameters; the values replace the initial adjustment parameters.
- RESTORE\_PARAM %CH\p.2.c\0.0 .... Restore the initial adjustment parameters.



**NOTE:** The values of the initial adjustment parameters are those defined using the PL7 PRO configuration editor or, alternatively, the most recently saved values.

These parameters are described in Appendix B.

## 6. Advanced Implementation of the Gateway

This chapter marks the start of **Part II** of the LUFFP1 Gateway User's Manual. This part comprises Chapters 6 and 7 and focuses on the use of "ABC-LUFFP Config Tool". This configuration tool enables the gateway to be configured and adjusted independently of PL7 PRO. The possibilities offered by the tool go well beyond the limits imposed by the standard profile FED C32 P. The whole of Part II therefore supplements Chapter 4, and supersedes certain sections.

### 6.1. Introduction to Advanced Implementation of the Gateway

The Advanced Implementation of the Gateway gets around certain limits imposed by the Software Implementation of the Gateway (Chapter 4). The main differences between the two implementation methods are summarized in the table below:

	Software Implementation of the Gateway (Chapter 4)	Advanced Implementation of the Gateway (Chapter 6)
FIPIO Profile	Standard profile "FED C32 P"	Standard profile "FED C32"
Modbus slaves	Maximum of 8 Modbus slaves	
Configuration method	Configuration and adjustment under PL7 PRO These parameters are downloaded to the gateway by the PLC on FIPIO connection and / or at the request of the FIPIO master (by explicit exchange)	Configuration performed entirely in ABC-LUFFP Config tool The download is performed by ABC-LUFFP Config Tool and the configuration is stored in the gateway's static memory
Periodicity of Modbus exchanges	1 fast cycle and 1 normal cycle for write commands Ditto for read commands	Each Modbus command has its own cycle time (see below)
Modbus command transfer modes	Commands: Periodic mode, with immediate transmission following any change in the command value Controls: Periodic mode	Periodic or aperiodic communications: each Modbus command is configured individually
Number of Modbus commands	Maximum of 26 Modbus commands (whether read or write commands)	
Number of words read / written per Modbus command	Only 1 word can be read / written for each command configured	The number of words read / written by the same Modbus command is defined at the config stage, but the maximum (N max) depends on the Modbus slave
Configuration of Modbus command degraded modes	All Modbus commands are configured in the same way	Each Modbus command has its own configuration
FIPIO setup objects	Adjustment..... %MW\p.2.c\0.0.20 to %MW\p.2.c\0.0.49 Configuration ... %KW\p.2.c\0.0 to %KW\p.2.c\0.0.29	These FIPIO objects do not exist, and the corresponding internal registers are therefore not accessible via the gateway's PKW service

The main advantage of ABC-LUFFP Config Tool lies in the possibility of configuring the gateway's "Modbus scanner" more precisely than under PL7 PRO, in that each Modbus command has its own configuration.

However, using this implementation method for the LUFFP1 gateway substantially modifies the operating principle described and illustrated in Chapter 7. The differences are set out below:

## 6. Advanced ABC-LUFP Config Tool

- ⇒ The exchanges between the gateway (acting as Modbus master) and the Modbus slaves are configured entirely using “ABC-LUFP Config Tool”. This configuration tool really goes into the details (exchange timeouts, communication modes, frame content, etc.), which makes it somewhat harder to use. A chapter of this guide has therefore been given over to the subject (Chapter 7).

By configuring Modbus command queries and responses with this tool, the user creates links between part of the contents of the Modbus frames and the contents of the gateway’s physical memory (input memory for the contents of the Modbus responses and output memory for the contents of the queries).

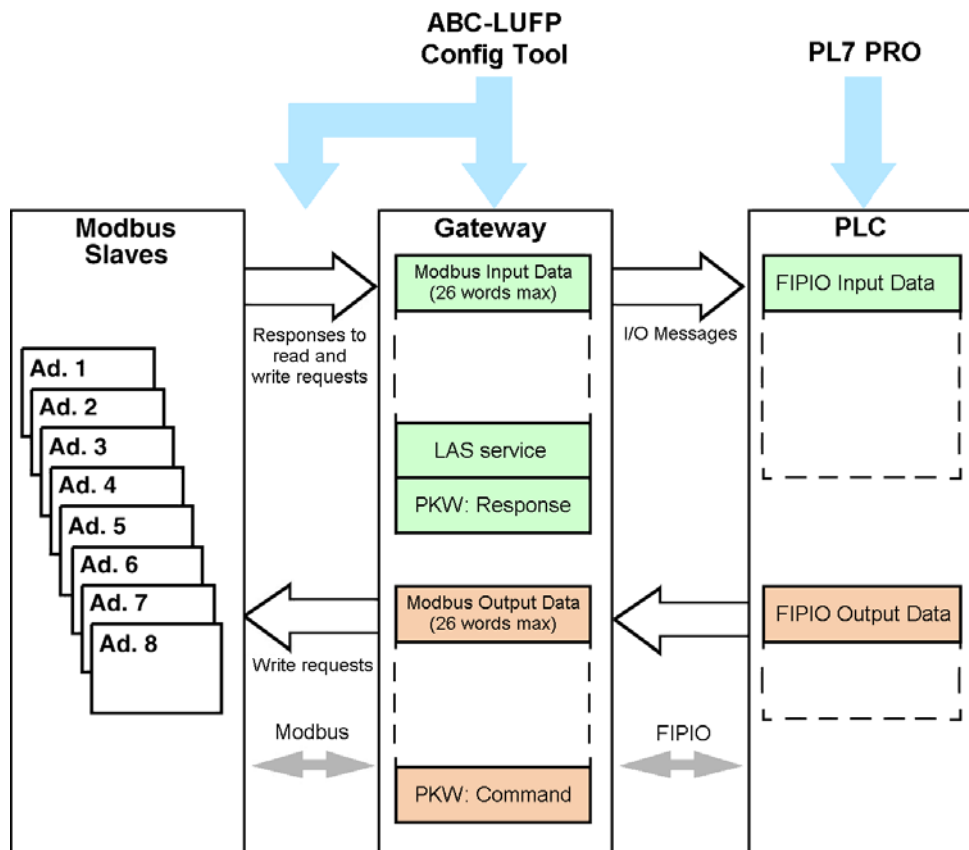
**NOTE:** Only use the address ranges indicated in this guide when creating these links.

- ⇒ The periodic exchanges between the FIPIO master PLC and the LUFP1 gateway are limited to transferring the contents of the gateway’s input memory to the %IW inputs on the PLC and transferring the PLC’s %QW outputs to the gateway’s output memory.

The FIPIO network is totally separate from the Modbus network. The frames on a network are not directly “translated” by the gateway to generate frames on the other network. Instead, the exchanges between the contents of the gateway’s memory and the Modbus slaves make up a system independent of the one entrusted with managing the exchanges between this gateway memory and the FIPIO master.

The figure that follows illustrates the independent management of each of the two networks:

— *Managing Gateway* ↔ *Modbus slave exchanges (FED C32)* —



# 6. Advanced ABC-LUFP Config Tool

---

## 6.2. Implementation under PL7 PRO

The implementation described below is based on the one in Chapter 4.2.4. We just present a summary of that implementation here, along with the modifications entailed by using profile FED C32 instead of FED C32 P:

A configuration example can be found on the <http://www.schneider-electric.com> website:

“LUFPI\_FEDC32\_Example.cfg”. The system architecture for this configuration is identical in all respects to the one described in Chapter 4. The example is provided mainly to assist you in understanding the uses and possibilities of ABC-LUFP Config Tool.

• **Adding the LUFPI Gateway under PL7 PRO**.....Section 4.2.2

⇒ Select basic module “FED C32” instead of basic module “FED C32 P”.

• **Configuring and Adjusting the LUFPI Gateway**.....Section 4.2.3

⇒ Do not double-click on basic module 

0		FED C32
---	--	---------

, as the gateway cannot be configured under PL7 PRO.

⇒ Ignore the description of how configuration and adjustment parameters are downloaded from the FIPIO master PLC to the LUFPI gateway. When ABC-LUFP Config Tool is used—and the configuration thus downloaded from ABC-LUFP Config Tool (see next steps)—the parameters will be located on the gateway only. From the viewpoint of PL7 PRO and the FIPIO master PLC, the gateway will have a “local configuration”.

• **Configuration Parameter Values, Standard Configuration**.....Section 4.2.4

⇒ Ignore this step for the time being; it describes elements which, under ABC-LUFP Config Tool, will come into play later on.

**Adjustment Parameter Values, Standard Configuration**.....Section 4.2.5

⇒ Ignore this step for the time being; it describes elements which, under ABC-LUFP Config Tool, will come into play later on.

• **Deleting One or More TeSys U Motor Starters from the Configuration** .....Section 4.2.6

⇒ This operation—described in Section 7.8—is specific to ABC-LUFP Config Tool. You must therefore run ABC-LUFP Config Tool and load the example file provided in order to delete a motor starter.

• **Configuring Gateway Inputs / Outputs** .....Section 4.2.7

⇒ As with PL7 PRO, no operation need be performed if you use the example file provided (this section describes the inputs / outputs corresponding to the configuration entered during the previous steps). If you want to modify the configuration or create a new one, consult the relevant sections of Chapter 7. The explanations that follow seek to describe the configuration illustrated in the example.

⇒ The gateway inputs / outputs are configured entirely under ABC-LUFP Config Tool. The Modbus frames configured for the various slaves (or “Nodes”) of the Modbus network (or “Sub-Network”) include “Data” or “Preset Data” elements. Each element is a link between the contents of the Modbus frame it represents and a precise location in the gateway memory. You can configure the size of the data exchanged, and the memory location for the exchanges. This allows you to organize the gateway inputs and outputs as you see fit, within the limit of the addresses that correspond to the inputs / outputs (see Section 5.1.3).

⇒ The configuration example provided with the gateway, “LUFPI\_FEDC32\_Example.cfg”, is designed to reproduce the standard configuration used in the gateway implementation with profile FED C32 P (8 TeSys U motor starters). The configuration of the gateway inputs / outputs is therefore identical to the one described for the standard configuration. Each “TeSys U n°” node in the example comprises a read command from the TeSys U status register (address = 455 = 0x01C7) and a write command from the TeSys U command register (address = 704 = 0x2C0). These commands are configured in exactly the same way, except for their “Data” elements. The “Modbus frame content / gateway memory address” links for these elements, on which the equivalence with the standard configuration is based, are configured as follows:



## 6. Advanced ABC-LUFP Config Tool

Complete tree structure of the ABC-LUFP Config Tool element containing the link	Data length	Data location
Sub-Network → TeSys U n°1 → Preset Multiple Registers → Query → Data	0x0002	0x0200
Sub-Network → TeSys U n°2 → Preset Multiple Registers → Query → Data	0x0002	0x0202
Sub-Network → TeSys U n°3 → Preset Multiple Registers → Query → Data	0x0002	0x0204
Sub-Network → TeSys U n°4 → Preset Multiple Registers → Query → Data	0x0002	0x0206
Sub-Network → TeSys U n°5 → Preset Multiple Registers → Query → Data	0x0002	0x0208
Sub-Network → TeSys U n°6 → Preset Multiple Registers → Query → Data	0x0002	0x020A
Sub-Network → TeSys U n°7 → Preset Multiple Registers → Query → Data	0x0002	0x020C
Sub-Network → TeSys U n°8 → Preset Multiple Registers → Query → Data	0x0002	0x020E
Sub-Network → TeSys U n°1 → Read Holding Registers → Response → Data	0x0002	0x0000
Sub-Network → TeSys U n°2 → Read Holding Registers → Response → Data	0x0002	0x0002
Sub-Network → TeSys U n°3 → Read Holding Registers → Response → Data	0x0002	0x0004
Sub-Network → TeSys U n°4 → Read Holding Registers → Response → Data	0x0002	0x0006
Sub-Network → TeSys U n°5 → Read Holding Registers → Response → Data	0x0002	0x0008
Sub-Network → TeSys U n°6 → Read Holding Registers → Response → Data	0x0002	0x000A
Sub-Network → TeSys U n°7 → Read Holding Registers → Response → Data	0x0002	0x000C
Sub-Network → TeSys U n°8 → Read Holding Registers → Response → Data	0x0002	0x000E

**NOTE:** The “Byte swap” field must *always* have the value “No swapping” if you wish to avoid inverting the LSB and MSB bytes of the data read or written using the Modbus commands.

• **Description of Services Assigned to Gateway I / O** .....Section 4.2.8

⇒ In the example provided, these services are identical to the ones in the standard configuration, for the following reasons:

- The *periodic communications (inputs / outputs)* are configured so that the gateway input / output configuration (see above) is identical to the one in the standard configuration.
- The *aperiodic communications* corresponding to the gateway’s PKW service must be active, whether the gateway is configured according to profile FED C32 (in PL7 PRO) or profile FED C32 P (in ABC-LUFP Config Tool).
- The *list of active slaves (LAS)* must also be active on the gateway.

• **Validating and Saving the FIPIO Network Configuration** .....Section 4.2.9

⇒ These operations have no equivalent under ABC-LUFP Config Tool, as ABC-LUFP Config Tool can configure only one gateway at a time. The operations you can perform on a configuration are: Open and Save.

⇒ Under PL7 PRO, by contrast, you must always validate FED C32 P profile configuration and setting.

• **Assigning Symbols to the Gateway’s FIPIO Objects** .....Section 4.2.10

⇒ The FIPIO objects remain the same, except for the 30 configuration parameters (%KW\p.2.c\0.0 to %KW\p.2.c\0.0.29) and the 30 adjustment parameters (%MW\p.2.c\0.0.20 to %KW\p.2.c\0.0.49), which disappear, as their function is now taken over by ABC-LUFP Config Tool.

• **Checking the Operational Status of the Gateway** .....Section 4.2.11

⇒ If you want to use the example “LUFPI\_FEDC32\_Example.cfg”, run ABC-LUFP Config Tool, open the configuration file (“Open...” command in the “File” menu), transfer it to the gateway (“Download configuration to ABC-LUFP” command in the “File” menu) and close ABC-LUFP Config Tool.

⇒ Once the configuration has been downloaded to the gateway, and the PL7 PRO application transferred to the FIPIO master PLC, use PL7 PRO, in **ONLINE** mode, to check that the basic module



corresponding to the gateway is operational on the FIPIO network.

## 6. Advanced ABC-LUFP Config Tool

---

⇒ A “Hardware configuration fault” will occur if you downloaded an ABC-LUFP Config Tool configuration based on FED C32 P rather than FED C32, which is not the case for the aforementioned example file.

• **Modifying the Gateway Parameters in ONLINE Mode** .....Section 4.2.12

⇒ This operation is now performed under ABC-LUFP Config Tool instead of under PL7 PRO. Downloading a new configuration to the gateway reinitializes the gateway, and in the process its FIPIO and Modbus communications.

• **Debugging and Using the Gateway Configuration** .....Section 4.2.13

⇒ The diagnostics, inputs and outputs of the gateway are accessible, in **ONLINE** mode, via the debug window called up by double-clicking the basic module **0 | FED C32** corresponding to the LUFP1 gateway.

• **Developing a FIPIO Application**.....Section 4.2.14

⇒ The PLC application example in Appendix D: Sample Use under PL7 PRO is fully compatible with the gateway when configured using profile FED C32. The only modification required is to change the gateway profile type (double-click the FIPIO connection point and select profile FED C32 instead of FED C32 P).

# 7. Using ABC-LUFP Config Tool

Each part of this chapter describes a separate step allowing users to personalize the gateway configuration according to their own particular needs, but **only when it is configured using standard profile FED C32 under PL7 PRO**. Each part introduces a basic operation, isolating it from the rest of the configuration and describing the operations to be carried out using ABC-LUFP Config Tool and how they affect the gateway's general behavior.

Some of the operations can also be performed **when the gateway is configured using standard profile FED C32 P**. This possibility will be flagged by the symbol **FED C32 P** in the corresponding description.

In each case, the first two steps are required, as they allow you to establish the dialogue between the gateway and the PC software that enables you to configure it, i.e. ABC-LUFP Config Tool.

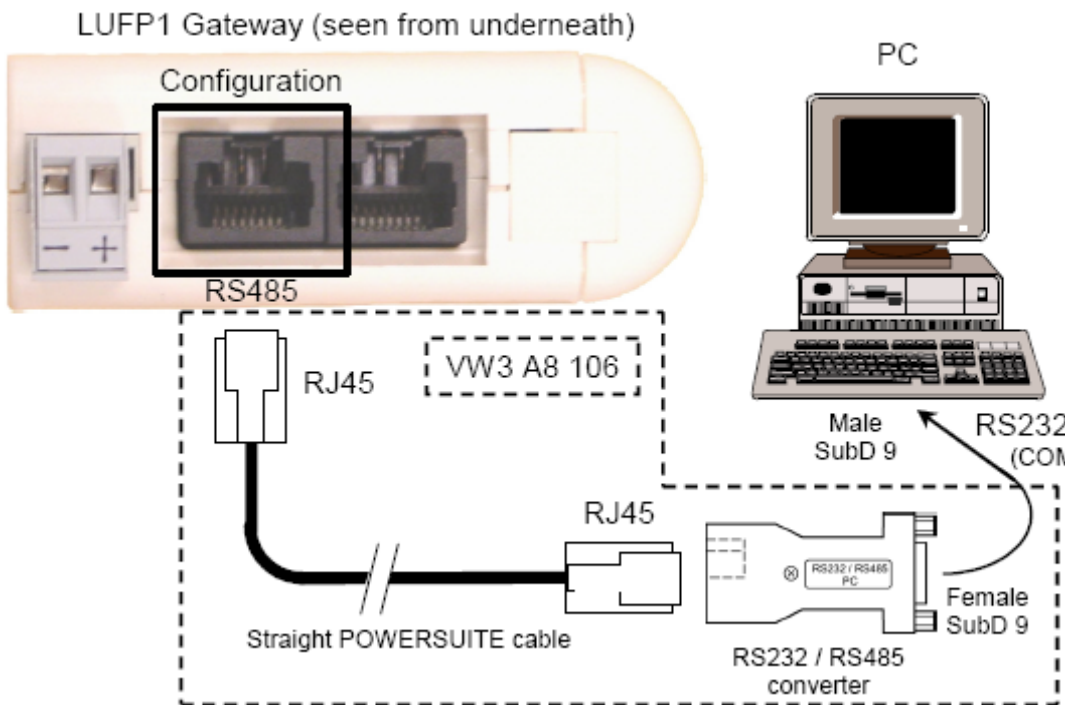
We strongly recommend that you read Chapter 4, and, above all, Chapter 6, as all the ABC-LUFP Config Tool operations described here assume that we are using the standard configuration (with 8 TeSys U motor starters) presented in the context of the two LUFP1 gateway implementation methods.

## 7.1. Connecting the Gateway to the Configuration PC

This step is required when setting up the gateway configuration application, ABC-LUFP Config Tool.

Connecting the gateway to one of the serial (COM) ports on a PC requires a straight PowerSuite cable and an RS232/RS485 converter. These two items are the same as those allowing dialogue with drives and soft start/soft stop units using the **PowerSuite** application and are both available from the catalogue (ref.: VW3 A8 106).

Be sure to use the "POWERSUITE" cable and the "RS232 / RS485 PC" converter. An "ATV28 before 09 / 2001" cable and an "ATV 58" converter are also supplied with these items, but they should not be used with the LUFP1 gateway.

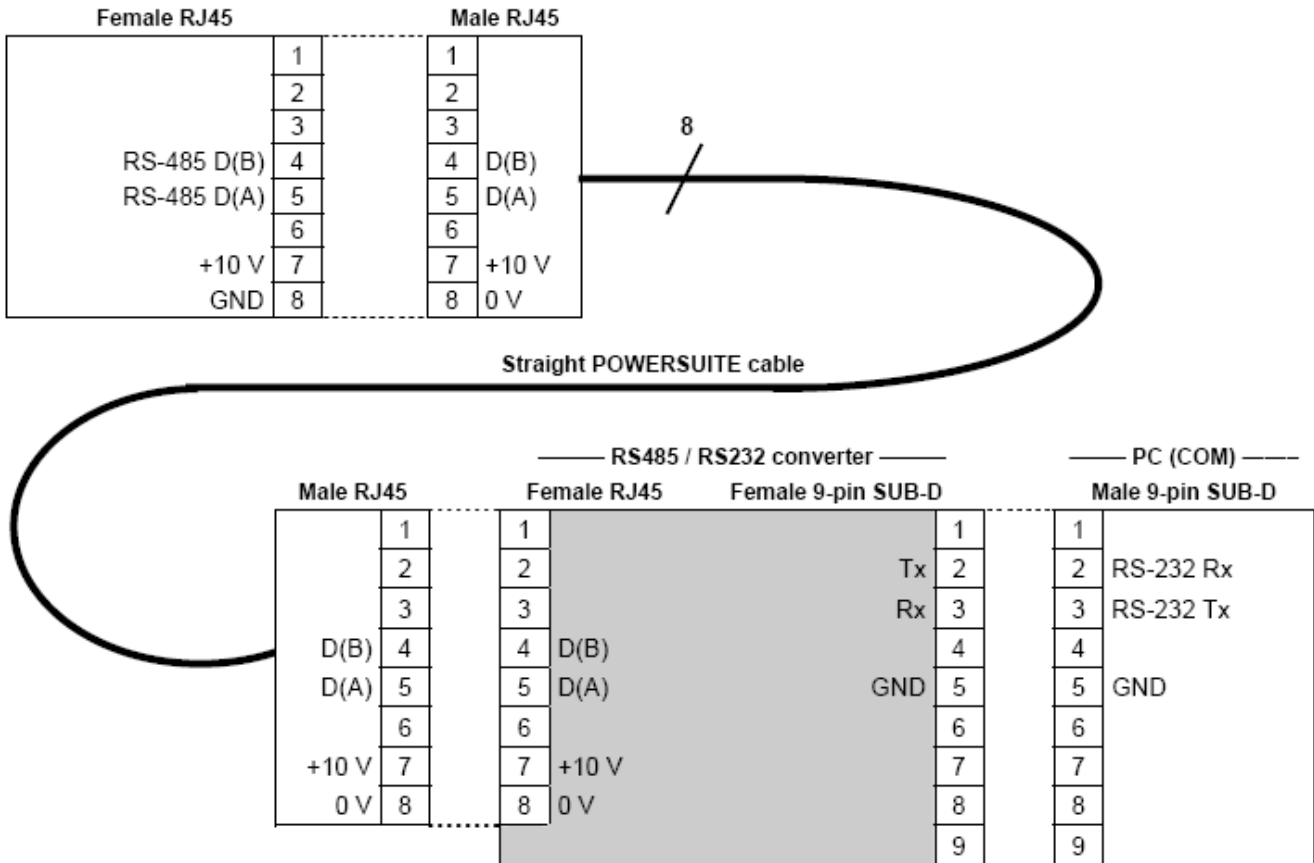


Once the gateway has been connected to a PC with the PowerSuite cable and the RS232/RS485 converter, you can change its configuration using "ABC-LUFP Config Tool". This configuration tool also allows you to carry out a few diagnostics on the gateway.

### 7.1.1. Pin-Outs

# 7. Using ABC-LUFP Config Tool

— LUF1 (Configuration) —



**NOTE:** The inversion of the Rx and Tx signals between the gateway and the PC is shown on the 9-pin Sub-D connectors, as beyond this junction, the RS-232 signals are replaced by the D(A) and D(B) polarizations of the RS-485 signals.

## 7.1.2. RS-232 Link Protocol

There is no need to configure the PC's COM port, as ABC-LUFP Config Tool uses a specific setup which replaces the one for the port being used. This replacement is temporary and is cancelled as soon as ABC-LUFP Config Tool stops using this serial port, i.e. when ABC-LUFP Config Tool is closed.

# 7. Using ABC-LUFP Config Tool

## 7.2. Installing ABC-LUFP Config Tool

The minimum system requirements for ABC-LUFP Config Tool are as follows:

- Processor.....Pentium 133 MHz
- Free hard disk space .....10 MB
- RAM..... 8 MB
- Operating system.....MS Windows 95 / 98 / ME / NT / 2000 / XP
- Browser.....MS Internet Explorer 4.01 SP1

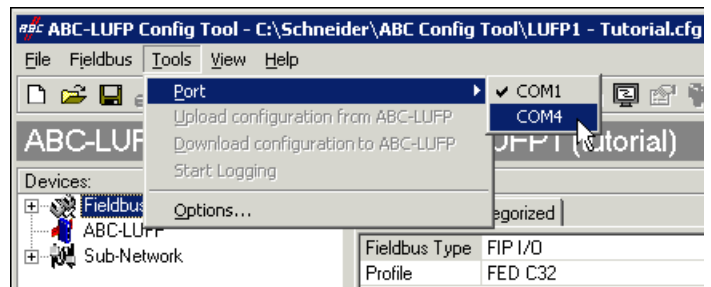
The ABC-LUFP Config Tool installation program can be found on the <http://www.schneider-electric.com> website (ref. VW3 A8 104). To install it, run “ABC-LUFP153.exe” and follow the on-screen instructions.

You can read about how to use ABC-LUFP Config Tool in a user manual entitled **AnyBus Communicator – User Manual** which is also on the <http://www.schneider-electric.com> website: “ABC\_User\_Manual.pdf”. We strongly recommend that you read this manual when using ABC-LUFP Config Tool: the present guide will limit itself to the ABC-LUFP Config Tool features relevant to a LUFP1 gateway implementation.

## 7.3. Connecting to / Disconnecting from the Gateway

In the ABC-LUFP Config Tool, the connection to the gateway must be performed manually.

But first, you should check which serial port the ABC-LUFP Config Tool will use for this connection. In the “Tools” menu, the “Port” sub-menu will reveal which serial ports (COM1, COM2, etc.) are currently available. If several COM ports are available, select the port you intend to use for connecting to and configuring the gateway.




An example is given here:


**NOTE:** If all the serial ports of your PC are already used by other applications, you must first close ABC-LUFP Config Tool, then “free” a serial port by disconnecting, closing, or terminating an application that currently uses a serial port.

Afterward, restart ABC-LUFP Config Tool because COM ports are only scanned during its start-up; the freed COM port should now be useable by it.

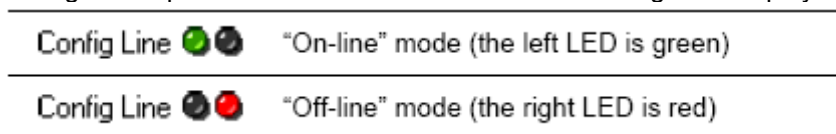
To connect ABC-LUFP Config Tool to the gateway:

- right-click on the “ABC-LUFP” element and click on “Connect” in the popup menu that appears, **or**
- select the “ABC-LUFP” element and choose “Connect” in the “ABC-LUFP” menu, **or**
- click on the  button.

Once connected, you can disconnect ABC-LUFP Config Tool from the gateway by:

- right-clicking on the “ABC-LUFP” element and clicking on “Disconnect” in the popup menu that appears, **or**
- selecting the “ABC-LUFP” element and choosing “Disconnect” in the “ABC-LUFP” menu, **or**
- clicking on the  button.

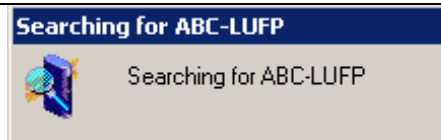
The rightmost part of the status bar of ABC-LUFP Config Tool displays its current connection mode:



In “On-line” mode, ABC-LUFP Config Tool periodically polls the gateway in order to detect if the gateway has been disconnected.

## 7. Using ABC-LUFP Config Tool

When an unwanted disconnection does occur, ABCLUFP Config Tool goes to “Off-line” mode (the red LED is displayed) and automatically retries to connect to the gateway. The “Searching for ABC-LUFP” window is visible for the duration of this search.





If the search fails, ABC-LUFP Config Tool prompts you with “No Module was found, retry?”.

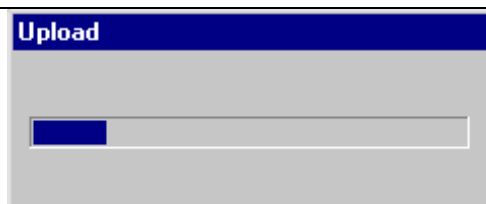
- Should you select the “Cancel” button, ABC-LUFP Config Tool remains in “Off-line” mode.
- Should you select the “Retry” button, ABC-LUFP Config Tool resumes the search for an ABC-LUFP gateway.

### 7.4. Importing the Gateway Configuration

Before you can make any changes to the gateway configuration, you will first need to import its current configuration. If you already have this configuration on your hard disk, all you will need to do is open the file corresponding to this configuration.

Check that the gateway has a valid configuration and that it is working properly, i.e. that LED  GATEWAY is flashing green.

In ABC-LUFP Config Tool, choose “Upload configuration from ABC-LUFP” from the “File” menu or click the  button in the ABC-LUFP Config Tool toolbar. The “Upload” window opens, containing a progress bar that tracks the advance of the gateway configuration uploading process. The window closes once the upload is completed.



This step is particularly important if you want to read details about the *current* contents of the gateway configuration rather than about a configuration in a “.cfg” file on a PC. You can then use this configuration as a template for any changes you wish to make subsequently, thus avoiding having to create one from scratch and reducing the potential risk of error.

#### NOTE:

- Save this configuration to your hard disk so that it is always available. This will allow you to reconfigure the gateway “cleanly” should the configuration become invalid.


**FED C32 P**

You can use the “Upload” command to check the configuration and settings that you made under PL7 PRO. This is because the gateway creates an equivalent configuration that can be viewed using the retrieve command.

# 7. Using ABC-LUFP Config Tool

## 7.5. Transferring a Configuration to the Gateway

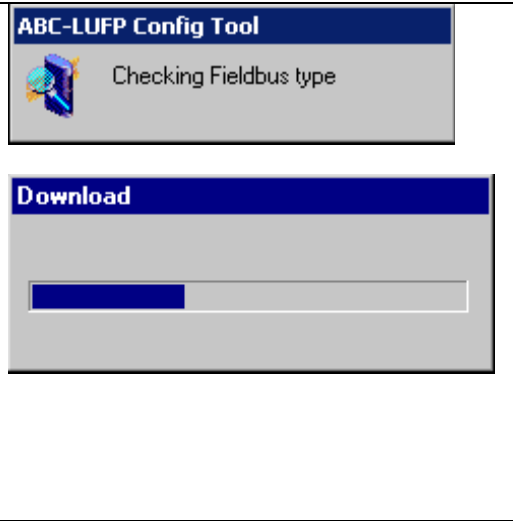
When using ABC-LUFP Config Tool, you can transfer the configuration you are editing to the gateway at any time.

Choose “Download configuration to ABC-LUFP” from the “File” menu or click the  button in the ABC-LUFP Config Tool toolbar. ABC-LUFP Config Tool initializes a check test of the gateway type.

**NOTE:** During this very quick test, the PC should not carry out any other operations, as this could lead to ABC-LUFP Config Tool apparently freezing up and slowing down the PC’s general operation for several minutes. After the test is complete, the PC will return to full speed, and may be used normally.

Once this test has finished, a window called “Download” opens and a progress bar shows the state of progress for the transfer of the configuration to the gateway.

**NOTE:** Do not interrupt this operation; otherwise you will have to start it again from the beginning.



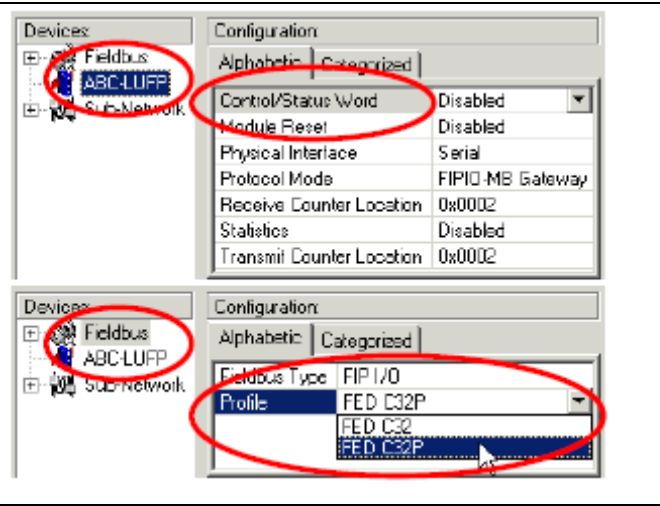
Check that the transfer has been correctly carried out: LED **6** GATEWAY should be flashing green. If this LED is flashing red/green, save the configuration you were editing, open a file containing a valid configuration (“LUFP1\_FEDC32\_Example.cfg”, for example), and transfer it to the gateway. This will restore it to a known initial state. You can then continue with the configuration you were transferring, and make any corrections which may be necessary.

### 7.5.1. Transferring the Internal Configuration (Profile FED C32 P)

**FED C32 P** When using the standard profile FED C32 P to set up the gateway under PL7 PRO, *never* use the command “Download configuration to ABC-LUFP” to transfer a configuration to the gateway.

The only exception to this rule involves the “Hardware configuration fault” (see Section 4.2.11).

- In this case, you should perform the following actions:
- Use the “New” command of the “File” menu;
  - Set the following properties for the “ABC-LUFP” element:
    - o Control/Status Word ..... Disabled
    - o Module Reset ..... Disabled
    - o Protocol Mode ..... FIPIO-MB Gateway
    - o Statistics ..... Disabled
  - Set the following properties for the “Fieldbus” element:
    - o Fieldbus Type ..... FIP I/O
    - o Profile ..... FED C32P
  - Do not add any node nor Modbus command to the “Sub-Network” element;
  - Download this “blank” configuration into the gateway.



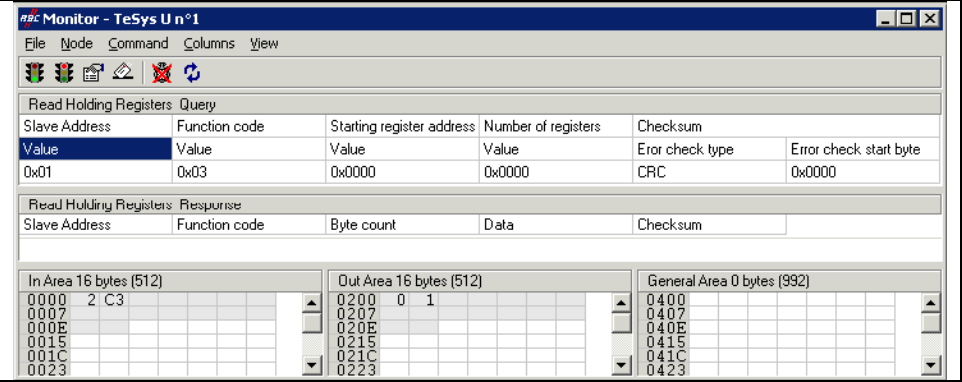
## 7.6. Monitoring the Contents of the Gateway’s Memory

One of the main commands that you will need to use when setting up the gateway is the command allowing you to read the contents of the gateway’s memory and to display it in a window designed for this purpose. This will be particularly useful when you are working on your PLC applications and configurations. However, it only shows data from the “Data” and “Preset Data” fields configured in the “Query” and “Response” elements of a single Modbus slave.

# 7. Using ABC-LUFP Config Tool

As the gateway's LAS and PKW services are not located in the memory zones reserved for exchanges with the Modbus slaves, the input and output data of these services cannot be viewed in the same way. Only PL7 PRO and the FIPIO master can access such data.

To monitor the contents of the gateway's memory, start by selecting the node corresponding to the Modbus slave whose data you wish to view, and then choose "Monitor" from the menu with the same name as the previously selected node. A monitoring window then appears.



The sample window above displays the contents of the memory exchanged, under the standard configuration, with TeSys U motor starter n°1.







In the upper pane you can choose a Modbus command, edit its contents, and send it to the Modbus network ("Command" menu). The response will then be displayed in the same pane. See Section 2.10 in the ABC-LUFP Config Tool user manual, entitled **AnyBus Communicator – User Manual**, for further information about how to use this window. The manual can be found on the <http://www.schneider-electric.com> website under "ABC\_User\_Manual.pdf".

The lower pane allows you to view the contents of the gateway's memory, but only the bytes used in the query and response frames of the commands configured for the selected node.

In the window shown above, the data displayed correspond to the values at the memory locations designated by the "Data" fields in the commands configured for the node "TeSys U n°1", namely: "Read Holding Registers" and "Preset Multiple Registers".

**NOTE:** The data exchanged with the previously selected Modbus slave are displayed MSB-first, i.e. in the order MSB /LSB (from left to right, in ascending order of memory address), provided that the "Byte Swap" option in the "Data" or "Preset Data" element of the corresponding Modbus command was set to "No swapping" (see Section 7.12.2.4).

A brief description of the toolbar buttons in this window is given below:

-   Stop / Start communications with the selected node (see "Node" menu, below)
-   Select / Send the Modbus command in the upper pane (see "Command" menu, below)
-   Stop / Resume refreshing the data in the lower pane



## 7. Using ABC-LUFP Config Tool

---

The menus of this window allow you to perform the following actions:

- “File” menu:
  - The “Exit” command closes the “Monitor” window, thus returning to ABC-LUFP Config Tool.
- “Node” menu:
  - The “Start Node” command restarts all the communications configured for the node currently monitored. Since a node is active by default, this command is only useful if the node has been explicitly stopped using the “Stop Node” command.
  - The “Stop Node” command stops all the communications configured for the node currently monitored. This means that all Commands configured for the node are inhibited.  
**Note:** The Stop / Start Node commands can be particularly useful when used to isolate one or more nodes in order to investigate Modbus communication problems.
- “Command” menu:
  - The “Select Command” command opens a “Select Command” window that enables you to select a Modbus command (see chapter 7.11.2). Once selected, the Query and Response frames of this command will be displayed in the upper part of the “Monitor” window. You can then edit the value associated with each field of the Query frame before sending the command using “Send Command” (see below).
  - The “Send Command” command triggers the emission of the Query displayed in the upper part of the “Monitor” window. As soon as a Modbus Response is received by the gateway, ABC-LUFP Config Tool displays its contents in the upper part of the “Monitor” window.
- “Columns” menu:
  - The “Free” choice configures the three monitoring columns (“In Area”, “Out Area”, and “General Area”) to automatically adjust their width on a 1-byte unit (1 byte, 2 bytes, 3 bytes, etc.) each time you modify the width of the “Monitor” window.
  - The “8 Multiple” choice configures the three monitoring columns to automatically adjust their width on an 8-byte unit (8 or 16 bytes) each time you modify the width of the “Monitor” window.
- “View” menu:
  - The “Hex” choice configures the three monitoring columns to display all monitored values and memory addresses in Hexadecimal.
  - The “Decimal” choice configures the three monitoring columns to display all monitored values and memory addresses in Decimal.

### FED C32 P

Proceed with Importing the Gateway Configuration (Section 7.4) before Monitoring the Contents of the Gateway’s Memory. This will ensure that the data displayed corresponds to the inputs and outputs exchanged with the selected Modbus slave.

### 7.7. Creating a New Configuration

When you want to create a new configuration, you can either use an existing configuration as a starting-point—as is the case with most of the operations described in Chapter 7—or you can use a blank configuration.

For a blank configuration, use the “New” command in the “File” menu:

- Set the “Protocol Mode” property to “FIPIO-MB Gateway” in the “ABC-LUFP” element;
- Set the “Fieldbus Type” and “Profile” properties to “FIP I/O” and “FED C32”, respectively, in the “Fieldbus” element.

### FED C32 P

The creation of new configurations is performed *entirely* under PL7 PRO. Only transfer a new configuration to the gateway if it signals a “Hardware configuration fault” (see Section 7.5.1).

### 7.8. Deleting a Modbus Slave

This step allows you, for instance, to free up a location on the downstream Modbus network, known as the “Sub-Network” in ABC-LUFP Config Tool, in order to replace one Modbus slave with another.

In fact the gateway’s standard configuration already allows it to communicate with eight TeSys U motor starters, and the maximum number of Modbus slaves with which it is permitted to communicate is limited to eight.

## 7. Using ABC-LUFP Config Tool

---

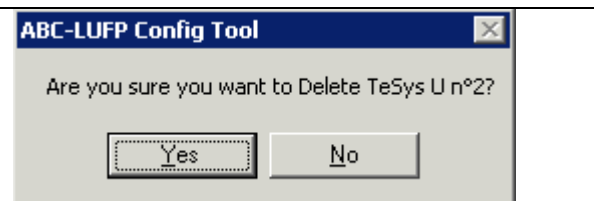
If the gateway is used to manage exchanges on a Modbus network with fewer than eight TeSys U motor starters, it is preferable to delete the redundant TeSys U motor starters from the gateway configuration: the deterioration in performance caused by the absence of one or more TeSys U motor-starters is such that it is preferable to carry out this operation using ABC-LUFP Config Tool.

### Procedure for deleting a Modbus slave

- 1) Select the node corresponding to the Modbus slave you wish to delete from the configuration. If this is the only node remaining in the configuration, you will not be able to delete it, as the downstream Modbus network must include at least one slave.
- 2) Right click on the icon or the name of this Modbus slave. A menu pops up underneath the mouse cursor.  
*or*  
In the ABC-LUFP Config Tool main menu, pull down the menu with the same name as the previously selected node.
- 3) In this menu, click "Delete". The following confirmation window then appears, asking you to either confirm that you want to delete the selected node ("TeSys U n°2" in this example) or cancel the operation.

4) If you confirm that you want to delete the node, the menu disappears, along with the previously selected node. Otherwise, the node will still be there when the window closes.

**Keyboard shortcut:** "Del" key.



# 7. Using ABC-LUFP Config Tool

---

## 7.9. Adding a Modbus Slave

This feature allows you to add a Modbus slave of a different type to the other Modbus slaves in the configuration. On the other hand, if the slave is of the same type as one of the previously configured slaves, it is preferable to copy the old slave rather than create a new one.

An additional import/export feature also allows you to individually save the complete configuration of a Modbus slave, in order to have access to it in ABC-LUFP Config Tool, from any configuration and at any time.

These two features are only available provided that there are fewer than 8 Modbus slaves declared, which is not the case in the standard configuration, as it comprises 8 TeSys U motor starters.

### Adding a new type of Modbus slave

Use one of the two methods described below:

- a) Select "Sub-Network", then choose "Add Node" from the "Sub-Network" menu. A new node is added after all the other configured nodes. By default, its name is "New Node".
- b) Select one of the nodes located under the "Sub-Network" element, then choose "Insert New Node" from the menu with the same name as the selected node. A new node is added just before the selected node. By default, its name is "New Node".

All of the steps in configuring the new node are described in Section 7.11.

### Copying a previously configured Modbus slave

Select the node corresponding to the slave whose configuration you want to copy, then choose "Copy" from the menu with the same name as the selected node. **Keyboard shortcut:** "Ctrl C".

Then use one of the two methods described below:

- a) Select "Sub-Network", then choose "Paste" from the "Sub-Network" menu. A new node is added after all the other configured nodes. Its name and its whole configuration are identical to that of the node you copied. **Keyboard shortcut:** "Ctrl V".
- b) Select one of the "Sub-Network" nodes, then choose "Insert" from the menu with the same name as the selected node. A new node is added just before the one which is selected. Its name and its whole configuration are identical to that of the node you copied.

As the new node and the original node are identical in every way, you will need to change (1) the name of the node, (2) the address of the corresponding Modbus slave and (3) the location of the data exchanged between the gateway memory and the Modbus slave. All these operations are described in Section 7.11 and Section 7.12.

## **WARNING**

### **DUPLICATE MODBUS ADDRESSES OR GATEWAY MEMORY RANGES**

If you choose to add a Modbus slave by copying the configuration of an existing Modbus slave, you must change the added device's Modbus address and the memory locations it uses to exchange data with the gateway.

Duplicated Modbus addresses or gateway memory locations may result in communications errors, incorrect information being written to a slave's registers, or in writing the registers of an unintended device. Any of these errors may result in unintended equipment operation.

**Failure to follow this instruction can result in death, serious injury, or equipment damage.**

### Importing/exporting a Modbus slave configuration

With ABC-LUFP Config Tool you can independently save and load a node configuration on the downstream "Sub-Network". This allows you, for example, to build up a library of Modbus slave templates, so that you can use them in any configuration.

To save the configuration of a Modbus slave, select the node it corresponds to, then choose "Save Node" from the menu with the same name as the selected node. A dialog box will then appear asking you to save the configuration (export in XML format).

## 7. Using ABC-LUFP Config Tool

---

To insert a node using the XML file containing a Modbus slave configuration as a template, use one of the two methods described below:

- a) Select “Sub-Network”, then choose “Load Node” from the “Sub-Network” menu. A dialog box asks you to choose a file containing a Modbus slave configuration (import in XML format). A new node is added after all the other configured nodes. Its name and its whole configuration are identical to those of the Modbus slave as configured when last saved.
- b) Select one of the “Sub-Network” nodes, then choose “Insert from File” from the menu with the same name as the selected node. A new node is added just before the selected node. Its name and its whole configuration are identical to those of the Modbus slave as configured when last saved.

You will then need to change (1) the name of the node, (2) the address of the corresponding Modbus slave and (3) the location of the data exchanged between the gateway memory and the Modbus slave. All these operations are described in Section 7.11 and Section 7.12.

### **WARNING**

#### **DUPLICATE MODBUS ADDRESSES OR GATEWAY MEMORY RANGES**

If you choose to add a Modbus slave by copying the configuration of an existing Modbus slave, you must change the added device’s Modbus address and the memory locations it uses to exchange data with the gateway.

Duplicated Modbus addresses or gateway memory locations may result in communications errors, incorrect information being written to a slave’s registers, or in writing the registers of an unintended device. Any of these errors may result in unintended equipment operation.

**Failure to follow this instruction can result in death, serious injury, or equipment damage.**

# 7. Using ABC-LUFP Config Tool

## 7.10. Modifying Periodic Data Exchanged with Modbus Slaves

This involves replacing, adding or deleting periodic data exchanged with one of the Modbus slaves. For each of these operations, we shall take the standard configuration as described in Chapter 6) as the starting-point: any changes previously made will have been cancelled at the start of each operation. Each operation is illustrated below by a specific example.

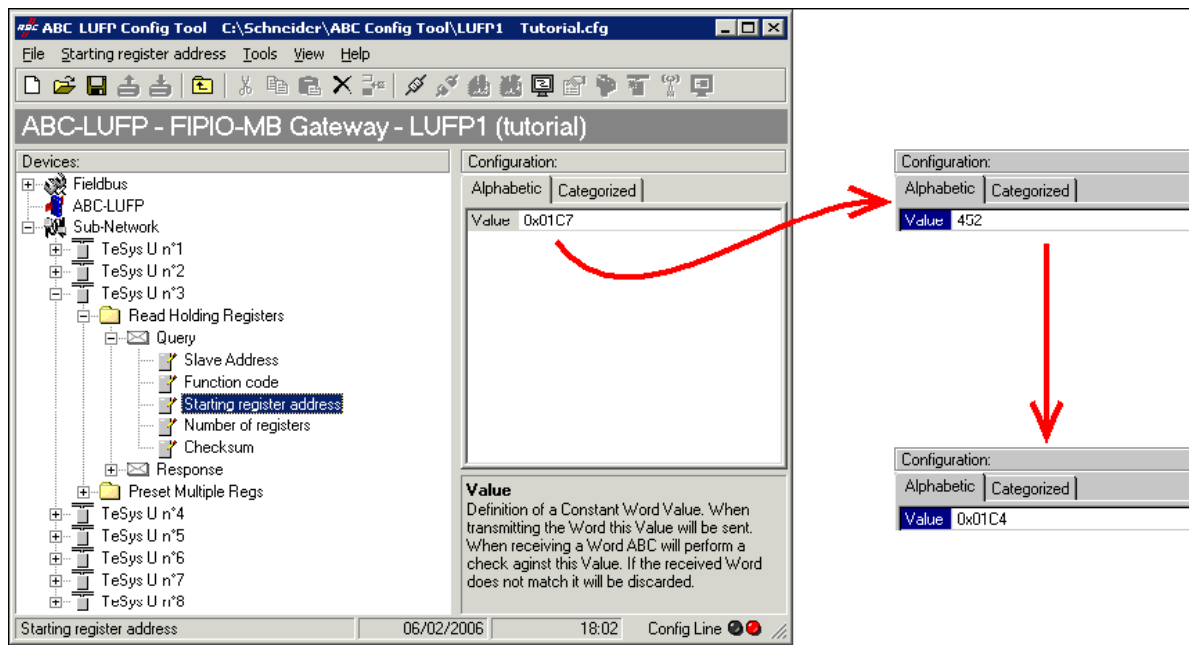
Always remember to save the changes you have made, or to transfer the whole configuration to the gateway. This will allow you to check that the configuration is valid, as the gateway automatically verifies the configuration when it is downloaded.

### 7.10.1. Replacing a Periodic Input

E.g. TeSys U Motor starter n°3: we want to replace the “TeSys U Status Register” control (address 455 = 0x01C7) with the “1st Fault Register” control (address 452 = 0x01C4).

The operation is a very simple one and consists merely of changing the value of the “Starting Register Address” element in the “Query” of the “Read Holding Registers” command (Modbus command for reading values of multiple registers).

Select this element, then change its value as shown at the top of the next page. You can enter the address of the parameter in decimal format: ABC-LUFP Config Tool will automatically convert it to hexadecimal.



This operation in no way alters the contents of the gateway memory, as we do not need to change the values of the “Data length” and “Data location” fields in the “Data” element of the “Response” to the aforementioned command; so no additional operations are required in ABC-LUFP Config Tool.

However, the FIPIO master PLC application will have to take account of the change in the nature of the corresponding input. In Input Data Memory, Appendix C, the description of the word located at address 0x0004 becomes “Value of the 1st fault register of motor starter □”. This word corresponds to PLC input word %IWp.2.c0.0.2 (see Section 4.2.7, Section 5.1.1, and Section 5.1.3).

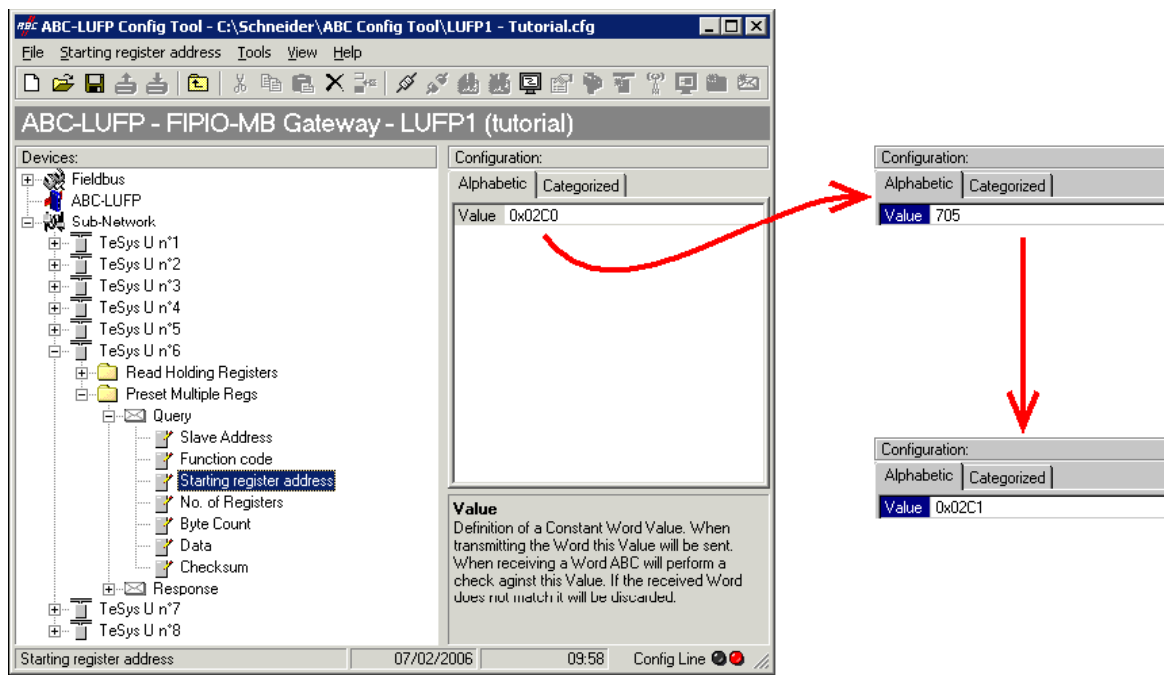
# 7. Using ABC-LUFP Config Tool

## 7.10.2. Replacing a Periodic Output

E.g. TeSys U Motor starter n°6: we want to replace the “Command Register” command (address 704 = 0x02C0) with the “2nd Command Register” command (address 705 = 0x02C1).

This operation consists in changing the value of the “Starting Register Address” element in the “Query” and the value of “Starting Address” in the “Response” to the “Preset Multiple Registers” command (Modbus command for writing values to multiple registers).

Select “Starting Register Address” in the “Query” and modify its value as shown at the top of the next page. You can enter the address of the parameter in decimal format: ABC-LUFP Config Tool will automatically convert it to hexadecimal. **Do the same for the “Starting Address” element of the “Response”**, because the gateway checks the value of this field when it receives each Modbus response. If the value does not correspond to that of the query, the gateway will ignore the response.



This operation in no way alters the contents of the gateway memory, as we do not need to change the values of the “Data length” and “Data location” fields in the “Data” element of the “Query”. No additional operations are required in ABC-LUFP Config Tool.

However, the FIPIO master PLC application will have to take account of the change in the nature of the corresponding output. In Output Data Memory, Appendix C, the description of the word located at address 0x020A becomes “Value of the 2nd command register of motor starter □”. This word corresponds to the PLC output word %QWp.2.c0.0.5 (see Section 4.2.7, Section 5.1.1, and Section 5.1.3).

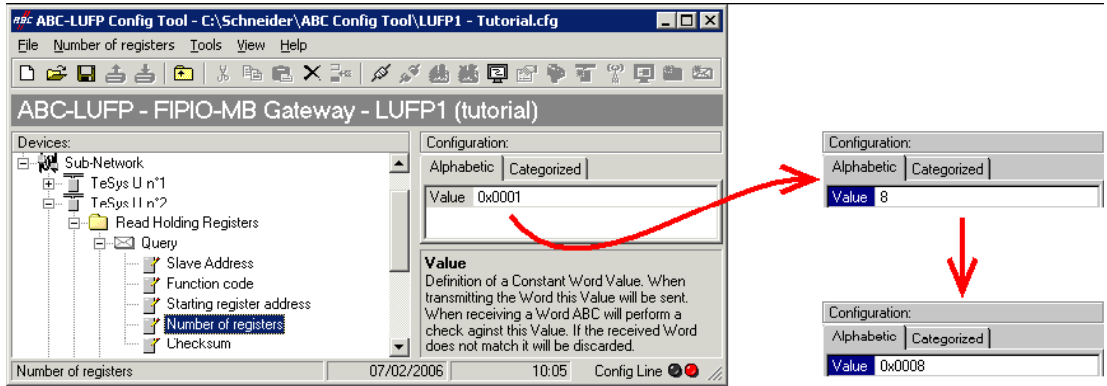
## 7.10.3. Increasing the Number of Periodic Inputs

E.g. TeSys U motor starter n°2: we want to extend the controls on this motor starter, starting with the currently controlled register—“TeSys U Status Register” (address 455 = 0x01C7)—and expanding as far as the “Reserved: 2<sup>nd</sup> Warning Register” (address 462 = 0x01CE). This will bring the number of registers controlled from 1 up to 8.

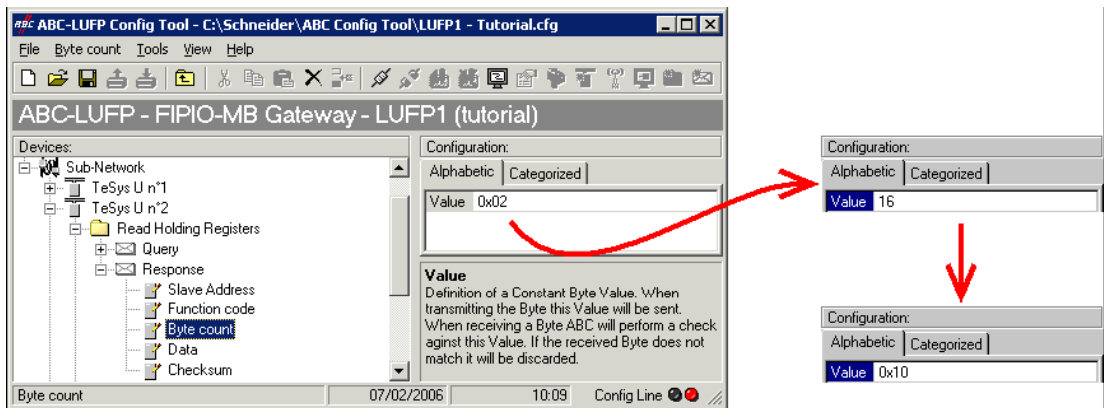
The necessary operations are described below:

# 7. Using ABC-LUFP Config Tool

- 1) Change the number of registers controlled: This step consists in modifying the value of the element “Number of Registers” in the “Query” of the “Read Holding Registers” command (Modbus command for reading the values of multiple registers). Select this element, then change its value as shown at the top of the next page. ABC-LUFP Config Tool will automatically convert any value entered in decimal to hexadecimal.



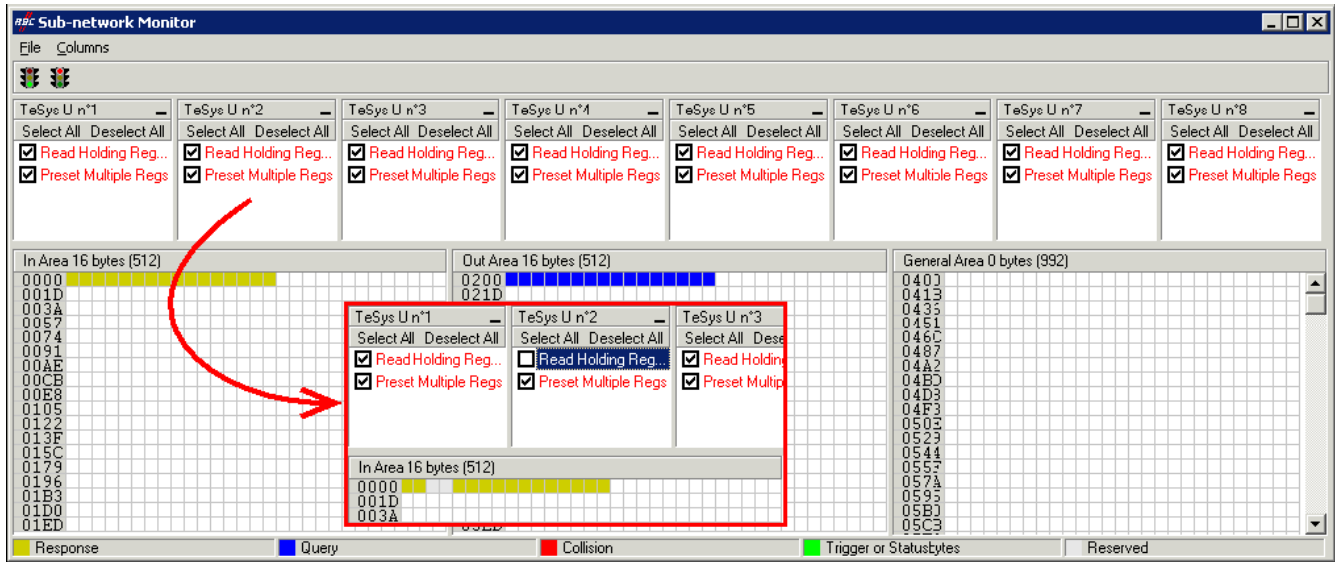
- 2) Change the number of data bytes in the Modbus response: The number of bytes read from the memory of TeSys U motor starter n°2 increases from 2 to 16, as the number of registers controlled has increased from 1 to 8. Select the “Byte count” element in the “Response” and change its value as shown below. ABC-LUFP Config Tool will automatically convert any value entered in decimal to hexadecimal.



- 3) Change the location of the Modbus data received in the gateway memory: As the number of bytes read (see previous step) has increased from 2 to 16, the Modbus data received must be placed at a different location in the gateway memory.

If you are unsure how much of the gateway’s memory is currently in use, select “Sub-Network” and choose “Monitor” from the “Sub-Network” menu. The following window appears, allowing you to see how much of the gateway’s memory is occupied.

# 7. Using ABC-LUFP Config Tool



To see which memory locations are occupied by data from the command we are interested in, simply uncheck the box corresponding to the “Read Holding Registers” command in the “TeSys U n°2” node, as shown above. This reveals that the Modbus data received in response to this command occupy 2 bytes, starting at 0x0002.

**NOTE:** Only memory locations 0x0000 to 0x0033 can be used (max.size of input data = 52 bytes).

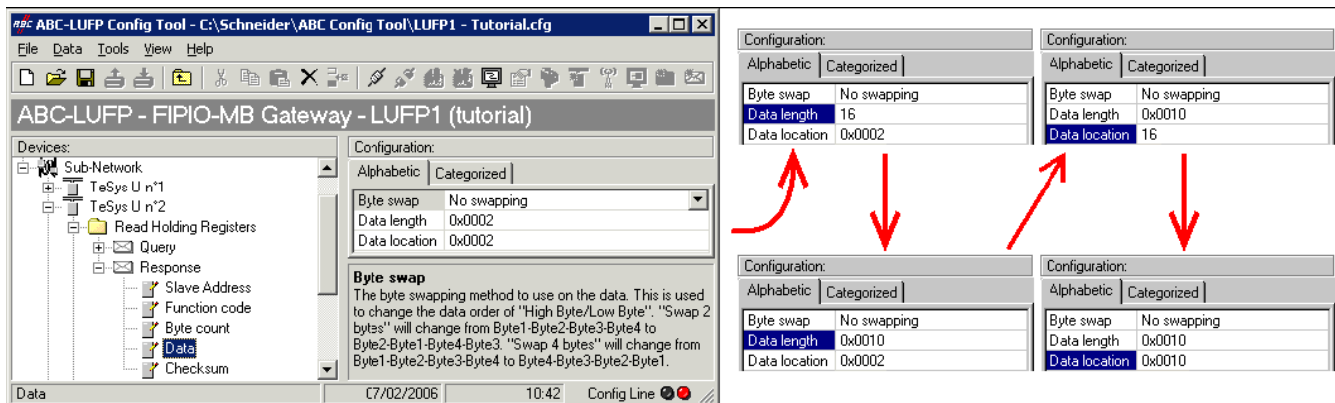
Using any other address in the gateway’s input data memory area (0x0034 to 0x01FF) will throw the gateway into **non-operational** mode, signaled by the sudden red-green flashing of LED 6 GATEWAY. LEDs 1, 2, 3 and 4 will also flash, at a frequency of 2 Hz.

You can ignore the size indications on the graphics in this window (“In Area 16 bytes” and “Out Area 16 bytes”), as the gateway has a fixed size for inputs and outputs alike.

If you want to memories the 16 bytes of Modbus data which will be received by the gateway for this command, once the changes have been made, you will have to either shift all the other input data by 14 bytes—a painstaking operation—or else change the memory location of the block of data received. In the example described here, we will be using the second solution, although the first solution is actually preferable, in principle, as it avoids leaving any “holes” in the gateway’s memory. This has no impact on the FIPIO exchanges; however, as the size of transfers over the gateway is set at 32 input words and 32 output words by the use of the standard profile FED C32 (P).

We will be placing the 16 bytes of data starting at address 0x0010 (16 in decimal), i.e. directly after the input data from the standard configuration. The 2 bytes at 0x0002 and 0x0003 become “free memory locations”.

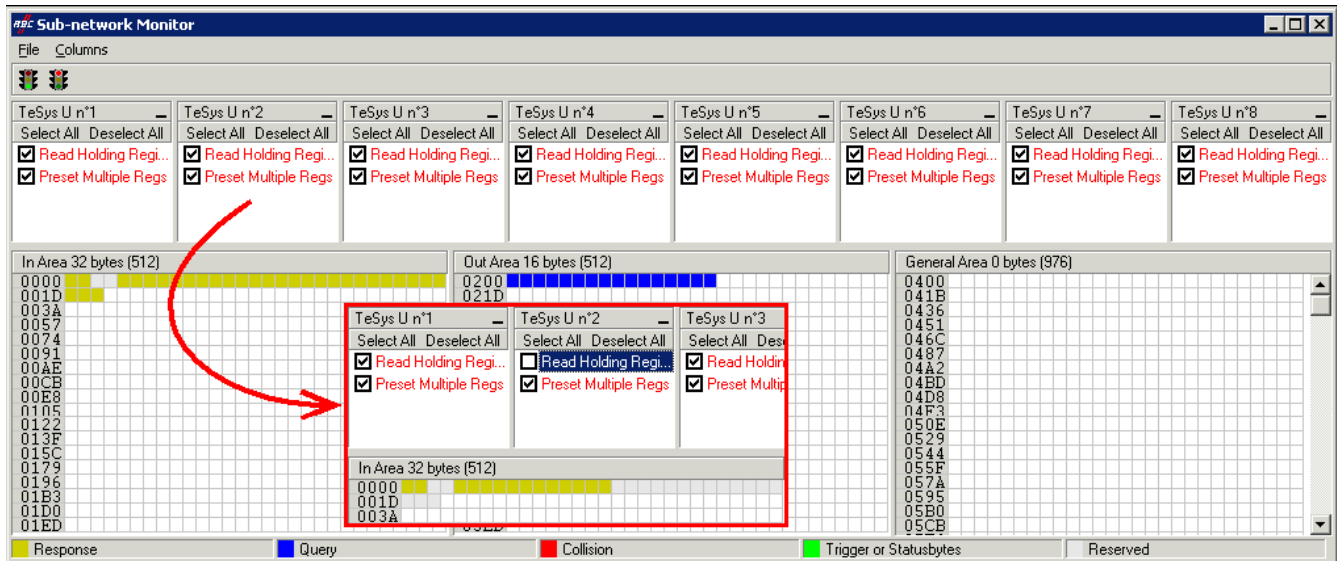
Close the “Sub-network Monitor” window, then, once you are back in the main ABC-LUFP Config Tool window, select the “Data length” and “Data location” fields respectively in the “Data” element of the “Response” and change their values as shown at the top of the next page. ABC-LUFP Config Tool will automatically convert any value entered in decimal to hexadecimal.





# 7. Using ABC-LUFP Config Tool

To check that these changes have been integrated into the configuration, choose the “Monitor” command again in the “Sub-Network” menu:



- 4) Download this configuration to the gateway: See Section 7.5. Check that the configuration is valid (LED  GATEWAY flashing green).
- 5) Save the configuration to your PC's hard disk.
- 6) Use the FIPIO master PLC inputs: It only remains to assign symbols and use the inputs corresponding to the new Modbus data read on motor starter n°2.

This gives us the mapping shown on the next page, derived from the one used for the standard gateway configuration. Changes in relation to the standard configuration are flagged by a grey background (except for the reserved word %IW/p.2.c\0.0.26).

## 7. Using ABC-LUFP Config Tool

Service	PLC input	Description	
		Bit 15.....Bit 8	Bit 7.....Bit 0
Periodic communications — Control of TeSys U motor starters	%IWp.2.c\0.0	Value of status register for motor starter ①	
	%IWp.2.c\0.0.1	Free memory location	
	%IWp.2.c\0.0.2	Value of status register for motor starter ②	
	%IWp.2.c\0.0.3	Value of status register for motor starter ③	
	%IWp.2.c\0.0.4	Value of status register for motor starter ④	
	%IWp.2.c\0.0.5	Value of status register for motor starter ⑤	
	%IWp.2.c\0.0.6	Value of status register for motor starter ⑥	
Periodic communications — Control of TeSys U motor starter ⑦	%IWp.2.c\0.0.7	Value of status register for motor starter ⑦	
	%IWp.2.c\0.0.8	Value of “TeSys U Status Register”	
	%IWp.2.c\0.0.9	Value of “Complementary Status Register”	
	%IWp.2.c\0.0.10	Value of “K7 Status Register”	
	%IWp.2.c\0.0.11	Value of “K7 Status Register 2 (free format)”	
	%IWp.2.c\0.0.12	Value of “K7 Status Register 3 (free format)”	
	%IWp.2.c\0.0.13	Value of “Warning Number”	
%IWp.2.c\0.0.14	Value of “Warning Register”		
%IWp.2.c\0.0.15	Value of “Reserved: 2nd Warning Register”		
Periodic communications — Free locations	%IWp.2.c\0.0.16 ...to...	Not used (10 words)	
	%IWp.2.c\0.0.25		
	%IWp.2.c\0.0.26	Reserved (1 word)	
List of active slaves (LAS) service	%IWp.2.c\0.0.27	List of active Modbus slaves	
Aperiodic communications — Indexed periodic variables (PKW) service – RESPONSE	%IWp.2.c\0.0.28	PKE – Address of the datum written / read	
	%IWp.2.c\0.0.29	DN – Address of the responding slave	R/W/N – Read / Write OK / Error
	%IWp.2.c\0.0.30	PWE – Value written / read (1st word)	
	%IWp.2.c\0.0.31	PWE – Value written / read (2nd word)	

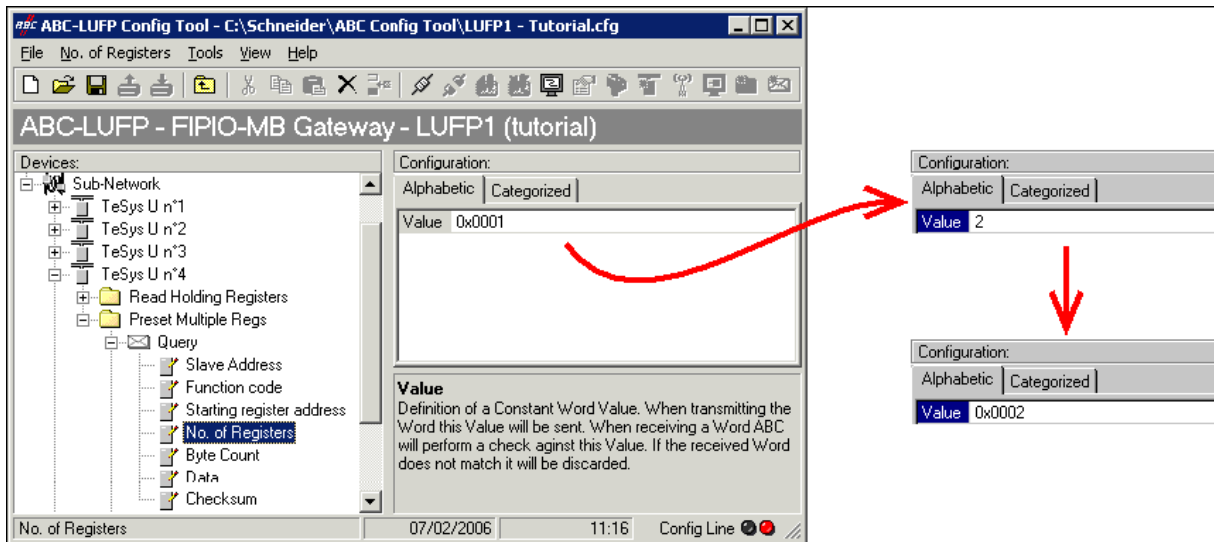
### 7.10.4. Increasing the Number of Periodic Outputs

E.g. TeSys U motor starter n°4: we want to extend the commands on this motor starter, retaining the currently commanded register, “Command Register” (address 704 = 0x02C0), and adding the next register, “2nd Command Register” (address 705 = 0x02C1). This will bring the number of registers commanded from 1 up to 2.

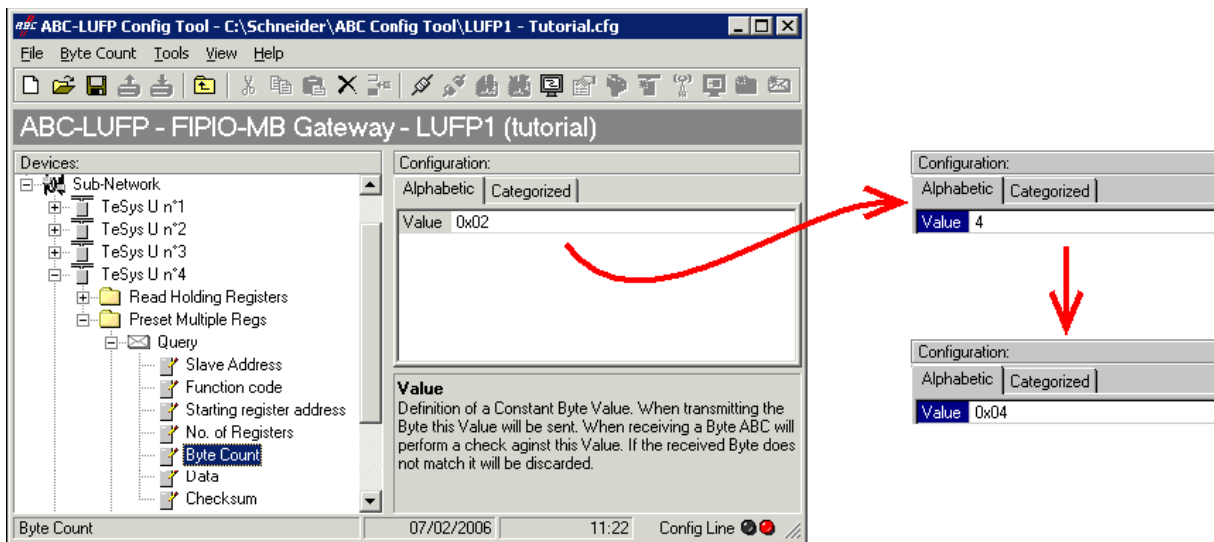
The necessary operations are described below:

- 1) Change the number of registers commanded: This step consists in modifying the value of the element “No. of Registers” in both the “Query” and “Response” of the “Preset Multiple Registers” command (Modbus command for writing values to multiple registers). Start by selecting the element “Starting Register Address” in the “Query”: change its value as shown at the top of the next page. ABC-LUFP Config Tool will automatically convert any value entered in decimal to hexadecimal. **Do the same for the “Starting Address” element of the “Response”**, because the gateway checks the value of this field when it receives each Modbus response. If the value does not correspond to that of the query, the gateway will ignore the response.

## 7. Using ABC-LUFP Config Tool



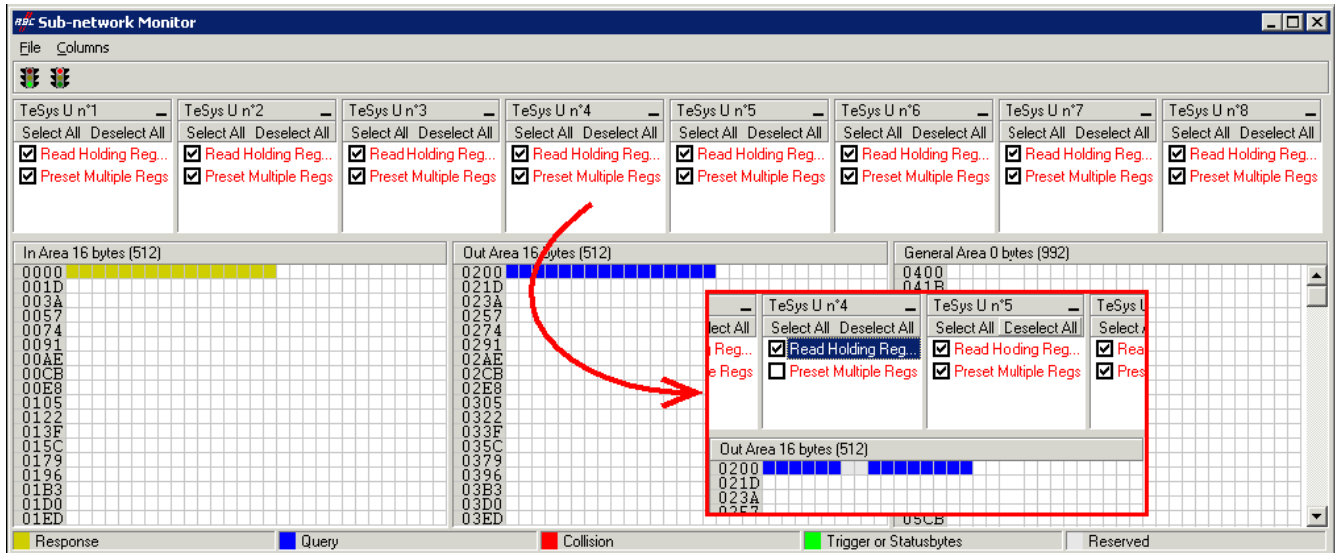
- 2) Change the number of data bytes in the Modbus query: The number of bytes written to the memory of TeSys U motor starter n°4 increases from 2 to 4, as the number of registers commanded has increased from 1 to 2. Select the "Byte count" element in the "Query" and change its value as shown below. ABC-LUFP Config Tool will automatically convert any value entered in decimal to hexadecimal.



- 3) Change the location of the Modbus data transmitted to the gateway memory: As the number of bytes transmitted (see previous step) has increased from 2 to 4, the Modbus data to be sent to TeSys U motor starter n°4 must be placed at a different location in the gateway memory.

If you are unsure how much of the gateway's memory is currently in use, select "Sub-Network" and choose "Monitor" from the "Sub-Network" menu. The following window appears, allowing you to see how much of the gateway's memory is occupied.

# 7. Using ABC-LUFP Config Tool



To see which memory locations are occupied by data from the command we are interested in, simply uncheck the box corresponding to the “Preset Multiple Registers” command in the “TeSys U n°4” node, as shown above. This reveals that the Modbus data transmitted with the query for the command occupy 2 bytes, starting at 0x0206.

**NOTE:** Only memory locations 0x0200 to 0x0233 can be used (max. size of input data = 52 bytes).

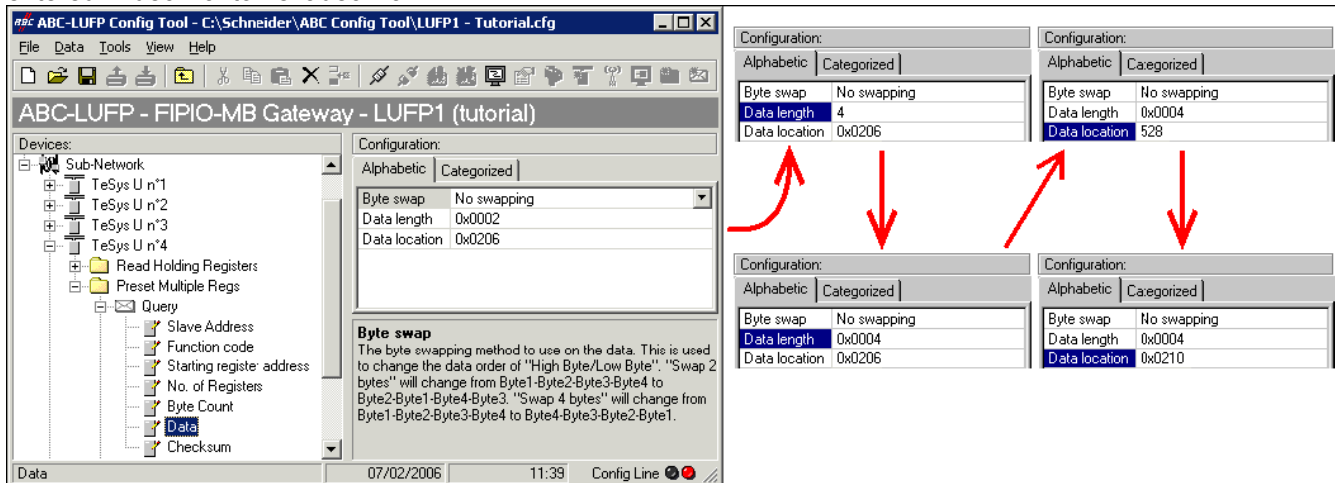
Using any other address in the gateway’s output data memory area (0x0234 to 0x03FF) will throw the gateway into **non-operational** mode, signaled by the sudden red-green flashing of LED 5 GATEWAY. LEDs 1, 2, 3 and 4 will also flash, at a frequency of 2 Hz.

You can ignore the size indications on the graphics in this window (“In Area 16 bytes” and “Out Area 16 bytes”), as the gateway has a fixed size for inputs and outputs alike.

If you want to memories the 4 bytes of Modbus data which will be transmitted by the gateway for this command, once the changes have been made, you will have to either shift all the other transmitted data by 2 bytes—a painstaking operation—or else change the memory location of the block of data transmitted. In the example described here, we will be using the second solution, although the first solution is actually preferable, in principle, as it avoids leaving any “holes” in the gateway’s memory. This has no impact on the FIPIO exchanges; however, as the size of transfers over the gateway is set at 32 input words and 32 output words by the use of the standard profile FED C32 (P).

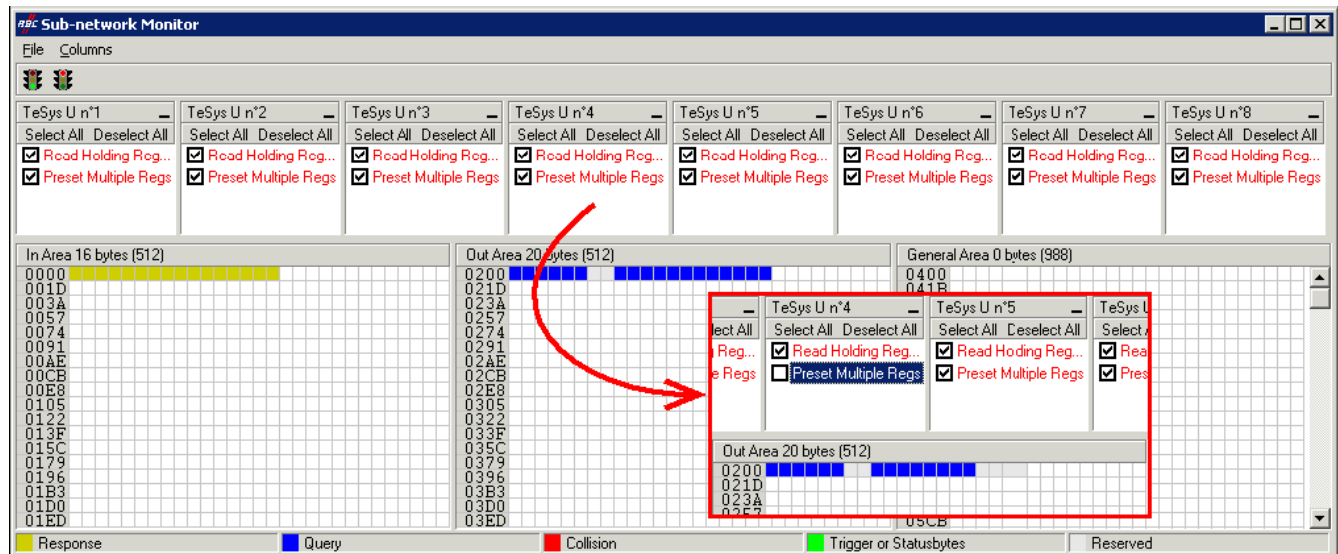
We will be placing the 4 bytes of data starting at address 0x0210 (528 in decimal), i.e. directly after the output data for the standard configuration. The 2 bytes at 0x0206 and 0x0207 become “free memory locations”.


Close the “Sub-network Monitor” window, then, once you are back in the main ABC-LUFP Config Tool window, select the “Data length” and “Data location” fields respectively in the “Data” element of the “Query” and change their values as shown at the top of the next page. ABC-LUFP Config Tool will automatically convert any value entered in decimal to hexadecimal.



## 7. Using ABC-LUFP Config Tool

To check that these changes have been integrated into the configuration, choose the “Monitor” command again in the “Sub-Network” menu:



- 4) Download this configuration to the gateway: See Section 7.5. Check that the configuration is valid (LED  GATEWAY flashing green).
- 5) Save the configuration to your PC's hard disk.
- 6) Use the FIPIO master PLC outputs: It only remains to assign symbols and use the outputs corresponding to the new Modbus data sent to motor starter n°4.

This gives us the mapping shown on the next page, derived from the one used for the standard gateway configuration. Changes in relation to the standard configuration are flagged by a grey background (except for the two reserved words %QWp.2.c\0.0.26 and %QWp.2.c\0.0.27).

## 7. Using ABC-LUFP Config Tool

Service	PLC output	Description	
		Bit 15.....Bit 8	Bit 7.....Bit 0
Periodic communications — Command of TeSys U motor starters	%QW\p.2.c\0.0	Value of command register for motor starter ①	
	%QW\p.2.c\0.0.1	Value of command register for motor starter ②	
	%QW\p.2.c\0.0.2	Value of command register for motor starter ③	
	%QW\p.2.c\0.0.3	Free memory location	
	%QW\p.2.c\0.0.4	Value of command register for motor starter ④	
	%QW\p.2.c\0.0.5	Value of command register for motor starter ⑤	
	%QW\p.2.c\0.0.6	Value of command register for motor starter ⑥	
	%QW\p.2.c\0.0.7	Value of command register for motor starter ⑦	
Periodic communications: — Command of TeSys U motor starter ⑧	%QW\p.2.c\0.0.8	Value of "Command Register"	
	%QW\p.2.c\0.0.9	Value of "2nd Command Register"	
Periodic communications — Free locations	%QW\p.2.c\0.0.10 ...to... %QW\p.2.c\0.0.25	Not used (16 words)	
	%QW\p.2.c\0.0.26 %QW\p.2.c\0.0.27	Reserved (2 words)	
Aperiodic communications — Indexed periodic variables (PKW) service – COMMAND	%QW\p.2.c\0.0.28	PKE – Address of the datum to be read / written	
	%QW\p.2.c\0.0.29	DN – Address of the queried slave	R/W – Read / Write command
	%QW\p.2.c\0.0.30	PWE – Value to be written (1st word)	
	%QW\p.2.c\0.0.31	PWE – Value to be written (2nd word)	

### 7.11. Changing a Modbus Slave Configuration

Configuring a Modbus slave itself remains very simple because it only involves the name and the Modbus address of the node to which it corresponds. By contrast, configuring Modbus commands is a more complex business and is discussed in a separate section (see Section 7.12).

You will need to change the configuration of a Modbus slave when you add a new Modbus device (see Section 7.9), using any method.

Changing the name of the node which corresponds to a Modbus slave helps to distinguish it from the other nodes, for instance when its Modbus command configuration has been changed.

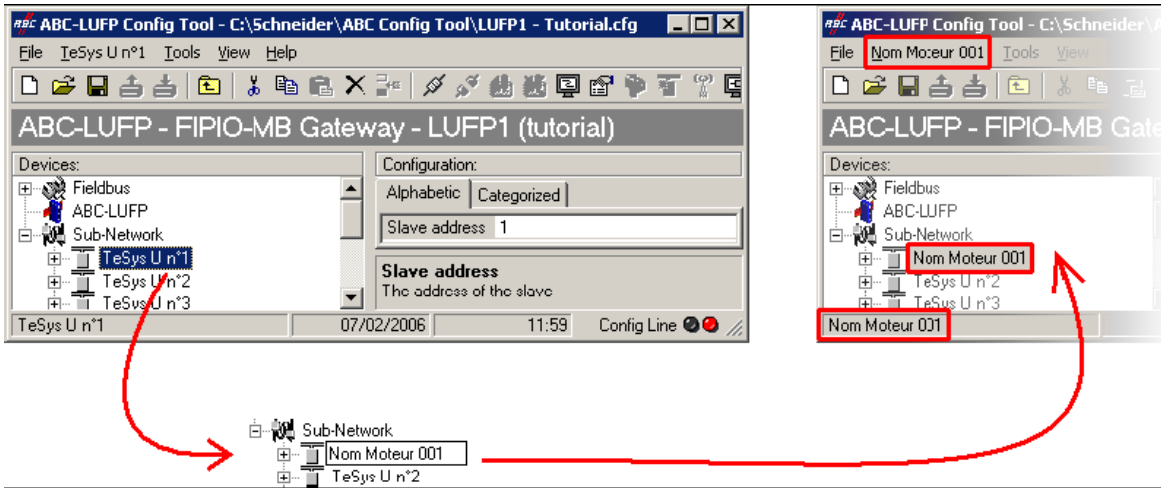
#### 7.11.1. Changing the Name of a Modbus Slave

To carry out this operation, all you have to do is select the node that corresponds to the Modbus slave involved ("Devices:" pane), and perform one of the following four actions:

- right-click on the node and click on "Rename" in the popup menu that appears, **or**
- select the node and click on the node's name, **or**
- select the node and choose "Rename" in the menu whose name matches the node's name, **or**
- use the F2 function key.

After confirming the new name ("Enter" key or click outside the node's name), it will be used to update the menu bar and the status bar of ABC-LUFP Config Tool. An example is given below. The three red frames shown in this example show the consequences of the change made.

# 7. Using ABC-LUFP Config Tool



## 7.11.2. Changing the Address of a Modbus Slave

To carry out this operation, simply select the node which corresponds to the Modbus slave involved (“Devices:” pane), click on the value of the current address (value of the “Slave address” field in the “Configuration:” pane), and change it.

**Reminder:** The address of a Modbus slave must be between 1 and 247. The system will not let you add a value > 247.

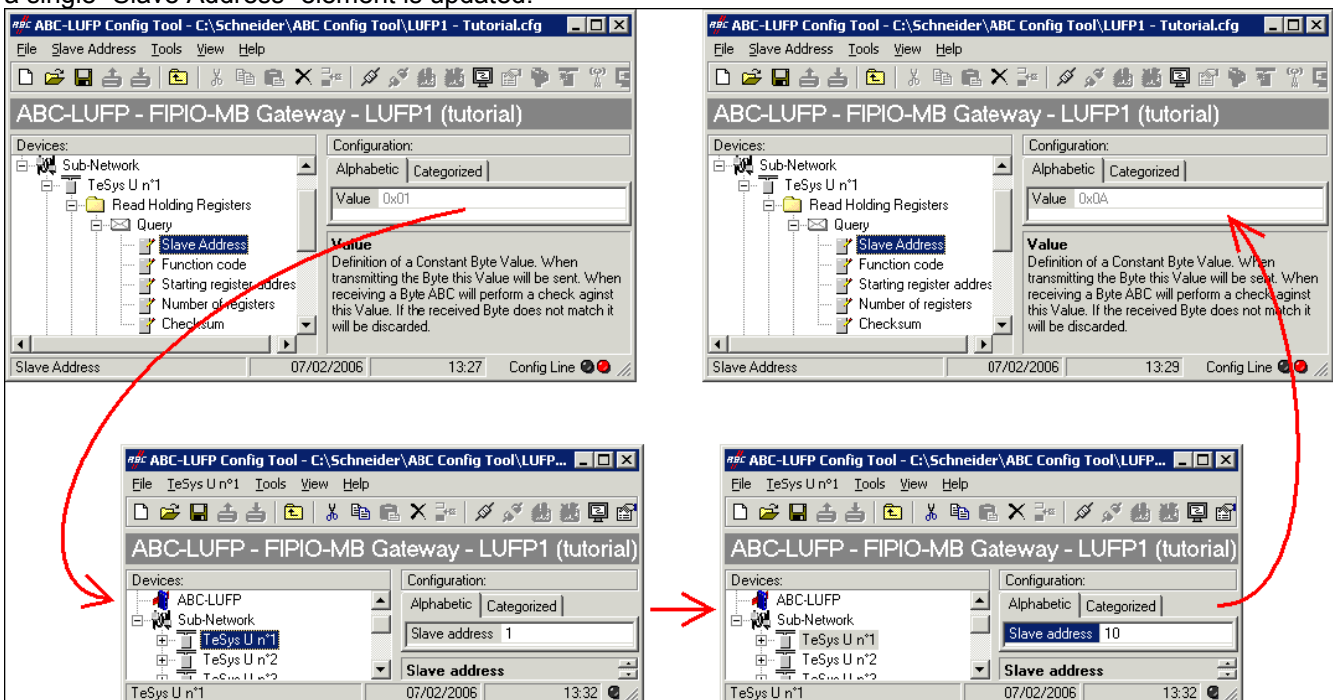
### ⚠ WARNING

#### USE OF RESERVED MODBUS ADDRESSES

Do not use Modbus addresses 65, 126, or 127 if a gateway’s Modbus slaves will include a Schneider Electric Adjustable-Speed Drive System device such as an Altistart soft-starter or an Altivar motor drive. The Altistart and Altivar devices reserve these addresses for other communications, and the use of these addresses in such a system can have unintended consequences.

**Failure to follow this instruction can result in death, serious injury, or equipment damage.**

When you confirm the new address (“Enter” key or click outside the Modbus slave address input field), it becomes effective in ABC-LUFP Config Tool, and the values of the “Slave Address” elements in the queries and responses of the Modbus commands for the selected node are automatically updated. In the example that follows, a single “Slave Address” element is updated:



# 7. Using ABC-LUFP Config Tool

## 7.11.3. Changing the Name of a Modbus Command

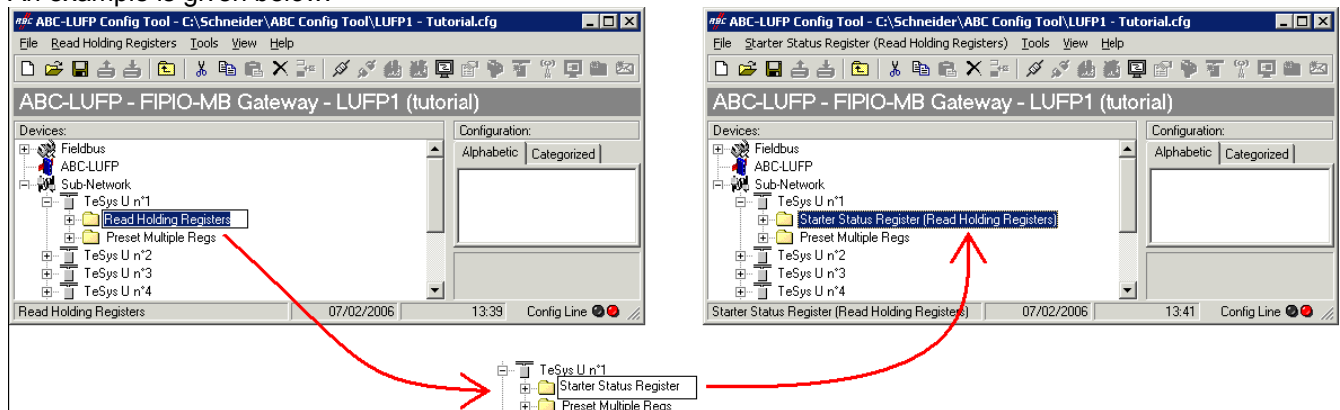
To rename a Modbus Command, first perform one of the following actions:

- right-click on the name of the command itself (e.g. Preset Multiple Regs) and click on “Rename” in the popup menu that appears, **or**
- select the name of the command and choose “Rename” in its corresponding menu, **or**
- select the name of the command and click inside its name, **or**
- select the name of the command and press the F2 key.

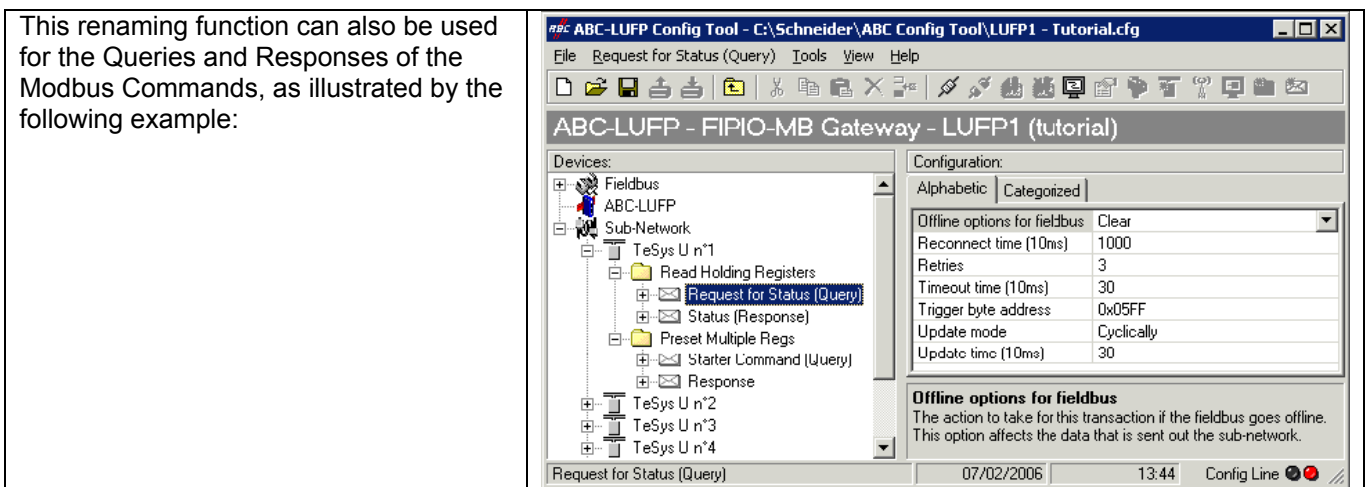
Then, type the new name of the command, and confirm it (“Enter” key or click outside the name’s field) or cancel it (“Escape” key). Once confirmed, the new name will become effective in ABC-LUFP Config Tool.

The type of command is automatically appended at the end of the Modbus command's new name.

An example is given below:



This renaming function can also be used for the Queries and Responses of the Modbus Commands, as illustrated by the following example:



## 7.12. Adding and Setting up a Modbus Command

### 7.12.1. With TeSys U Motor Starters

With TeSys U motor starters, the main point of adding a Modbus command is to enable you to command or control additional registers, without having to change the elements in the standard configuration. The periodic and aperiodic communication services therefore operate in the same way as for the standard configuration, unlike the various operations described in Section 7.10.

Instead of adding a command and configuring it from scratch, it is easier to copy one of the two default commands for TeSys U motor starters—“Read Holding Registers” (read / control) or “Preset Multiple Regs” (write / command)—and paste it into the list of Modbus commands for the appropriate node.

To copy a ready-configured Modbus command, select it, and then choose “Copy” from the menu with the same name as the selected command. **Keyboard shortcut:** “Ctrl C”. Then proceed with either of the following methods:

- a) Select the node corresponding to the Modbus slave you want to add the command to (e.g. “TeSys U n°4”), then choose “Paste” from the menu with the same name as the selected node. A new command is added after

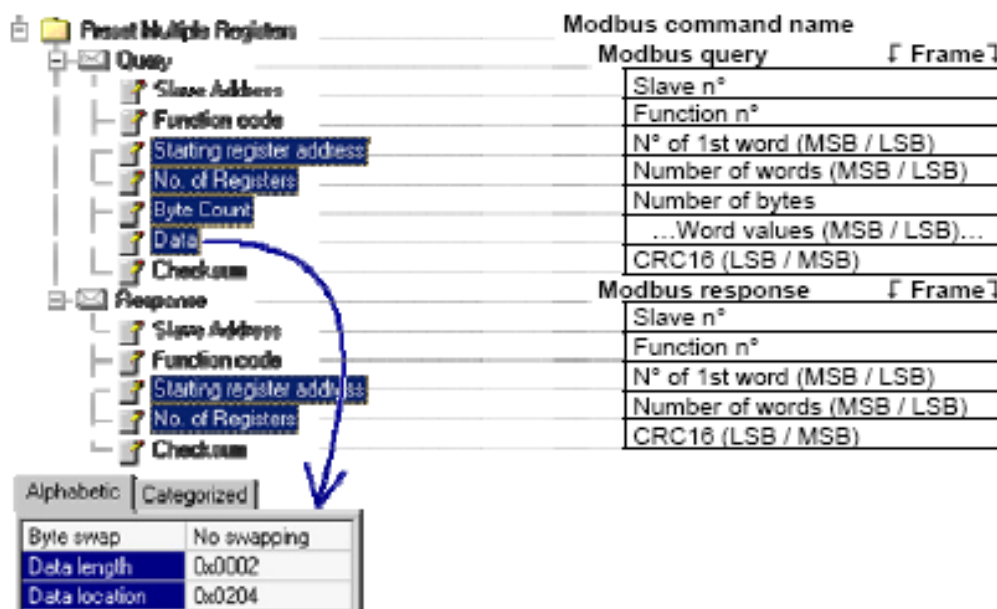
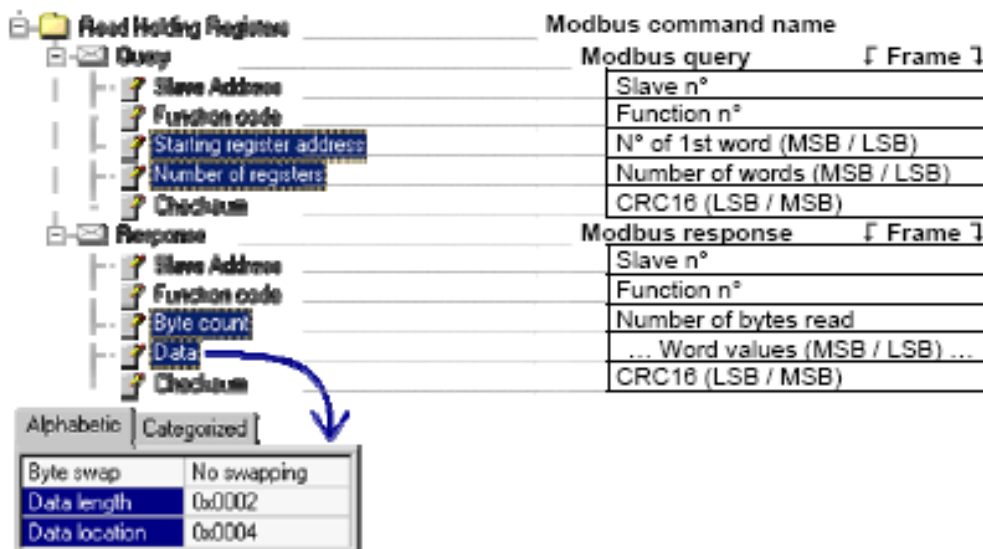


## 7. Using ABC-LUFP Config Tool

all the other commands configured for this node. Its configuration is totally identical to that of the previously copied command. **Keyboard shortcut:** “Ctrl V”.

- or:
- b) Select one of the commands on the node concerned, then choose “Insert” from the menu with the same name as the selected command. A new command is added just before the one selected. Its configuration is totally identical to that of the previously copied command.

As the new Modbus command and the original Modbus command are identical, you will need to make changes to the fields **highlighted in blue** in one of the two diagrams below, depending on whether it is a “Read Holding Registers” command or a “Preset Multiple Registers” command (see Section 7.10). The correspondence between the different elements in these tree structures and the standard Modbus terminology is shown on the right:



**NOTE:** In all cases, the “Query / Slave Address” and “Response / Slave Address” elements are automatically updated by ABC-LUFP Config Tool according to the node in which the command is located. Their values are not usermodifiable. Likewise, the “Query / Function code” and “Response / Function code” fields depend on the nature of the Modbus command and cannot be changed by you.

The operations to be carried out are much the same as those used for changing the default commands. For the “Read Holding Registers” command, see Section 7.10.1 and Section 7.10.3. For the “Preset Multiple Regs” command, see Section 7.10.2 and Section 7.10.4.

# 7. Using ABC-LUFP Config Tool

## 7.12.2. With a Generic Modbus Slave

Unlike in the previous section, here we will be looking at adding and setting up a Modbus command which is different from those configured by default with the LUF1 gateway. We will use this opportunity to take an in-depth look at the fields with which you can set up communications for such commands.

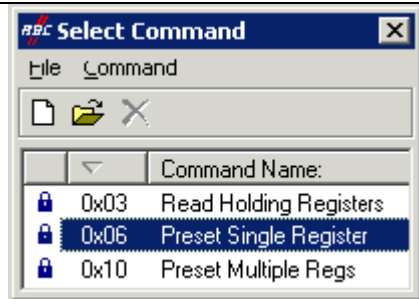
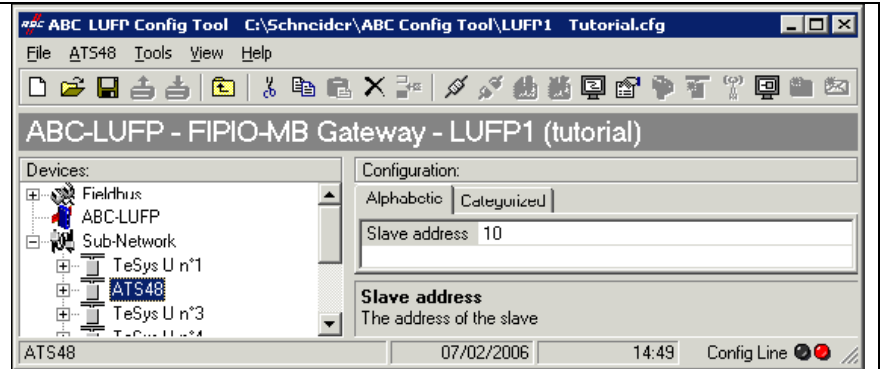
See Appendix E for a list and description of the Modbus functions supported by the LUF1 gateway. Only a very few Modbus commands are authorized, in order to limit the complexity of configuring Modbus exchanges on the gateway. Moreover, ABC-LUFP Config Tool cannot be used to configure new Modbus commands (created from scratch) for LUF1 gateways.

*E.g.* To illustrate the various operations to be carried out and the explanations given, we will be taking the example of a Altistart starter, the ATS48, and a Modbus command recognized both by the gateway and the ATS48. This is the “Preset Single Register” command, whose function code is 6 and which allows you to write the value of a single output word. This function will be used to periodically write the value of the ATS48’s CMD command register, located at address W400 (address 400 = 0x0190).

As the standard gateway configuration already has 8 Modbus slaves, you will need to delete one of them, such as the “TeSys U n°2” node, for example, and to add a new node in its place (see Section 7.8, and Section 7.9).

After creating the new node, we rename it and assign it Modbus address 10, as shown at right:

We then add the “Preset Single Register” command by choosing “Add Command” from the “ATS48” menu.

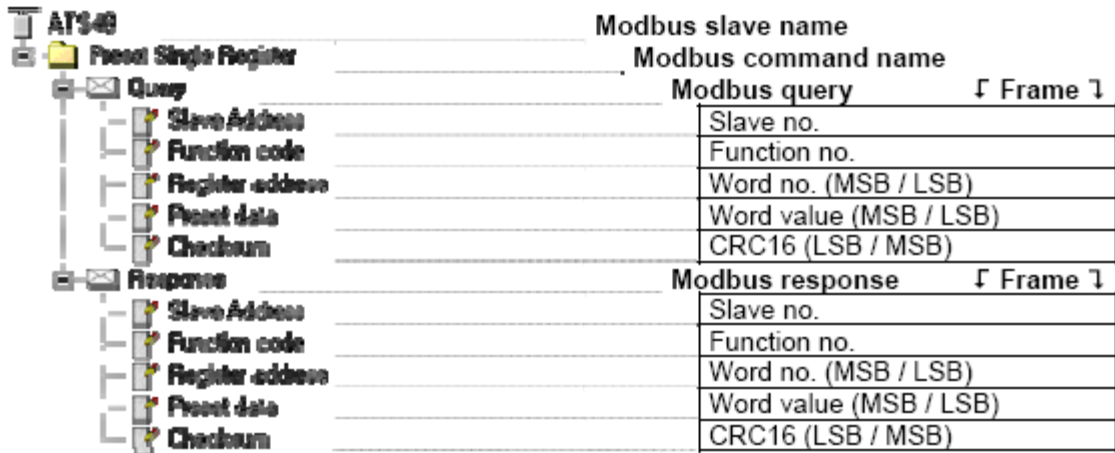


In the window which appears (opposite), select the command “0x06 Preset Single Register” and choose “Select” in the “File” menu.

Back in the main ABC-LUFP Config Tool window, the “Preset Single Register” command now appears in the list of Modbus commands for the “ATS48” node.

# 7. Using ABC-LUFP Config Tool

Expand the full tree structure for this command, as shown below. The correspondence between the different elements in this tree structure and the standard Modbus terminology is shown on the right:



These elements can be configured using ABC-LUFP Config Tool. There is a description of them in the following sections. We will then return to the example of the ATS48 to illustrate how to use these elements.

### 7.12.2.1. Managing Degraded Modes

Due to the number of hardware elements and software tools used, the following table shows a summary of the various degraded modes available to a FIPIO application. In this instance, the application is running on a LUFP1 gateway and a Premium PLC with a FIPIO master coupler on its processor card.

Desired behavior		Event			
		Premium PLC: CPU stop/failure	Disconnection of the upstream FIPIO network (1)	LUFP1 gateway failure	Disconnection of the downstream Modbus RTU network (1) (3)
Outputs	Reset	"Offline options for fieldbus" = "Clear"		Depends on the configuration of the Modbus slaves (2)	
	Hold	"Offline options for fieldbus" = "Freeze"			
	No refresh	"Offline options for fieldbus" = "NoScanning"			
Inputs	Reset	---	---	"Offline options for sub-network" = "Clear"	
	Hold	---	Yes (4)	"Offline options for sub-network" = "Freeze"	

- (1) The "Offline options for fieldbus" and "Offline options for sub-network" are described in the next section.
- (2) The desired behavior with regard to outputs should be directly configured on each of the Modbus slaves. In the case of drives marketed by *Schneider Electric*, for instance, the outputs are reset by setting the NTO bit to 0 (command with communication control), and held by setting NTO to 1 (command without communication control).
- (3) As regards inputs, we strongly recommend that you use the List of Active Slaves (LAS) Service (see Section 5.2) to detect the underlying event.
- (4) Use the gateway's FIPIO Diagnostic Objects (Section 5.4) to detect the underlying event.

# 7. Using ABC-LUFP Config Tool

## Managing Degraded Modes (7.11.2.1)

### PLC processor stopped or on failure

<b>PLC processor response</b>
Outputs: Software error, outputs reset to default state or hold their present state depending on configuration. Hardware error (EEPROM or hardware failure), output state will be undetermined
Inputs: PLC stops responding to inputs in any error state.
<b>FIPIO master response</b>
Depending on the configuration of the FIPIO master: the FIPIO master stops to communicate with the LUFP1 gateway, <b>or</b> the FIPIO master forces outputs to 0 and refreshes the inputs, <b>or</b> the FIPIO master holds outputs in their last state and refreshes the inputs.
<b>LUFP1 gateway response</b>
If the FIPIO master stops communicating with the LUFP1 gateway (in the case of a Premium PLC), for each Modbus Command, the behavior depends on its "Offline options for fieldbus": Clear: The output data sent to the Modbus slave concerned is reset to 0 and the input memory continues to be refreshed with the data of the slaves' response. Freeze: The output data sent to the Modbus slave concerned retains its current value and the input memory continues to be refreshed with the data of the slaves' response. No Scanning: The query of this Modbus Command is no longer transmitted and the input memory is no longer refreshed with the data of the slaves' response.
If the FIPIO master forces outputs to 0 and refreshes the inputs, the periodic Modbus exchanges continue to run: all output Data are reset to 0 and sent to the Modbus slaves, the input memory continues to be refreshed with the Data of the Modbus slaves' responses,
If the FIPIO master forces holds outputs in their last state and refreshes the inputs, the periodic Modbus exchanges continue to run: all output Data stay in their last state and are sent to the Modbus slaves, the input memory continues to be refreshed with the Data of the Modbus slaves' responses,
<b>TeSys U response</b>
Depending on the slave.

### FIPIO scanner stopped or on failure

<b>PLC processor response</b>
The PLC processor provides some error and/or diagnostic objects to the application if FIPIO master stop or failure (input/output not valid) occurs. Refer to the PLC user manual for their description. This information must be managed in the PLC application.
<b>FIPIO master response</b>
If the FIPIO master is stopped (command coming from the application): the scanner stops communicating with the LUFP1 gateway.
If the FIPIO master is on failure: the scanner stops communicating with the processor and the LUFP1 gateway.
<b>LUFP1 gateway response</b>
If the FIPIO master stops communicating with the LUFP1 gateway (in the case of a Premium PLC), for each Modbus Command, the behavior depends on its "Offline options for fieldbus": Clear: The output data sent to the Modbus slave concerned is reset to 0 and the input memory continues to be refreshed with the data of the slaves' response. Freeze: The output data sent to the Modbus slave concerned retains its current value and the input memory continues to be refreshed with the data of the slaves' response. No Scanning: The query of this Modbus Command is no longer transmitted and the input memory is no longer refreshed with the data of the slaves' response.
<b>TeSys U response</b>
Depending on the slave.

## 7. Using ABC-LUFP Config Tool

---

### LUFP1 gateways disconnected on FIPIO side

<b>PLC response</b>
The PLC processor provides some error and/or diagnostic objects to the application if slave disconnection from the application occurs. Refer to the PLC user manual for their description. This information must be managed in the PLC application.
<b>FIPIO master response</b>
The FIPIO master provides the processor with some error and diagnostic objects if FIPIO slave disconnection occurs.
<b>LUFP1 gateway response</b>
If the FIPIO master stops communicating with the LUFP1 gateway (in the case of a Premium PLC), for each Modbus Command, the behavior depends on its "Offline options for fieldbus": Clear: The output data sent to the Modbus slave concerned is reset to 0 and the input memory continues to be refreshed with the data of the slaves' response. Freeze: The output data sent to the Modbus slave concerned retains its current value and the input memory continues to be refreshed with the data of the slaves' response. No Scanning: The query of this Modbus Command is no longer transmitted and the input memory is no longer refreshed with the data of the slaves' response.
<b>TeSys U response</b>
Depending on the slave.

### LUFP1 gateways failure

<b>PLC response</b>
The PLC processor provides some error and/or diagnostic objects to the application if slave failure to the application occurs. Refer to the PLC user manual for their description. This information must be managed in the PLC application.
<b>FIPIO master response</b>
The FIPIO master provides the processor with some error and diagnostic objects if FIPIO slave failure occurs.
<b>LUFP1 gateway response</b>
In case of a failure, the gateway stops communicating with the FIPIO master and the Modbus slaves.
<b>TeSys U response</b>
Depending on the slave.

## 7. Using ABC-LUFP Config Tool

---

### LUFP1 gateways disconnected on Modbus side or slave failure

#### PLC response

The processor has access to the List of Active Slaves (LAS) coming from the FIPIO master input table. This word must be managed in the PLC application in order to detect if a Modbus slave is missing.

#### FIPIO master response

The FIPIO master continues communicating with the PLC processor and the LUFP1 gateway.

#### LUFP1 gateway response

Depending on the following properties, all of which are configurable for each Modbus Command:

Timeout time (10 ms) =  $x \times 10$  ms,

Retries =  $y$ ,

Reconnect time (10 ms) =  $z \times 10$  ms, *and*

Offline options for sub-network = Clear / Freeze.

After sending a request to a slave, if there is no response after ( $x \times 10$ ) ms, the gateway will send it again  $y$  times before writing information about the missing slave to the List of Active Slave (LAS), if no other Modbus Command has succeeded for this slave.

If, despite the  $y$  Retries, there is no answer from the slave, the input Data sent to the FIPIO master (Read requests) is reset to 0 (if Clear) or retains its current value (if Freeze), but only for the data received from the missing slave.

The gateway will try to reconnect the missing slave by communicating with it again every ( $z \times 10$ ) ms.

#### Slave response

If the LUFP1 gateway is disconnected on Modbus side, the behavior depends on the slave.

In the event of a slave failure:

No response is sent to the gateway and the state of the slave is undetermined.

This case **must** be managed in the PLC application.

## 7. Using ABC-LUFP Config Tool


### 7.12.2.2. Configuring the Query

Select the "Query" element from the Modbus command. The various elements in the configuration of the query for this command are shown opposite. The values displayed correspond to the default values for any new command.

These elements allow you to configure how the whole command is managed, including how degraded modes are managed (number of retransmissions, for example).

Alphabetic	Categorized
Offline options for fieldbus	Clear
Reconnect time (10ms)	1000
Retries	3
Timeout time (10ms)	100
Trigger byte address	0x05FF
Update mode	Cyclically
Update time (10ms)	100

Each of these elements is described, in order, in the table below. When a unit is assigned to an element, it is shown in brackets after the name of that element.

Configuration element	Description
Offline options for fieldbus	This element affects the data <b>sent to the Modbus slave</b> , only for the command to which the element belongs, whenever the gateway is disconnected from the FIPIO network. It takes one of the following three values: <ul style="list-style-type: none"> <li>- Clear .....All data sent to the Modbus slave using this command is now set to 0x0000 (resetting the output data in the gateway's memory).</li> <li>- Freeze.....All data sent to the Modbus slave using this command retains its current value (freezing the output data in the gateway's memory).</li> <li>- NoScanning ..The command is no longer transmitted to the Modbus slave by the gateway.</li> </ul>
Reconnect time (10ms)  Default value: 10ms x 1000 = 10s	If there is no response from the Modbus slave to a query, or following the receipt of an incorrect response, the gateway uses the "Retries" and "Timeout time (10ms)" elements to carry out retransmissions. If the Modbus slave has still not responded correctly after these retransmissions, the gateway stops sending it the corresponding query for a period of time which can be adjusted using "Reconnect time (10ms)". When this period is over, the gateway attempts to restore communication with the Modbus slave.
Retries Default value: 3	This element indicates the number of retransmissions carried out by the gateway if there is no response from the Modbus slave to a query, or if the response is incorrect. This retransmission process ceases as soon as the gateway gets a correct response within a given time. If none of the retransmissions elicits a correct response, the Modbus slave is deemed to be off-line, but only as regards the command in question. The gateway then uses the "Offline options for sub-network" and "Reconnect time (10ms)" elements, and LED  MODBUS becomes red. It will only revert to green when the Modbus command receives a correct response, following reconnection (see the element "Reconnect time (10ms)"). If the number of retransmissions is set to 0, this process will not be executed.
Timeout time (10ms)  Default value: 10ms x 100 = 1s	This element represents the time that the Modbus slave will wait for a response. If a response has not reached the gateway within the given time, as defined by this element, the gateway tries retransmitting. This process continues until it reaches the last retransmission allowed (see "Retries"), then the gateway declares the Modbus slave offline, but only for the command to which the element "timeout time (10ms)" belongs.

## 7. Using ABC-LUFP Config Tool

Configuration element	Description
Trigger byte address	<p>This element is only used by the gateway if “Update mode” is set to “Change of state on trigger”. In this case, it specifies the address, in the gateway’s output memory (0x0200 to 0x0233), of an 8-bit counter managed by the FIPIO master. The FIPIO master updates this counter in the same way as the other gateway outputs (see Section 5.1.3).</p> <p>When the value located at this address is changed by the FIPIO master to any value other than zero, the query configured with a “Change of state on trigger” for that address is transmitted to the Modbus slave by the gateway.</p> <p>Unlike the “On data change” and “Change of state or Cyclically” modes, this mode enables you to send a command on a specific order from the FIPIO master if, for example, the latter is unable to update all the data for any given query at the same time.</p> <p><b>NOTE:</b> The “trigger byte” does not have to be an output updated by the FIPIO master. It is quite possible for it to be an input between 0x0000 and 0x0033, in which case the exchanges of the command currently being configured will be conditioned by the Modbus slave that updates the byte.</p>
Update mode	<p>This element is used to specify the transmission mode for the query on the Modbus network. It takes one of the following four values:</p> <ul style="list-style-type: none"> <li>- Cyclically.....Default communication mode. The query is transmitted <b>periodically</b> over the Modbus network (see “Update time”).  <div style="background-color: yellow; display: inline-block; padding: 2px;">FED C32 P</div> This is the communication mode used systematically for all Modbus commands generated by the periodic control words configured by the user under PL7 PRO.</li> <li>- On data change ..... The gateway transmits the query over the Modbus network when at least one data item in the query is modified by the FIPIO master. This is therefore an <b>aperiodic</b> communication mode that works in a similar way to the indexed periodic variables (PKW) service described in Section 5.3. All the data from a single query must therefore be updated at the same time by the FIPIO master. If you are uncertain as to whether your software is able to update <b>all the output data</b> from a query at the same time, we recommend using “Change of State on trigger” mode rather than “On data change”.</li> <li>- Single Shot ..... This transmission mode only allows a single Modbus exchange for the whole of the time that the gateway is operating. This exchange takes place just after the initialization of the gateway.</li> <li>- Change of state on trigger ..... With this <b>aperiodic</b> communication mode, the Modbus query is sent every time that the FIPIO master changes the value of an 8-bit counter designated by the “Trigger byte address” element. See the description of this element for further information about how to use this communication mode.</li> <li>- Change of state or Cyclically .... In this communication mode—a combination of the “Cyclically” and “On data change” modes—the query is transmitted periodically over the Modbus network (see “Update time”), and also whenever one of the outputs in the query is modified by the FIPIO master.  <div style="background-color: yellow; display: inline-block; padding: 2px;">FED C32 P</div> This is the communication mode used systematically for all Modbus commands generated by the periodic command words configured by the user under PL7 PRO.</li> </ul>
Update time (10ms)  Default value: 10ms x 100 = 1s	<p>This element is only used by the gateway if “Update mode” is set to “Cyclically” or “Change of state or Cyclically”. In this case, it specifies the query’s transmission period on the Modbus network.</p>



## 7. Using ABC-LUFP Config Tool

For example, with the ATS48, we will be using the configuration shown opposite. The most notable points of this configuration are:

- The data are reset when either network is disconnected.
- 3 retransmissions with a 100 ms timeout.
- Periodic communications with a cycle time of 300 ms.

Alphabetic	Categorized
Offline options for fieldbus	Clear
Reconnect time (10ms)	1000
Retries	3
Timeout time (10ms)	10
Trigger byte address	0x05FF
Update mode	Cyclically
Update time (10ms)	30

### 7.12.2.3. Configuring the Response

Select the “Response” element from the Modbus command. The various configuration elements of the response for this command are shown opposite. The values displayed are the default values for any new command.

Alphabetic	Categorized
Offline options for sub-network	Clear
Trigger byte	Disabled
Trigger byte address	0x05FF

These elements allow you to configure just one command management feature, described below. The elements are described, in order, in the following table:

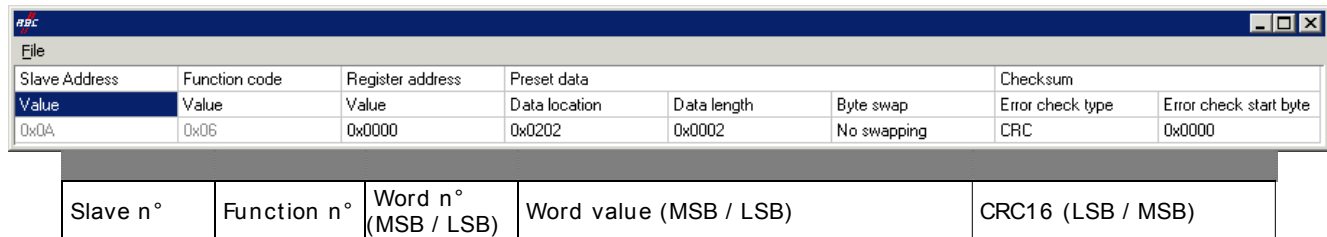
Configuration element	Description
Offline options for sub-network	<p>This element affects the input data <i>sent to the FIPIO master</i>, but only for the data of the Response to which this element belongs to, whenever the Modbus slave does not answer to the corresponding Query (or upon disconnection from the Modbus sub-network).</p> <p>This element takes one of the following two values:</p> <p>Clear All data sent to the FIPIO master for this Response is set to 0x0000 (resetting of the input data in the gateway’s memory).</p> <p>Freeze All data sent to the FIPIO master for this Response retain their current values (the input data in the gateway’s memory is frozen).</p>
Trigger byte	<p>This element is used by the gateway to activate the unitary incrementation of an 8-bit counter in order to notify the FIPIO master of the receipt of a new response to the associated Modbus command. It takes one of the following two values:</p> <p>- Disabled.....Default configuration. The gateway does not increment any counter on receipt of the Modbus response.</p> <p>- Enabled..... Each time that the gateway receives a new response to the associated Modbus command, it increments the value of an 8-bit counter designated by the “Trigger byte address” element (see below). This is useful when the command query is configured to use one of the <b>aperiodic</b> communication modes (see “Update mode”), as the counter mechanism allows the FIPIO master to ignore response input data except when the counter is incremented.</p>
Trigger byte address	<p>This element is only used by the gateway if the element “Trigger byte” is set to “Enabled”. In this case, it specifies the address, in the gateway’s input memory (0x0000 to 0x0033), of an 8-bit counter managed by the gateway. The FIPIO master reads the value of this counter in the same way as the other gateway inputs (see Section 5.1.3).</p> <p>When the gateway receives a response to the associated Modbus command, it increments the value of this counter by one unit (value = value+1).</p> <p>This mode keeps the FIPIO master informed when a new response is available. This can be useful, for example, where there is a chance that the data from two consecutive responses could be identical.</p>

For example, with the ATS48, we do not want the response to be event driven, so we will retain the default configuration.

## 7. Using ABC-LUFP Config Tool

### 7.12.2.4. Configuring the Content of the Query Frame

The window below is obtained using “Edit Transaction” from the “Query” menu. Unlike the tree structure in the main ABCLUFP Config Tool window, this display has the advantage of showing all of the frame’s fields along with their values. The values displayed below are the values assigned by default to the Modbus command query we have created. The correspondence with the content of the corresponding Modbus frame is shown underneath.



Slave Address	Function code	Register address	Preset data			Checksum	
Value	Value	Value	Data location	Data length	Byte swap	Error check type	Error check start byte
0x0A	0x06	0x0000	0x0202	0x0002	No swapping	CRC	0x0000

Slave n°	Function n°	Word n° (MSB / LSB)	Word value (MSB / LSB)	CRC16 (LSB / MSB)

Edit the values which are not grayed out, one after the other. There is a description of them below.

The nature of a frame’s fields depends on the Modbus command to which it corresponds. However, a certain number of these fields are common to all frames, whereas others are common to a number of them. The description of those shown above is given on the next page, as a part of the example described at the beginning of Section 7.12.2.

Field in the frame	Size in the frame	Description
Slave Address	1 byte	This field is not user-modifiable and its value is grayed out accordingly. ABC-LUFP Config Tool updates the value of this field automatically using the address of the Modbus slave corresponding to the current node.  <b>NOTE:</b> This field is common to all Modbus command queries.  <i>Example:</i> The value of this field is set to the address of the Modbus slave corresponding to the “ATS48” node, namely 0x0A.
Function Code	1 byte	This field is not user-modifiable and its value is grayed out accordingly. ABC-LUFP Config Tool updates the value of this field automatically using the function code of the corresponding Modbus command.  <b>NOTE:</b> This field is common to all Modbus command queries.  <i>E.g.</i> The value of this field is set to the code for the “Preset Single Register” command (write the value of an output word), namely 0x06.
Register Address	2 bytes	Address of an output word, or of a register, in the Modbus slave’s memory. This field designates the memory object to which the command relates.  <b>NOTE:</b> This field is common to all Modbus command queries which seek to access one or more locations in the memory of a Modbus slave. When accessing several memory locations, the “Register” field designates the address of the first word affected by the command.  <i>E.g.</i> The value of this field should be changed by entering the address of the CMD command register, namely 400 (0x0190). This value will be automatically converted to hexadecimal if the user enters it in decimal.
Preset Data	2 bytes, or more for a data block	<b>Data Location:</b> Address, in the gateway’s output data memory (0x0200 to 0x0233), of the datum to be transmitted in the “Preset Data” field of the query frame.  <b>NOTE:</b> The “Data location” field is used for each frame that conveys data between the Modbus slaves and the FIPIO master. In this instance, it designates the starting address of the data block to be transmitted.

## 7. Using ABC-LUFP Config Tool

### WARNING

#### RISK OF UNINTENDED EQUIPMENT OPERATION

You must use even values for the “Data Location” field (*i.e.* 514, 516, 518, etc.). The selection of odd locations complicates application programming.

**Failure to follow this instruction can result in death, serious injury, or equipment damage.**

Returning to our previous example, the value to be assigned to the ATS48’s CMD register should be placed in the gateway’s output data memory area. We will be using the first free location starting at an even address, *i.e.* 0x0210 in the case of the standard configuration.

Data length: Length of the output data block, in the gateway’s memory, of which the values are to be transmitted in the “Preset Data” field of the query frame. It is expressed in number of bytes.

**NOTE:** The “Data length” field is always used together with the “Data location” field, described above.

*E.g.* Since the “Preset Single Register” command is used to write the value of a single (16-bit) register, the value of the “Data length” field must be set to 2.

See the documentation for each Modbus slave to find out the maximum amount of 8-bit data which can be placed in “Data” type fields in the queries and responses for this slave. With the ATS48, for instance, the maximum is 30 16-bit words.

Byte swap: Specifies whether or not the output data bytes to be transmitted to the Modbus slave must be swapped before being placed in the Modbus frame. The three possible values are as follows:

- No swapping .....Default configuration. The data is sent in the same order as it appears in the gateway’s memory. **This is the case which must be used by default**, because for a 16-bit datum, the most significant byte (MSB) is placed first in the Modbus frame, and is always written MSB-first into the gateway’s memory by a FIPIO master.
- Swap 2 bytes.....The bytes to be transmitted are swapped two by two.
- Swap 4 bytes.....The bytes to be transmitted are swapped four by four. This is rarely used, as it only relates to 32-bit data. It works along the same lines as “Swap 2 bytes”.

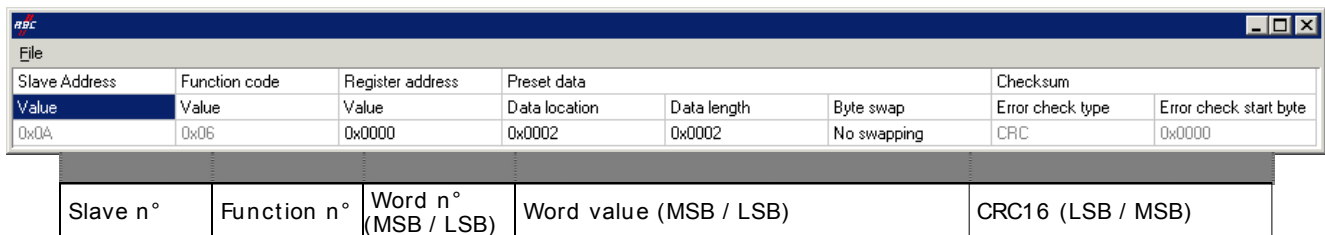
*Example:* We will be using the “No swapping” value, because the two bytes of the value to be written into the ATS48’s CMD register, as transmitted by the FIPIO master, are placed in the gateway memory MSB-first.

## 7. Using ABC-LUFP Config Tool

Field in the frame	Size in the frame	Description
Checksum	2 bytes	<p>Error check type: Type of error check for the frame.            - CRC.....Default method.            This is the method adopted for the Modbus RTU protocol. It cannot be changed.</p> <hr/> <p>Error check start byte: Indicates the number of the byte, in the frame, from which the “checksum” calculation should begin. The first byte in each frame carries the number 0.</p> <p><b>NOTE:</b> The calculation of a frame’s checksum should always begin with the first byte. Do not change the error check start byte from its default of zero. A non-zero value will result in an incorrect CRC, and all Modbus communications will return an error.</p>

### 7.12.2.5. Configuring the Content of the Response Frame

The window shown below is obtained using “Edit Transaction” from the “Response” menu. The values displayed are those assigned by default to the Modbus command response we have created. The correspondence with the content of the corresponding Modbus frame is shown underneath.



Slave Address	Function code	Register address	Preset data			Checksum	
Value	Value	Value	Data location	Data length	Byte swap	Error check type	Error check start byte
0x04	0x06	0x0000	0x0002	0x0002	No swapping	CRC	0x0000

Slave n°	Function n°	Word n° (MSB / LSB)	Word value (MSB / LSB)	CRC16 (LSB / MSB)

Edit the values which are not grayed out, one after the other.

There is a description of them below, but also see the previous section, as the nature of the content of response frames is very similar to that of the fields in Modbus query frames.

**NOTE:** If the value of one of the fields in the response from a Modbus slave is different from that configured via ABC-LUFP Config Tool, the response will be rejected by the gateway. It will then proceed to retransmit the query, provided that at least one retransmission has been configured for this command (see 7.12.2.2 Configuring the Query).

## 7. Using ABC-LUFP Config Tool

Field in the frame	Size in the frame	Description
Slave Address	1 byte	Identical to the query's "Slave Address" field.
Function Code	1 byte	Identical to the query's "Function" field
Register Address	2 bytes	Identical to the query's "Register" field, since the Modbus response of any "Preset Single Register" command is an echo to the corresponding query. Here you should also enter the address of the memory object to which the command relates. If receiving an exception code, see (*).
Preset Data	2 bytes, or more for a data block	Data Location: Address, in the gateway's input data memory (0x0000 to 0x0033), of the datum received in the "Preset Data" field of the response frame.  <b>NOTE:</b> As far as possible, ensure that the data is located at even-numbered addresses in order to align the Modbus data (in 16-bit format) with the %IW\p.2.c\0.0.** inputs of the FIPIO master.  <i>E.g.</i> The value sent back as an echo to the command must be placed in the gateway's input data memory area. We shall use the first two free bytes following the standard configuration input data, i.e. 0x0020-0x0021.

\* If receiving an exception code, the gateway re-transmits the request according to the number of retries that has been defined. Then, it will disconnect the slave.

### WARNING

#### RISK OF UNINTENDED EQUIPMENT OPERATION

You must use even values for the "Data Location" field (i.e. 0, 2, 4, 6, etc.). The selection of odd data locations complicates application programming and increases the likelihood of improper Modbus values being written to or read from the slave devices. Depending on the user's configuration, unintended equipment operation may result

**Failure to follow this instruction can result in death, serious injury, or equipment damage.**

Field in the frame	Size in the frame	Description
		<u>Data length</u> : Length of the block of input data received in the "Preset Data" field of the response frame. It is expressed in number of bytes.  <i>E.g.</i> The value of the "Data length" field must be set to 2. ----- <u>Byte swap</u> : Identical to the query's "Byte swap" field.  <i>E.g.</i> We will also be using the "No swapping" value, for the same reasons as with the query.
Checksum	2 bytes	<u>Error check type</u> : Identical to the query's "Error check type" field. <u>Error check start byte</u> : Identical to the query's "Error check start byte" field.  However, these two fields are not user-modifiable and their values are grayed out accordingly. ABC-LUFP Config Tool updates the values of these fields automatically using the values in the query's "Error check type" and "Error check start byte" fields.

# 7. Using ABC-LUFP Config Tool

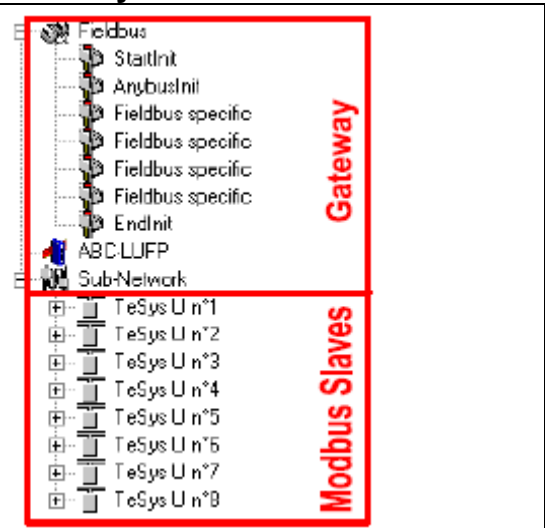
## 7.13. Configuring the General Characteristics of the Gateway

This operation relates to the gateway's general characteristics ("Fieldbus" to "Sub-Network" elements), whereas the previous sections described the configuration of the Modbus slaves (elements located under the "Sub- Network" element).

The "Fieldbus" element describes the upstream network, i.e. the FIPIO network in the case of the LUFP1 gateway.

The "ABC-LUFP" and "Sub-Network" elements describe the downstream network—i.e. the Modbus network in the case of the LUFP1 gateway—and identify the version number of the gateway software.

The configuration of these three elements, along with the commands they give access to, are described in the next three sections.

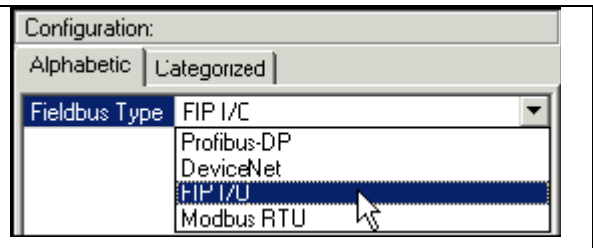


### 7.13.1. "Fieldbus" Element

Below this element is a list of the mailboxes configured by default. These elements are not described here, as they are only designed for the internal management of the gateway. The mailboxes can neither be modified nor deleted. Their number and nature depend on the type of upstream network.

When the "Fieldbus" element is selected, you can select the type of upstream network. With the LUFP1 gateway, **you must not modify the selection "FIP I / O"**.

If your PC is connected to the gateway using the PowerSuite cable and you are using ABC-LUFP Config Tool in "on-line" mode when ABC-LUFP Config Tool starts up, then the type of upstream network will be detected automatically.

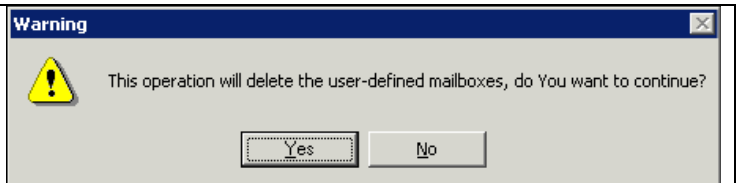


The only command accessible from the "Fieldbus" menu is "Restore Default Mailboxes". The usage of this command is recommended if you inadvertently inserted a user-defined Mailbox under the "Fieldbus" device. Since the usage of user-defined Mailboxes is not described in the present document, only the FIPIO Default Mailboxes should be set under the "Fieldbus" device, in the following order:

- StartInit
- AnybusInit
- Fieldbus specific
- Fieldbus specific
- Fieldbus specific
- Fieldbus specific

EndInit

Should any other Mailbox also appear in this list, please perform the "Restore Default Mailboxes" command. Then, confirm the operation by clicking on the "Yes" button in the confirmation / warning window that appears.

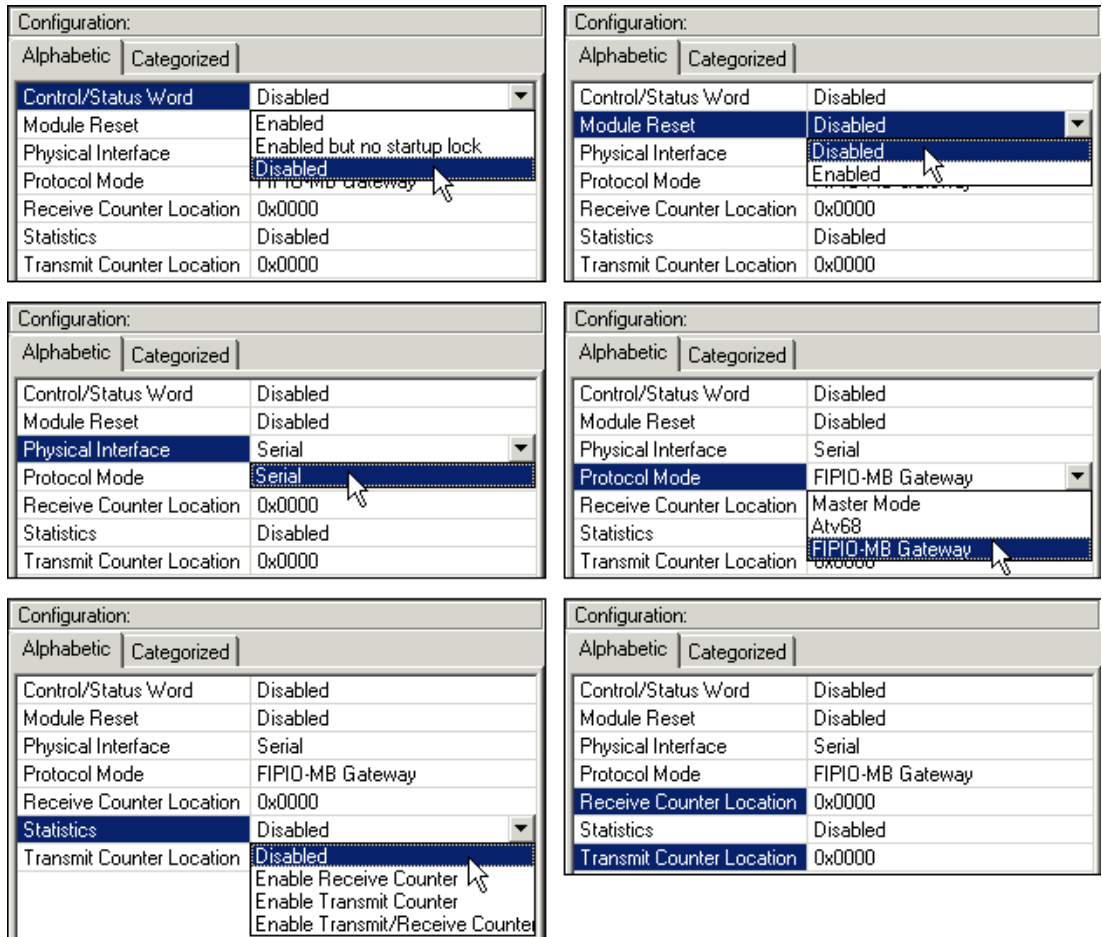


# 7. Using ABC-LUFP Config Tool

## 7.13.2. "ABC-LUFP" Element

The only command accessible from the "ABC-LUFP" menu is "Disconnect" (or "Connect" if you are in "off-line" mode). See chapter 7.3 for details about on-line" and off-line" modes.

In the configuration of the LUFF1 gateway's "ABC-LUFP" element, the "Control/Status Word", the "Physical Interface", and the "Protocol Mode" properties **must not be changed**. Their value, respectively, must always be set to "Disabled", "Serial", and "FIPIO-MB Gateway".



These seven properties allow you to configure some of the gateway's system aspects:

- **Control/Status Word:** The three possibilities available for this property are not described in the LUFF1 Gateway User Manual, as the option is reserved for other devices in the same product family. **This option must keep its existing value, i.e. "Disabled"**.
- **Module Reset:** By default, this property prevents the gateway from reinitializing itself when there is an internal operation problem. Changing this option is mainly intended for "laboratory" type use.
- **Physical Interface:** The only possibility offered by this property indicates that the physical interface of the downstream network of the gateway (Modbus) is a serial link.

# 7. Using ABC-LUFP Config Tool

- **Protocol Mode:** This property should not be changed, because it indicates the type of protocol used on the downstream network of the gateway. With the LUFP1 gateway, “FIPIO-MB Gateway” **must not be deselected**. The other possibilities available are reserved for other products from the same family as this gateway.

**NOTE:** Be careful not to modify the “Protocol Mode” property, as it would reset the entire configuration. If, by any chance, this should happen, start again with a clean basic configuration (see Section 7.6) or load a previously saved configuration.

- **Statistics:** This property dictates the presence or absence of the two Receive and Transmit Counters in the input memory of the gateway (see below). The four possibilities are:
  - Disabled The two “Receive Counter Location” and “Transmit Counter Location” properties are ignored.
  - Enable Receive Counter Only the “Receive Counter Location” property is used by the gateway.
  - Enable Transmit Counter Only the “Transmit Counter Location” property is used by the gateway.
  - Enable Transmit/Receive Counter Both the “Receive Counter Location” and “Transmit Counter Location” properties are used by the gateway.
- **Receive Counter Location:** This property is only used by the gateway if “Statistics = Enable Receive Counter” or “Statistics = Enable Transmit/Receive Counter”. It represents the 1-byte input memory address (from 0x0000 to 0x0033) where the Modbus Responses counter is copied into. This is a modulo 256 counter (*i.e.* it starts over at 0 once it goes past 255) which is updated each time a Modbus frame is received by the gateway.
- **Transmit Counter Location:** This property is only used by the gateway if “Statistics = Enable Transmit Counter” or “Statistics = Enable Transmit/Receive Counter”. It represents the 1-byte input memory address (from 0x0000 to 0x0033) where the Modbus Queries counter is copied into. This is a modulo 256 counter (*i.e.* it starts over at 0 once it goes past 255) which is updated each time a Modbus frame is emitted by the gateway, retries included.

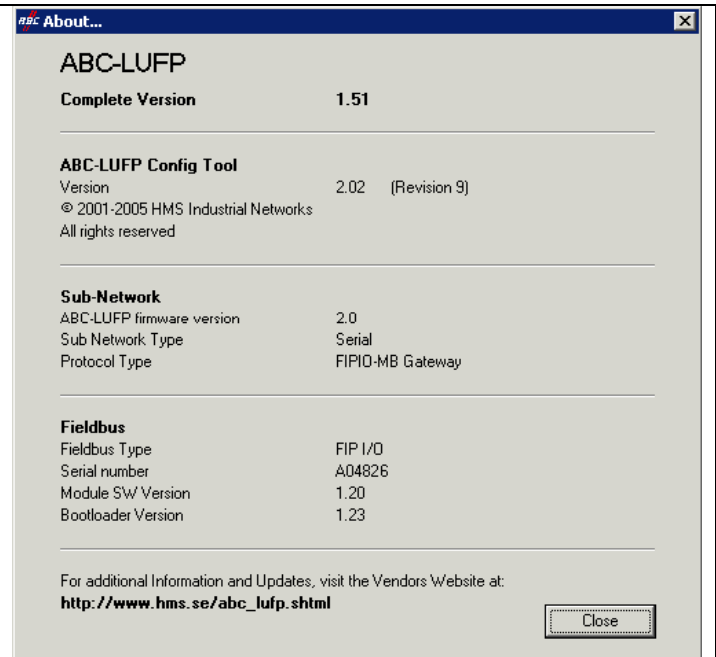
Finally, a useful command from the “Help” menu will allow you to check the software versions of the LUFP1 gateway (the “ABC-LUFP” element), but only in “on-line” mode; of course, it also shows the version of the ABC-LUFP Config Tool.

To obtain this information, execute the “About...” command of the “Help” menu. An example in “online” mode is given here:

In “off-line” mode, all versions and information from the “Sub-Network” and “Fieldbus” categories are replaced with “Unknown” since they could not be obtained from an existing and connected gateway.

The “[http://www.hms.se/abc\\_lufp.shtml](http://www.hms.se/abc_lufp.shtml)” text is a hypertext link. By clicking on it, you are directly redirected to the **Schneider Electric's** Web page dedicated to the ABC-LUFP gateways.

This page features many downloadable items related to the family of LUFP• gateways, including the latest version of ABC-LUFP Config Tool.





## 7. Using ABC-LUFP Config Tool

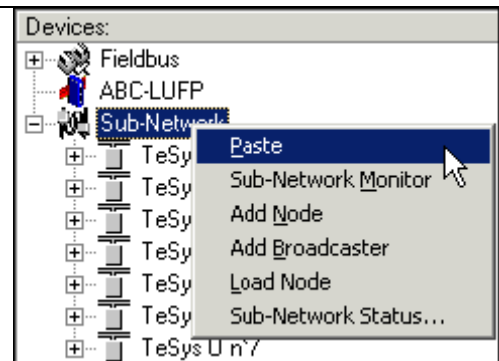
### 7.13.3. “Sub-Network” Element

The five commands accessible from the “Sub-Network” menu are:

- “Paste”: Appends a copy of the last copied node (after a “Copy” command on an existing node), or a replica of the cut node (after a “Cut” command), to the list of nodes of the “Sub-Network” element. This command is only available if a node has been previously copied or cut, and only if the 8 nodes limit has not been reached yet.

- “Sub-Network Monitor”: Allows you to view the mapping between the Modbus command data and the content of the gateway’s memory.

Examples of how to use this command are shown in Sections 7.10.3 and 7.10.4.



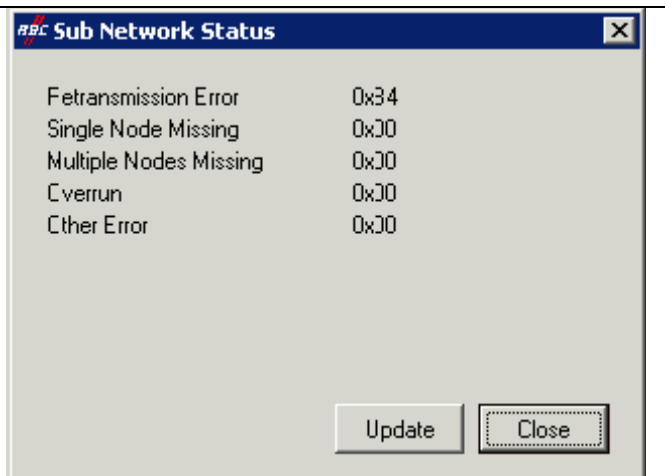
- “Add Node”: Allows you to add a new node on the downstream Modbus network. Each node corresponds to a different Modbus slave. This command is not available if there are already 8 Modbus slaves, as is the case with the gateway’s standard configuration.

- “Add Broadcaster”: Allows you to add a broadcaster node (see Section 7.13.4).

- “Load Node”: Allows you to add a pre-configured node on the downstream Modbus network. The configuration for this node is contained in an XML file (see the section on “Importing/Exporting a Modbus Slave Configuration” in Section 7.9). This command is not available if there are already 8 Modbus slaves, as is the case with the gateway’s standard configuration.

- “Sub-Network Status...”: In “on-line” mode (see Section 7.13.2), this command opens a window displaying the values of the gateway’s error counters. The “Update” button allows you to refresh the values of these counters.

When you run this command in “off-line” mode, all of the values are replaced by the word “Unknown” to show that they cannot be read off the gateway. The “Update” button then becomes inaccessible.



**NOTE:** The “Sub Network Status” window may be useful to detect problems on the Modbus sub-network. So, if the number of retransmission errors increases upon using the “Update” button, this denotes the absence of one or more slaves, Modbus cabling or speed problems, or invalid Commands. Since retransmission errors tend to lower the general performances of the Modbus communications, you should undertake actions to prevent these retransmission errors from increasing.

## 7. Using ABC-LUFP Config Tool

When the “Sub-Network” element is selected, you have access to all of the options allowing you to configure the gateway’s communication protocol format on the Modbus network. The various settings you can make are described below. All of the Modbus slaves present must support these settings and must be configured appropriately.

- Bitrate (bits/s): The gateway supports a limited number of communication speeds. Choose the one that suits your Modbus network.
- Data bits: 8 bits (required).
- Parity: Choose the parity according to the format chosen for communications on your Modbus network.
- Physical standard: RS485 (required).
- PKW Broadcast Interval (10ms): When the PKW service is used in broadcast mode (DN = 255), this option defines the interval following the transmission of the broadcast command. The next Modbus message, of whatever type, will not be sent by the gateway until this period has elapsed. The interval must therefore be long enough to give the slowest Modbus slave the time to process the broadcast command.

Configuration:	
Alphabetic   Categorized	
Bitrate (bits/s)	19200
Data bits	1200
Parity	2400
Physical standard	4800
PKW Broadcast Interval (10ms)	9600
PKW Retries	19200
	3

Configuration:	
Alphabetic   Categorized	
Bitrate (bits/s)	19200
Data bits	8
Parity	7
Physical standard	8
PKW Broadcast Interval (10ms)	100

Configuration:	
Alphabetic   Categorized	
Bitrate (bits/s)	19200
Data bits	8
Parity	None
Physical standard	None
PKW Broadcast Interval (10ms)	Odd
PKW Retries	Even

Configuration:	
Alphabetic   Categorized	
Bitrate (bits/s)	19200
Data bits	8
Parity	None
Physical standard	RS485
PKW Broadcast Interval (10ms)	RS232
PKW Retries	RS485
PKW Timeout time (10ms)	100

Configuration:	
Alphabetic   Categorized	
Bitrate (bits/s)	19200
Data bits	8
Parity	None
Physical standard	RS485
PKW Broadcast Interval (10ms)	100

Configuration:	
Alphabetic   Categorized	
Bitrate (bits/s)	19200
Data bits	8
Parity	None
Physical standard	RS485
PKW Broadcast Interval (10ms)	100
PKW Retries	3

Configuration:	
Alphabetic   Categorized	
Bitrate (bits/s)	19200
Data bits	8
Parity	None
Physical standard	RS485
PKW Broadcast Interval (10ms)	100
PKW Retries	3
PKW Timeout time (10ms)	100

Configuration:	
Alphabetic   Categorized	
Bitrate (bits/s)	19200
Data bits	8
Parity	None
Physical standard	RS485
PKW Broadcast Interval (10ms)	100
PKW Retries	3
PKW Timeout time (10ms)	100
Stop bits	1
	2

- PKW Retries: Number of retransmissions attempted by the gateway when no response, or an erroneous response, is received from a queried Modbus slave using the PKW service. The retries stop as soon as the gateway receives a correct response.
- PKW Timeout time (10ms): Waiting time for a response from a Modbus slave using the PKW service. When the timeout elapses, the gateway starts a number of retries (see previous option).
- Stop bits: 1 or 2 bits.

# 7. Using ABC-LUFP Config Tool

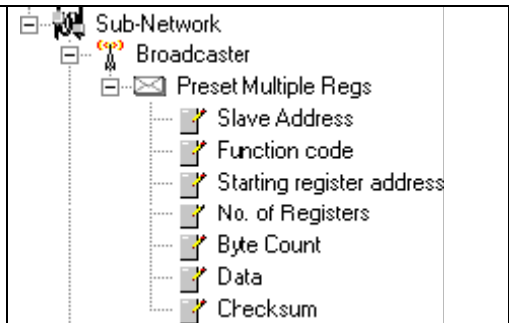
## 7.13.4. Adding a Broadcaster Node

**NOTE:** In the specific case of the LUFP1 gateway, you should use the PKW service in broadcast mode (DN = 255) instead of using a broadcaster node. Thus, you will save precious output data in the limited output memory (from 0x0200 to 0x233), and this data could be used for other Modbus commands. Please refer to section 5.3, for a description of this service.

A broadcaster node does not correspond to any Modbus slave in particular: it applies to **all** Modbus slaves. All the commands configured for this node will be transmitted with the “Slave Address” field set to 0x00. This means that all of the slaves will execute the command, although none of them will respond to it.

To add a broadcaster node, select “Sub-Network”, then choose “Add Broadcaster” from the “Sub-Network” menu. The broadcaster node created in this way does not count in the limit on the number of configurable nodes. A simple example is shown here:

The addition and configuration of a Modbus command in the list of broadcaster node commands is done in the same way as for other nodes, but with the following differences:



- The list of standard Modbus commands which can be used in broadcast is smaller. Only functions 0x06 and 0x10 can be used (see list in chapter 7.11.2).
- The command is made up of a query, but does not include any response. The query bears the name of the command itself, instead of the name “Query”.
- The value of the query frame’s “Slave Address” field is set to 0x00.

Please see chapter 7.12.2.2, for further details on how to configure a Modbus query

# Appendix A: Technical Characteristics

## Environment

Dimensions (excluding connectors)	Height: 120 mm (4.724 in.) Width: 27 mm (1.063 in.) Depth: 75 mm (2.953 in.)
External appearance	Plastic housing with snap-on connection to DIN rail.
Torque	PSU connector: between 5 and 7 lbs.-in.
Power supply	<p>⎓ 24V insulated <math>\pm 10\%</math></p> <p>Maximum consumption: 280 mA (typically around 100 mA)</p> <p>Maximum internal consumption for all of the gateway's electronic cards relating to the internal 5V PSU: 450 mA</p>
Protection class	IP20
Maximum relative humidity	95% without condensation or see page, according to IEC 68-2-30
Ambient air temperature around the device, in a dry environment	<p>According to IEC 68-2-1 Ab, IEC 68-2-2 Bb and IEC 68-2-14 Nb:</p> <ul style="list-style-type: none"> <li>• Storage: <math>-55^{\circ}\text{C} (\pm 3)</math> to <math>+85^{\circ}\text{C} (\pm 2)</math> (<math>-72.4^{\circ}\text{F}</math> to <math>-61.6^{\circ}\text{F}</math>) (<math>181^{\circ}\text{F}</math> to <math>189^{\circ}\text{F}</math>)</li> <li>• Operation: <math>-5^{\circ}\text{C} (\pm 3)</math> to <math>+55^{\circ}\text{C} (\pm 2)</math> (<math>+17.8^{\circ}\text{F}</math> to <math>28.4^{\circ}\text{F}</math>) (<math>127^{\circ}\text{F}</math> to <math>135^{\circ}\text{F}</math>)</li> </ul>
UL	E 214107 certificate "Open type" category The product should be installed in an electrical cabinet or equivalent location.
EC	Certified as complying with European standards, unless otherwise stated.
Electromagnetic compatibility (EMC): Transmission	Complies with industrial environment standard EN 50 081-2:1993. Tested for class A radiation under EN 55011:1990.
Electromagnetic compatibility (EMC): Immunity	Complies with industrial environment standards EN 50 082-2:1995 and EN 61 000-6-2:1999. Tested according to ENV 50 204:1995, EN 61000-4-2:1995, EN 61000-4-3:1996, EN 61000-4-4:1995, EN 61000-4-5:1995 and EN 61000-4-6:1996.

## Communication Characteristics

"Upstream" network	FIPIO
"Downstream" network	Modbus RTU

## Appendix A: Technical Characteristics

---

FIPIO characteristics	<ul style="list-style-type: none"><li>• Physical layer conforming to IEC IS 1158-2.</li><li>• WorldFIP communication profile: Profile 2 (Device WorldFIP / FIPIO).</li><li>• Nature of network: Device-oriented bus (DeviceBus).</li><li>• Network topology: Multipoint linear topology (bus) with adapted and passive line terminations (see Section 2.6.2).</li><li>• Physical media (●●● = 100, 200 or 500 m):<ul style="list-style-type: none"><li>- Main cable: 150 <math>\Omega</math> shielded single twisted-pair copper cable: TSX FP CA●●●</li><li>- Tap link cable: 150 <math>\Omega</math> shielded double twisted-pair copper cable: TSX FP CC●●●</li></ul></li><li>• Connections: 9-pin SUB-D connectors preferred (see Section 2.6.2).</li><li>• Transmission rate: 1 Mbit/s.</li><li>• Maximum length of an electrical segment: 1,000 m.</li><li>• Maximum length of the FIPIO network: 15 km, in 15 segments of 1 km.</li><li>• Maximum number of stations: 32 stations per segment, repeater not included; up to 128 stations for the entire FIPIO network (repeaters not included). Addresses 0 and 63 are reserved for the FIPIO master and the programming terminal respectively.</li><li>• Mono-master network: Managed by a bus arbitrator (FIPIO master); the producer / consumer principle is used for inter-station exchanges.</li><li>• Access methods: Periodic and aperiodic communications handled in bus "macro-cycle"; "time-critical" communication system.</li></ul>
-----------------------	--

## Appendix A: Technical Characteristics

<p>FIPIO characteristics (cont'd)</p>	<ul style="list-style-type: none"> <li>• FIPIO stations are configured according to standard exchange profiles (maximum of 32 words in input and in output for each profile).</li> <li>• Limited station configuration and adjustment possibilities using the FIPIO master.</li> <li>• Standard and customized station diagnostics.</li> <li>• Stations can be connected and disconnected without affecting communications between the other stations.</li> </ul>
<p>FIPIO specifics of LUFFP1 gateway</p>	<ul style="list-style-type: none"> <li>• Communication profile (protocol): FIPIO (WorldFIP profile 2).</li> <li>• Communication profile (exchanges):             <ul style="list-style-type: none"> <li>- FED C32 P ... <i>Default profile</i>; configured by means of 30 configuration parameters and 30 adjustment parameters transmitted to the gateway by the FIPIO master on connection.</li> <li>- FED C32..... "Simplified" profile for the FIPIO master; configured using ABC-LUFFP Config Tool.</li> </ul> </li> </ul> <p>Characteristics common to both profiles: 32 input words (%IW/p.2.c/0.0 to %IW/p.2.c/0.0.31), 32 output words (%QW/p.2.c/0.0 to %QW/p.2.c/0.0.31).</p> <ul style="list-style-type: none"> <li>• Standard connections: male 9-pin SUB-D connector.</li> <li>• Single transmission rate: 1 Mbit/s.</li> <li>• FIPIO station of "slave" type.</li> <li>• FIPIO address configured using 2 coding wheels (address between 1 and 99). In the case of Premium PLCs, addresses 0 and 63 are reserved.</li> <li>• Standard FIPIO diagnostics, but without "input validity".</li> <li>• Available inputs / outputs: 28 input words (%IW/p.2.c/0.0 to %IW/p.2.c/0.0.25) &amp; 28 output words (%QW/p.2.c/0.0 to %QW/p.2.c/0.0.25) for profile FED C32; profile FED C32 P is limited to a sum total of 28 words.</li> <li>• List of active slaves (LAS) service occupies word %IW/p.2.c/0.0.27.</li> <li>• Indexed periodic variables (PKW) service occupies words %IW/p.2.c/0.0.28 to %IW/p.2.c/0.0.31 and %QW/p.2.c/0.0.28 to %QW/p.2.c/0.0.31.</li> </ul>
<p>Modbus RTU characteristics</p>	<ul style="list-style-type: none"> <li>• Physical media: RS485 serial link</li> <li>• Network topology: Multipoint linear topology with adapted line terminations (impedance of 120 <math>\Omega</math> in parallel with a capacity of 1 nF)</li> <li>• Transmission rate: 1,200 to 57,600 kbits/s</li> <li>• Data bits: 8</li> <li>• Subscriber addresses: 1 to 247. Address 0 reserved for broadcasting. Addresses 65, 126 and 127 reserved if drives or starters from <i>Schneider Electric</i> are used on the same Modbus network.</li> <li>• Silence time: Equivalent to the transmission of 3.5 characters.</li> </ul>

# Appendix A: Technical Characteristics

## ⚠ WARNING

### USE OF RESERVED MODBUS ADDRESSES

Do not use Modbus addresses 65, 126, or 127 if a gateway's Modbus slaves will include a Schneider Electric Adjustable-Speed Drive System device such as an Altistart soft-starter or an Altivar motor drive. The Altistart and Altivar devices reserve these addresses for other communications, and the use of these addresses in such a system can have unintended consequences.

**Failure to follow this instruction can result in death, serious injury, or equipment damage.**

Modbus RTU specifics of LUF1 gateway	<ul style="list-style-type: none"> <li>• Maximum number of subscribers (excluding gateway): 8 Modbus slaves.</li> <li>• Maximum number of configurable commands:               <ul style="list-style-type: none"> <li>- FED C32 P ... Max. 28 periodic Modbus commands (read + write), each limited to a single word read or written, which in turn limits the total number of words read and written to 28.</li> <li>- FED C32..... 28 periodic and / or aperiodic Modbus commands, with no limit on the number of words read or written using any one command; all available input / outputs (28 input words and 28 output words) can therefore be used.</li> </ul> </li> </ul>
Modbus RTU specifics of LUF1 gateway	<ul style="list-style-type: none"> <li>• Characteristics configured under PL7 PRO (profile FED C32 P) or under ABC-LUF1 Config Tool (profile FED C32):               <ul style="list-style-type: none"> <li>- Modbus read commands: Periodic mode mandatory (in FED C32 P) or user-configurable mode under ABC-LUF1 Config Tool (in FED C32).</li> <li>- Modbus write commands: Periodic mode with transmission on data change (1) mandatory (in FED C32 P) or user-configurable mode under ABC-LUF1 Config Tool (in FED C32).</li> <li>- Transmission rate: 1200, 2400, 4800, 9600 or 19200 bits/s.</li> <li>- Parity: None, even or odd.</li> <li>- Start bits: 1 bit.</li> <li>- Stop bits: 1 or 2 bits.</li> <li>- Period of silence: Not possible to raise the gateway's period of silence.</li> </ul> </li> </ul>

(1) Mode of communication (Update mode) for which the ABC-LUF1 Config Tool equivalent is called "Change of state or Cyclically". The query for the command thus configured is transmitted periodically, but this periodicity is overridden if any of the data to be sent are modified. This command mode prioritizes write commands, ensuring that they are sent as soon as possible.

# Appendix A: Technical Characteristics

<p>LUFF1 gateway memory structure:</p> <p style="text-align: center;"><b>Inputs</b></p>	<ul style="list-style-type: none"> <li>• 52 bytes accessible to the FIPIO master in the form of input data (see Section 5.1.3, for the mapping between the addresses of these bytes and the gateway's FIPIO objects).</li> <li>• 460 input bytes inaccessible to the FIPIO master due to the use of profile FED C32 / FED C32 P, inputs reserved for the LAS and PKW services, and the presence of a reserved word.</li> </ul> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th style="text-align: left;">Addresses</th> <th style="text-align: center;">Input data area</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">0x0000 : 0x0033</td> <td style="text-align: center;">Inputs accessible to the FIPIO master (52 bytes)</td> </tr> <tr style="background-color: #cccccc;"> <td style="text-align: center;">0x0034 : 0x01FF</td> <td style="text-align: center;">Inputs inaccessible to the FIPIO master (460 bytes)</td> </tr> </tbody> </table>	Addresses	Input data area	0x0000 : 0x0033	Inputs accessible to the FIPIO master (52 bytes)	0x0034 : 0x01FF	Inputs inaccessible to the FIPIO master (460 bytes)
Addresses	Input data area						
0x0000 : 0x0033	Inputs accessible to the FIPIO master (52 bytes)						
0x0034 : 0x01FF	Inputs inaccessible to the FIPIO master (460 bytes)						

<p>LUFF1 gateway memory structure:</p> <p style="text-align: center;"><b>Outputs</b></p>	<ul style="list-style-type: none"> <li>• 52 bytes accessible by the FIPIO master in the form of output data (see Section 5.1.3 for the mapping between the addresses of these bytes and the gateway's FIPIO objects).</li> <li>• 460 output bytes inaccessible to the FIPIO master due to the use of profile FED C32 / FED C32 P, outputs reserved for the LAS and PKW services, and the presence of two reserved words.</li> </ul> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th style="text-align: left;">Addresses</th> <th style="text-align: center;">Output data area</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">0x0200 : 0x0233</td> <td style="text-align: center;">Outputs accessible to the FIPIO master (52 bytes)</td> </tr> <tr style="background-color: #cccccc;"> <td style="text-align: center;">0x0234 : 0x03FF</td> <td style="text-align: center;">Outputs inaccessible to the FIPIO master (460 bytes)</td> </tr> </tbody> </table>	Addresses	Output data area	0x0200 : 0x0233	Outputs accessible to the FIPIO master (52 bytes)	0x0234 : 0x03FF	Outputs inaccessible to the FIPIO master (460 bytes)
Addresses	Output data area						
0x0200 : 0x0233	Outputs accessible to the FIPIO master (52 bytes)						
0x0234 : 0x03FF	Outputs inaccessible to the FIPIO master (460 bytes)						



# Appendix A: Technical Characteristics

LUFF1 gateway memory structure:	<ul style="list-style-type: none"> <li>• 1024 bytes inaccessible to the FIPIO master.</li> </ul>																	
General data	<table border="1" data-bbox="605 275 1430 695"> <thead> <tr> <th data-bbox="605 275 753 304">Addresses</th> <th data-bbox="980 275 1203 304">General data area</th> </tr> </thead> <tbody> <tr> <td data-bbox="605 310 753 340">0x0400</td> <td data-bbox="753 310 1430 380" rowspan="2">Input area reserved for the Mailboxes (288 bytes)</td> </tr> <tr> <td data-bbox="605 346 753 375">0x051F</td> </tr> <tr> <td data-bbox="605 384 753 413">0x0520</td> <td data-bbox="753 384 1430 453" rowspan="2">Output area reserved for the Mailboxes (288 bytes)</td> </tr> <tr> <td data-bbox="605 417 753 447">0x063F</td> </tr> <tr> <td data-bbox="605 455 753 485">0x0640</td> <td data-bbox="753 455 1430 552" rowspan="2">Internal area reserved for the management of the upstream network (384 bytes; area not used by the LUFF1 gateway)</td> </tr> <tr> <td data-bbox="605 491 753 520">0x07BF</td> </tr> <tr> <td data-bbox="605 554 753 583">0x07C0</td> <td data-bbox="753 554 1430 623" rowspan="2">Internal area reserved for the control registers (62 bytes / MSB first for 16-bit data)</td> </tr> <tr> <td data-bbox="605 588 753 617">0x07FD</td> </tr> <tr> <td data-bbox="605 625 753 655">0x07FE</td> <td data-bbox="753 625 1430 695" rowspan="2">Gateway status / FIPIO master command (2 reserved bytes on the LUFF1 gateway)</td> </tr> <tr> <td data-bbox="605 661 753 690">0x07FF</td> </tr> </tbody> </table> <p data-bbox="558 726 1500 932"><b>NOTE:</b> You can use this data area to store data from a Modbus response that you don't want to pass on to the FIPIO master. You can also use this area for data transfers between commands as this area is both an input and an output area. In this case, <i>always use 0x0400 as the starting address</i>. If you use the same addresses more than once in this area, the corresponding locations will be displayed in red in the "General Area" pane of the "Sub-network Monitor" window, but it will not affect the operation of the gateway.</p>	Addresses	General data area	0x0400	Input area reserved for the Mailboxes (288 bytes)	0x051F	0x0520	Output area reserved for the Mailboxes (288 bytes)	0x063F	0x0640	Internal area reserved for the management of the upstream network (384 bytes; area not used by the LUFF1 gateway)	0x07BF	0x07C0	Internal area reserved for the control registers (62 bytes / MSB first for 16-bit data)	0x07FD	0x07FE	Gateway status / FIPIO master command (2 reserved bytes on the LUFF1 gateway)	0x07FF
Addresses	General data area																	
0x0400	Input area reserved for the Mailboxes (288 bytes)																	
0x051F																		
0x0520	Output area reserved for the Mailboxes (288 bytes)																	
0x063F																		
0x0640	Internal area reserved for the management of the upstream network (384 bytes; area not used by the LUFF1 gateway)																	
0x07BF																		
0x07C0	Internal area reserved for the control registers (62 bytes / MSB first for 16-bit data)																	
0x07FD																		
0x07FE	Gateway status / FIPIO master command (2 reserved bytes on the LUFF1 gateway)																	
0x07FF																		
Data transfer order ( <i>swapping</i> )	<ul style="list-style-type: none"> <li>• FIPIO network: LSB first and MSB last.</li> <li>• LUFF1 gateway: MSB stored in the lowest memory address.</li> <li>• Modbus RTU network: MSB first and LSB last.</li> </ul> <p data-bbox="540 1058 1500 1136">→ The option which should be selected for Modbus data stored in the gateway's memory is "No swapping". This option relates to all "Data" and "Preset data" fields in the Modbus query and response frames.</p>																	

## Appendix B: LUPF1 Gateway Settings

The settings described here correspond to the configuration and adjustment parameters for the LUPF1 gateway. **They only exist when the gateway is used according to standard profile FED C32 P.** The exchange modes for the corresponding FIPIO objects are described in Section 5.6. Once they have been downloaded to the gateway, the values of these parameters can be read and / or replaced using the gateway's PKW service (see Section 5.3.5).

**NOTE:** Some of the configuration and adjustment parameters cannot be modified after the initial setup of the gateway using the PLC. See the Configuration Parameters and Adjustment Parameters tables in Chapter 4.2.

### Configuration Parameters

The configuration parameters enable the gateway to configure the Modbus commands that it exchanges with the Modbus slaves. The configuration comprises:

- 1) The number of periodic write commands (periodic command words) for each slave (0 to 15);
- 2) The number of periodic read commands (periodic control words) for each slave (0 to 15);
- 3) The address of each periodic command or control word.

**The total number of periodic words (command words + control words) is limited to 26.**

Each **periodic command word** corresponds to a Modbus write command to a single register (function 0x06 "Preset Single Register"). This command is cyclical, with advance transmission of the write query if the new value for the register is modified. This exchange mode is equivalent to "Update mode = Change of state or cyclically".

Each **periodic control word** corresponds to a Modbus read command for several registers (function 0x03 "Read Holding Registers"), even though the command is used to read a single register. This command is strictly cyclical. This exchange mode is equivalent to "Update mode = Cyclically".

The configuration parameter values for the standard 8 TeSys U motor starter configuration are listed in Section 4.2.4.

PL7 Parameter	Access	Values	Description
%KWp.2.cI0.0.0	RO	0x0000 to 0xFFFF	Number of periodic command words for slaves n°1 to n°4: Bits 0- 3: Number of periodic command words for slave n°1 = 0 to 15 Bits 4- 7: Number of periodic command words for slave n°2 = 0 to 15 Bits 8-11: Number of periodic command words for slave n°3 = 0 to 15 Bits 12-15: Number of periodic command words for slave n°4 = 0 to 15
%KWp.2.cI0.0.1	RO	0x0000 to 0xFFFF	Number of periodic command words for slaves n°5 to n°8: Bits 0- 3: Number of periodic command words for slave n°5 = 0 to 15 Bits 4- 7: Number of periodic command words for slave n°6 = 0 to 15 Bits 8-11: Number of periodic command words for slave n°7 = 0 to 15 Bits 12-15: Number of periodic command words for slave n°8 = 0 to 15
%KWp.2.cI0.0.2	RO	0x0000 to 0xFFFF	Number of periodic control words for slaves n°1 to n°4: Bits 0- 3: Number of periodic control words for slave n°1 = 0 to 15 Bits 4- 7: Number of periodic control words for slave n°2 = 0 to 15 Bits 8-11: Number of periodic control words for slave n°3 = 0 to 15 Bits 12-15: Number of periodic control words for slave n°4 = 0 to 15
%KWp.2.cI0.0.3	RO	0x0000 to 0xFFFF	Number of periodic control words for slaves n°5 to n°8: Bits 0- 3: Number of periodic control words for slave n°5 = 0 to 15 Bits 4- 7: Number of periodic control words for slave n°6 = 0 to 15 Bits 8-11: Number of periodic control words for slave n°7 = 0 to 15 Bits 12-15: Number of periodic control words for slave n°8 = 0 to 15

# Appendix B: LUF P1 Gateway Settings

PL7 Parameter	Access	Values	Description	PL7 Parameter	Access	Values	Description
%KWlp.2.c\0.0.4	R/W	0x****	Address of periodic word n° 1	%KWlp.2.c\0.0.17	R/W	0x****	Address of periodic word n°14
%KWlp.2.c\0.0.5	R/W	0x****	Address of periodic word n° 2	%KWlp.2.c\0.0.18	R/W	0x****	Address of periodic word n°15
%KWlp.2.c\0.0.6	R/W	0x****	Address of periodic word n° 3	%KWlp.2.c\0.0.19	R/W	0x****	Address of periodic word n°16
%KWlp.2.c\0.0.7	R/W	0x****	Address of periodic word n° 4	%KWlp.2.c\0.0.20	R/W	0x****	Address of periodic word n°17
%KWlp.2.c\0.0.8	R/W	0x****	Address of periodic word n° 5	%KWlp.2.c\0.0.21	R/W	0x****	Address of periodic word n°18
%KWlp.2.c\0.0.9	R/W	0x****	Address of periodic word n° 6	%KWlp.2.c\0.0.22	R/W	0x****	Address of periodic word n°19
%KWlp.2.c\0.0.10	R/W	0x****	Address of periodic word n° 7	%KWlp.2.c\0.0.23	R/W	0x****	Address of periodic word n°20
%KWlp.2.c\0.0.11	R/W	0x****	Address of periodic word n° 8	%KWlp.2.c\0.0.24	R/W	0x****	Address of periodic word n°21
%KWlp.2.c\0.0.12	R/W	0x****	Address of periodic word n° 9	%KWlp.2.c\0.0.25	R/W	0x****	Address of periodic word n°22
%KWlp.2.c\0.0.13	R/W	0x****	Address of periodic word n°10	%KWlp.2.c\0.0.26	R/W	0x****	Address of periodic word n°23
%KWlp.2.c\0.0.14	R/W	0x****	Address of periodic word n°11	%KWlp.2.c\0.0.27	R/W	0x****	Address of periodic word n°24
%KWlp.2.c\0.0.15	R/W	0x****	Address of periodic word n°12	%KWlp.2.c\0.0.28	R/W	0x****	Address of periodic word n°25
%KWlp.2.c\0.0.16	R/W	0x****	Address of periodic word n°13	%KWlp.2.c\0.0.29	R/W	0x****	Address of periodic word n°26

In the above table, the mapping between the addresses of periodic words n°1 to 26 and the command and control words of the various Modbus slaves is not given, because it depends *entirely* on the numbers of periodic command and control words for the Modbus slaves as a whole. These periodic words are organized as follows:

Address of periodic word n° 1	Addresses of the periodic command words for slave n°1
⋮	⋮
⋮	Addresses of the periodic command words for slave n°8
⋮	Addresses of the periodic control words for slave n°1
⋮	⋮
⋮	Addresses of the periodic control words for slave n°8
⋮	0x0000 (1)
⋮	⋮
Address of periodic word n°26	0x0000 (1)

(1) If you do not use all of the 26 periodic words available, you must reset the addresses of the unused periodic words.

## Adjustment Parameters

The adjustment parameters enable the gateway to configure the general features of its exchanges with the Modbus slaves.

The adjustment parameters values for the standard 8 TeSys U motor starter configuration are listed in Section 4.2.5.

## Appendix B: LUF1 Gateway Settings

PL7 parameter	Access	Bits	Description	Possible values or range of values
%MWip.2.c/0.0.20	RO	0- 7	Modbus speed	0x00 = 1200 bits/s    0x03 = 9600 bits/s 0x01 = 2400 bits/s    0x04 = 19200 bits/s 0x02 = 4800 bits/s
		8	Modbus format	Data bits 0 = 7 bits (Modbus ASCII) → PROHIBITED 1 = 8 bits (Modbus RTU) → MANDATORY
		9		Stop bits 0 = 1 stop bit 1 = 2 stop bits
		10-11		Parity Bit 11 = 0 and bit 10 = 0: No parity Bit 11 = 0 and bit 10 = 1: Even parity Bit 11 = 1 and bit 10 = 0: Odd parity
		12-15	Not used	2#0000
%MWip.2.c/0.0.21	RO	0- 7	Modbus Timeout: Max. waiting period for a response from a Modbus slave. <b>NOTE: Allow for the slave with the longest response time.</b>	1 to 255 (unit: 100 ms) = 100 ms to 25.5 s
		8-15	Number of retransmissions of a frame if no response from a Modbus slave. The slave is declared absent if these retransmissions are completed without a correct response being received from it.	0 = No retransmission 1 to 15 = Number of consecutive retransmissions of the same Modbus frame by the gateway
%MWip.2.c/0.0.22	RO	0- 7	Reconnection time for a Modbus slave declared absent (see Number of retransmissions, above). The gateway stops communicating with the slave for the set duration.	1 to 255 (unit: 1 s) = 1 s to 255 s
		8-15	Not used	0x00
%MWip.2.c/0.0.23	RO	0- 7	Effect on Modbus exchanges of disconnection of the FIPIO network, or non-refreshment of FIPIO output data.	0x00 = "Freeze" = Modbus exchanges continue normally; command words keep their current values. 0x01 = "NoScanning" = Stop all Modbus exchanges. 0x02 = "Clear" = Modbus exchanges continue, but command word values are reset to zero.
		8-15	Effect on FIPIO exchanges of disconnection of the Modbus network, or non-refreshment of Input data.	0x00 = "No action" = Inputs keep their current values. 0x02 = "Clear" = Input values are reset to zero.
%MWip.2.c/0.0.24	RO	0-15	Not used	0x0000
%MWip.2.c/0.0.25	R/W	0- 7	Periodicity of Modbus exchanges	Fast commands
%MWip.2.c/0.0.26	R/W	8-15		Normal commands
		0- 7		Fast controls
		8-15		Normal controls
%MWip.2.c/0.0.27	R/W	0	Cycle time of periodic Modbus commands	Periodic word n° 1
		⋮		Periodic word n° 8
		7		Periodic word n° 9
		8		Periodic word n°16
%MWip.2.c/0.0.28	R/W	⋮		Periodic word n°17
		7		Periodic word n°24
		8		Periodic word n°25
		9		Periodic word n°26
		10-15	Not used	2#000000

## Appendix B: LUF1 Gateway Settings

PL7 parameter	Access	Bits	Description	Possible values or range of values	
%MW\p.2.c\0.0.29	R/W	0 : 7	Cycle time of periodic	Periodic word n° 1 ..... Periodic word n° 8	0 = Fast control  1 = Normal control
		8 : 15		Periodic word n° 9 ..... Periodic word n°16	
%MW\p.2.c\0.0.30	R/W	0 : 7	Modbus controls	Periodic word n°17 ..... Periodic word n°24	2#000000
		8 9		Periodic word n°25 Periodic word n°26	
		10-15		Not used	
%MW\p.2.c\0.0.31	RO	0- 7 8-15	Addresses of Modbus slaves	Modbus slave n°1	0 = No Modbus slave at this location  1-247 = Address of Modbus slave at this location (no "empty locations" are allowed between 2 slaves)
%MW\p.2.c\0.0.32	RO	0- 7 8-15		Modbus slave n°2	
		%MW\p.2.c\0.0.33		RO	
%MW\p.2.c\0.0.34	RO				
			Modbus slave n°6		
			Modbus slave n°7		
			Modbus slave n°8		
%MW\p.2.c\0.0.35 ..... %MW\p.2.c\0.0.49	RO	0-15	Not used		0x0000

**Periodicity of Modbus exchanges and Cycle time of periodic Modbus commands / controls:** The parameters %MW\p.2.c\0.0.25 to %MW\p.2.c\0.0.30 set the cycle time of the Modbus commands corresponding to the periodic command and control words defined by the configuration parameters:

- The “periodicity of Modbus exchanges” parameters (%MW\p.2.c\0.0.25 and %MW\p.2.c\0.0.26) set fast or slow periodicities for the periodic command and control words.
- Each of the bits in the “cycle time of periodic Modbus commands” (%MW\p.2.c\0.0.27 and %MW\p.2.c\0.0.28) determines either fast periodicity (bit at 0) or slow periodicity (bit at 1) for the commands, both defined by the parameter %MW\p.2.c\0.0.25.
- Each of the bits in the “cycle time of periodic Modbus controls” parameters (%MW\p.2.c\0.0.29 and %MW\p.2.c\0.0.30) determines either fast periodicity (bit at 0) or slow periodicity (bit at 1) for the controls, both defined by the parameter %MW\p.2.c\0.0.26.

**Addresses of Modbus slaves:** Remember that you *must* define the Modbus slaves starting at Modbus slave n°1 and then on in sequence, without leaving any “empty slots”.

# Appendix B: LUF1 Gateway Settings

## Management of Degraded Modes

The management of degraded modes on the LUF1 gateway, described in Section 4.1.4, now depends on two gateway adjustment parameters. The table below is the same as the one in that section, except that the two new parameters have been included:

Desired behavior		Event			
		Premium PLC: CPU stop or failure	Disconnection of the upstream FIPIO network (1)	Failure of the LUF1 gateway	Disconnection of the downstream Modbus RTU network
Outputs	Reset	%MWp.2.c\0.0.23:X0-X7 = 0x02		Depending on the configuration of the Modbus slaves (2)	
	Hold	%MWp.2.c\0.0.23:X0-X7 = 0x00			
	No refresh	%MWp.2.c\0.0.23:X0-X7 = 0x01			
Inputs	Reset	—	—	%MWp.2.c\0.0.23:8-X15 = 0x02 (3)	
	Hold	—	Yes (4)	%MWp.2.c\0.0.23:X8-X15 = 0x00 (3)	

- (1) See previous page for a description of the adjustment parameter used in this column.
- (2) The desired behavior with regard to the outputs should be directly configured on each of the Modbus slaves. In the case of drives marketed by Schneider Electric, for instance, the outputs are reset by setting the NTO bit to 0 (command with communication control), and held by setting NTO to 1 (command Without communication control).
- (3) As regards inputs, we strongly recommend that you use the List of Active Slaves (LAS) Service (see Section 5.2, page 48) to detect the underlying event.
- (4) Use the FIPIO Diagnostic Objects (Section 5.4, page 58) to detect the underlying event.

### PLC processor stopped or on failure

#### PLC processor response

##### Outputs:

Software error, outputs reset to default state or hold their present state depending on configuration.  
Hardware error (EEPROM or hardware failure), output state will be undetermined

Inputs: PLC stops responding to inputs in any error state.

#### FIPIO master response

Depending on the configuration of the FIPIO master:

the FIPIO master stops communicating with the LUF1 gateway, **or**  
the FIPIO master forces outputs to 0 and refreshes the inputs, **or**  
the FIPIO master holds outputs in their last state and refreshes the inputs.

#### LUF1 gateway response

If the FIPIO master stops communicating with the LUF1 gateway (in the case of a Premium PLC), for each Modbus Command, the behavior depends on bits 0-7 of the %MWp.2.c\0.0.23 parameter:

- 0x00 (Freeze) The output data sent to the Modbus slave concerned retains its current value and the input memory continues to be refreshed with the data of the slaves' response.
- 0x01 (No Scanning) The query of this Modbus Command is no longer transmitted and the input memory is no longer refreshed with the data of the slaves' response.
- 0x02 (Clear) The output data sent to the Modbus slave concerned is reset to 0 and the input memory continues to be refreshed with the data of the slaves' response.

If the FIPIO master forces outputs to 0 and refreshes the inputs, the periodic Modbus exchanges continue to run:  
all output data are reset to 0 and sent to the Modbus slaves,

## Appendix B: LUFP1 Gateway Settings

the input memory continues to be refreshed with the data of the Modbus slaves' responses, If the FIPIO master forces holds outputs in their last state and refreshes the inputs, the periodic Modbus exchanges continues to run: all output data stay in their last state and are sent to the Modbus slaves, the input memory continues to be refreshed with the data of the Modbus slaves' responses
<b>TeSys U response</b>
Depending on the slave.

### FIPIO scanner stopped or on failure

<b>PLC processor response</b>
The PLC processor provides some error and/or diagnostic objects to the application if FIPIO master stop or failure (input/output not valid) occurs. Refer to the PLC user manual for their description. This information must be managed in the PLC application.
<b>FIPIO master response</b>
If the FIPIO master is stopped (command coming from the application): the scanner stops communicating with the LUFP1 gateway.
If the FIPIO master is on failure: the scanner stops communicating with the processor and the LUFP1 gateway.
<b>LUFP1 gateway response</b>
For all Modbus Commands, the behavior depends on bits 0-7 of the %MWp.2.c\0.0.23 parameter: 0x00 (Freeze)      The output data sent to the Modbus slave concerned retains its current value and the input memory continues to be refreshed with the data of the slaves' response. 0x01 (No Scanning) The query of this Modbus Command is no longer transmitted and the input memory is no longer refreshed with the data of the slaves' response. 0x02 (Clear)      The output data sent to the Modbus slave concerned is reset to 0 and the input memory continues to be refreshed with the data of the slaves' response.
<b>TeSys U response</b>
Depending on the slave.

### LUFP1 gateways disconnected on FIPIO side

<b>PLC response</b>
The PLC processor provides some error and/or diagnostic objects to the application if slave disconnection from the application occurs. Refer to the PLC user manual for their description. This information must be managed in the PLC application.
<b>FIPIO master response</b>
The FIPIO master provides the processor with some error and diagnostic objects if FIPIO slave disconnection occurs.
<b>LUFP1 gateway response</b>
For all Modbus Commands, the behavior depends on bits 0-7 of the %MWp.2.c\0.0.23 parameter: 0x00 (Freeze)      The output data sent to the Modbus slave concerned retains its current value and the input memory continues to be refreshed with the data of the slaves' response. 0x01 (No Scanning) The query of this Modbus Command is no longer transmitted and the input memory is no longer refreshed with the data of the slaves' response. 0x02 (Clear)      The output data sent to the Modbus slave concerned is reset to 0 and the input memory continues to be refreshed with the data of the slaves' response.
<b>TeSys U response</b>
Depending on the slave.

## Appendix B: LUF1 Gateway Settings

---

### LUF1 gateways failure

<b>PLC response</b>
The PLC processor provides some error and/or diagnostic objects to the application if slave failure to the application occurs. Refer to the PLC user manual for their description. This information must be managed in the PLC application.
<b>FIPIO master response</b>
The FIPIO master provides the processor with some error and diagnostic objects if FIPIO slave failure occurs.
<b>LUF1 gateway response</b>
In case of a failure, the gateway stops communicating with the FIPIO master and the Modbus slaves.
<b>TeSys U response</b>
Depending on the slave.

### LUF1 gateways disconnected on Modbus side or slave failure

<b>PLC response</b>
The processor has access to the List of Active Slaves (LAS) coming from the FIPIO master input table. This word must be managed in the PLC application in order to detect if a Modbus slave is missing.
<b>FIPIO master response</b>
The FIPIO master continues communicating with the PLC processor and the LUF1 gateway.
<b>LUF1 gateway response</b>
Depending on the following properties, all of which are configurable for each Modbus Command: %MW\p.2.c\0.0.21:X0-X7 (Modbus Timeout) = $x \times 100$ ms, %MW\p.2.c\0.0.21:X8-X15 (Number of retransmissions) = $y$ , %MW\p.2.c\0.0.22:X0-X7 (Reconnection time) = $z \times 1$ sec, <i>and</i> %MW\p.2.c\0.0.23:X8-X15 (Effect on FIPIO exchanges) = 0x00 (No action) / 0x02 (Clear). After sending a request to a slave, if there is no response after ( $x \times 100$ ) ms, the gateway will send it again $y$ times before writing information about the missing slave to the List of Active Slave (LAS), if no other Modbus Command has succeeded for this slave. If, despite the $y$ Retries, there is no answer from the slave, the input Data sent to the FIPIO master (Read requests) is reset to 0 (if 0x02/Clear) or retains its current value (if 0x00/No action), but only for the data received from the missing slave. The gateway will try to reconnect the missing slave by communicating with it again every ( $z \times 1$ ) ms.
<b>Slave response</b>
If the LUF1 gateway is disconnected on Modbus side, the behavior depends on the slave.
In the event of a slave failure No response is sent to the gateway and the state of the slave is undetermined. This case <b>must</b> be managed in the PLC application.

### Sample Gateway Settings

The example below illustrates a system architecture and Modbus exchanges that differ from those in the standard configuration described in the rest of this manual.

The example uses the configuration and adjustment parameters described in the previous sections.



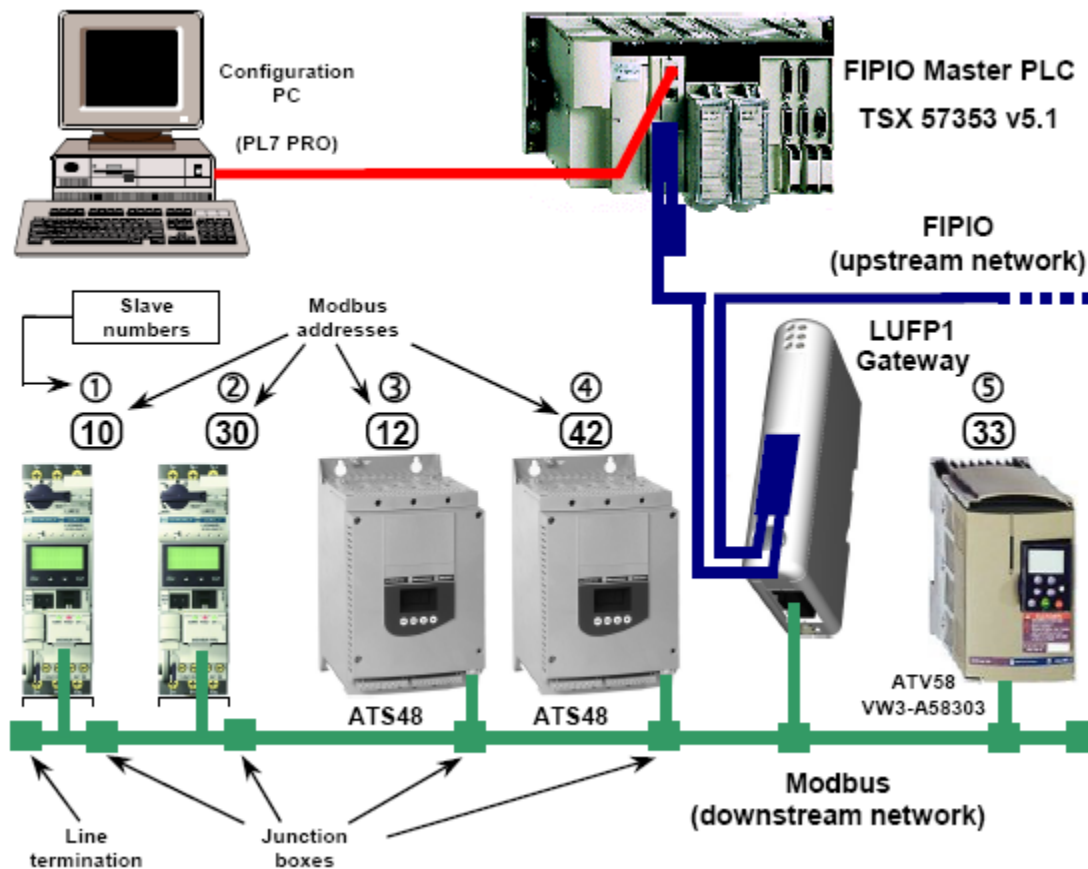
# Appendix B: LUF1 Gateway Settings

## System Architecture

The Modbus slaves are grouped by category, rather than in order of address, in the list of Modbus slaves on the UFP1 gateway. The slaves, as seen by the gateway, are as follows:

Slave	Product type	Product name	Modbus address
N°1	Motor starter	TeSys U	10
N°2		TeSys U	30
N°3	Soft start	Altistart 48	12
N°4		Altistart 48	42
N°5	Speed drive	Altivar 58 + Option VW3-A58303	33
N°6			0 (1)
N°7			0 (1)
N°8			0 (1)

- (1) In the gateway settings, assigning a null address to a slave means that there is no slave at that location. Not to be confused with the broadcasting of Modbus commands.



# Appendix B: LUF P1 Gateway Settings

## Modbus Exchanges

The table below lists all the registers read or written periodically by the gateway on the Modbus slaves. It will subsequently be used to configure the gateway's periodic command and control words.

Product	Exchange	Address	Register addressed	Periodicity
TeSys U (x2)	Command	704	Motor starter command	Fast
		700	Cassette command	Normal
	Control	455	Motor starter status	Fast
		452	Faults present	Normal
		461	Alarms present	Normal
ATS48 (x2)	Command	400	Command register	Fast
	Control	458	Status register	Fast
		4 072	Active power	Normal
		4 063	Torque	Normal
ATV58 (x1)	Command	400	DRIVECOM command register	Fast
		401	Online frequency setpoint (complement of 2)	Fast
		402	Internal command register (applications)	Normal
	Control	458	DRIVECOM status register	Fast
		451	Output frequency applied to motor (absolute value)	Normal
		452	Motor speed estimated by drive (absolute value)	Normal
		453	Motor current	Normal

Number of periodic command words = 2 (x2) + 1 (x2) + 3 (x1) = 4 + 2 + 3 = 9

Number of periodic control words = 3 (x2) + 3 (x2) + 4 (x1) = 6 + 6 + 4 = 16

Total number of periodic words = 9 + 16 = 25

Reserve: 1 periodic command or control word (maximum of 26 periodic words).

The periodic words are assigned firstly to the periodic command words, and then to the periodic control words. Each category starts with the first Modbus slave and ends with the last configured slave.

In the present example, this gives us the following configuration of periodic words; a grey background indicates a periodic word with a "fast" periodicity (otherwise the periodicity is "normal"):

Periodic word	Modbus slave	Register
1 Command n°1	N°1 TeSys U 10	704
2 Command n°2	N°1 TeSys U 10	700
3 Command n°3	N°2 TeSys U 30	704
4 Command n°4	N°2 TeSys U 30	700
5 Command n°5	N°3 ATS48 12	400
6 Command n°6	N°4 ATS48 42	400
7 Command n°7	N°5 ATV58 33	400
8 Command n°8	N°5 ATV58 33	401
9 Command n°9	N°5 ATV58 33	402
10 Control n° 1	N°1 TeSys U 10	455
11 Control n° 2	N°1 TeSys U 10	452
12 Control n° 3	N°1 TeSys U 10	461
13 Control n° 4	N°2 TeSys U 30	455

Periodic word	Modbus slave	Register
14 Control n° 5	N°2 TeSys U 30	452
15 Control n° 6	N°2 TeSys U 30	461
16 Control n° 7	N°3 ATS48 12	458
17 Control n° 8	N°3 ATS48 12	4 072
18 Control n° 9	N°3 ATS48 12	4 063
19 Control n°10	N°4 ATS48 42	458
20 Control n°11	N°4 ATS48 42	4 072
21 Control n°12	N°4 ATS48 42	4 063
22 Control n°13	N°5 ATV58 33	458
23 Control n°14	N°5 ATV58 33	451
24 Control n°15	N°5 ATV58 33	452
25 Control n°16	N°5 ATV58 33	453
26		

## Appendix B: LUFP1 Gateway Settings

---

- Modbus speed ..... 19,200 bits/s
- Modbus format..... 8 data bits (Modbus RTU) — 1 stop bit — no parity
- Degraded modes ..... Timeout of 300 ms — 2 retransmissions — Reconnect after 30 s — Reset command values if FIPIO bus is disconnected — Reset the control values of a Modbus slave if this slave is disconnected.
- Periodicities ..... Fast commands at 300 ms / normal at 900 ms  
Fast controls at 300 ms / normal at 1,800 ms



### WARNING

#### PLANNING OF THE MODBUS EXCHANGES

When you create a new configuration and you want to set up the gateway to handle the Modbus exchanges for this configuration, it is highly advisable to plot out all such exchanges in advance, as illustrated above.

**Failure to follow this instruction can result in death, serious injury, or equipment damage.**

**NOTE:** The configuration of the Modbus slaves themselves is beyond the scope of this guide, and is therefore not described here. See the user manuals of the corresponding products.

# Appendix B: LUF1 Gateway Settings

## Configuration Parameter Values

PL7 parameter	Value	Description
%KWp.2.c\0.0.0	4 386 0x1122	Number of periodic command words for slaves n°1 to n°4: Bits 0- 3: Nbr of periodic command words for slave n°1 (TeSys U) = 2 Bits 4- 7: Nbr of periodic command words for slave n°2 (TeSys U) = 2 Bits 8-11: Nbr of periodic command words for slave n°3 (ATS48) = 1 Bits 12-15: Nbr of periodic command words for slave n°4 (ATS48) = 1
%KWp.2.c\0.0.1	3 0x0003	Number of periodic command words for slaves n°5 to n°8: Bits 0- 3: Nbr of periodic command words for slave n°5 (ATV58) = 3 Bits 4- 7: Nbr of periodic command words for slave n°6 = 0 Bits 8-11: Nbr of periodic command words for slave n°7 = 0 Bits 12-15: Nbr of periodic command words for slave n°8 = 0
%KWp.2.c\0.0.2	13 107 0x3333	Number of periodic control words for slaves n°1 to n°4: Bits 0- 3: Nbr of periodic control words for slave n°1 (TeSys U) = 3 Bits 4- 7: Nbr of periodic control words for slave n°2 (TeSys U) = 3 Bits 8-11: Nbr of periodic control words for slave n°3 (ATS48) = 3 Bits 12-15: Nbr of periodic control words for slave n°4 (ATS48) = 3
%KWp.2.c\0.0.3	4 0x0004	Number of periodic control words for slaves n°5 to n°8: Bits 0- 3: Nbr of periodic control words for slave n°5 (ATV58) = 4 Bits 4- 7: Nbr of periodic control words for slave n°6 = 0 Bits 8-11: Nbr of periodic control words for slave n°7 = 0 Bits 12-15: Nbr of periodic control words for slave n°8 = 0
%KWp.2.c\0.0.4	704 0x02C0	Address of periodic word n°1: 1st prd. command word of slave n°1 (TeSys U)
%KWp.2.c\0.0.5	700 0x02BC	Address of periodic word n°2: 2nd prd. command word of slave n°1 (TeSys U)
%KWp.2.c\0.0.6	704 0x02C0	Address of periodic word n°3: 1st prd. command word of slave n°2 (TeSys U)
%KWp.2.c\0.0.7	700 0x02BC	Address of periodic word n°4: 2nd prd. command word of slave n°2 (TeSys U)
%KWp.2.c\0.0.8	400 0x0190	Address of periodic word n°5: 1st prd. command word of slave n°3 (ATS48)
%KWp.2.c\0.0.9	400 0x0190	Address of periodic word n°6: 1st prd. command word of slave n°4 (ATS48)
%KWp.2.c\0.0.10	400 0x0190	Address of periodic word n°7: 1st prd. command word of slave n°5 (ATV58)
%KWp.2.c\0.0.11	401 0x0191	Address of periodic word n°8: 2nd prd. command word of slave n°5 (ATV58)
%KWp.2.c\0.0.12	402 0x0192	Address of periodic word n°9: 3rd prd. command word of slave n°5 (ATV58)
%KWp.2.c\0.0.13	455 0x01C7	Address of periodic word n°10: 1st prd. control word of slave n°1 (TeSys U)
%KWp.2.c\0.0.14	452 0x01C4	Address of periodic word n°11: 2nd prd. control word of slave n°1 (TeSys U)
%KWp.2.c\0.0.15	461 0x01CD	Address of periodic word n°12: 3rd prd. control word of slave n°1 (TeSys U)
%KWp.2.c\0.0.16	455 0x01C7	Address of periodic word n°13: 1st prd. control word of slave n°2 (TeSys U)
%KWp.2.c\0.0.17	452 0x01C4	Address of periodic word n°14: 2nd prd. control word of slave n°2 (TeSys U)
%KWp.2.c\0.0.18	461 0x01CD	Address of periodic word n°15: 3rd prd. control word of slave n°2 (TeSys U)
%KWp.2.c\0.0.19	458 0x01CA	Address of periodic word n°16: 1st prd. control word of slave n°3 (ATS48)
%KWp.2.c\0.0.20	4 072 0x0FE8	Address of periodic word n°17: 2nd prd. control word of slave n°3 (ATS48)
%KWp.2.c\0.0.21	4 063 0x0FDF	Address of periodic word n°18: 3rd prd. control word of slave n°3 (ATS48)
%KWp.2.c\0.0.22	458 0x01CA	Address of periodic word n°19: 1st prd. control word of slave n°4 (ATS48)
%KWp.2.c\0.0.23	4 072 0x0FE8	Address of periodic word n°20: 2nd prd. control word of slave n°4 (ATS48)
%KWp.2.c\0.0.24	4 063 0x0FDF	Address of periodic word n°21: 3rd prd. control word of slave n°4 (ATS48)
%KWp.2.c\0.0.25	458 0x01CA	Address of periodic word n°22: 1st prd. control word of slave n°5 (ATV58)
%KWp.2.c\0.0.26	451 0x01C3	Address of periodic word n°23: 2nd prd. control word of slave n°5 (ATV58)
%KWp.2.c\0.0.27	452 0x01C4	Address of periodic word n°24: 3rd prd. control word of slave n°5 (ATV58)
%KWp.2.c\0.0.28	453 0x01C5	Address of periodic word n°25: 4th prd. control word of slave n°5 (ATV58)
%KWp.2.c\0.0.29	0 0x0000	Address of periodic word n°26: Not used (reserved)

# Appendix B: LUFPP1 Gateway Settings

## Adjustment Parameter Values

Parameter	Value	Description
%MWlp.2.c\0.0.20	260 0x0104	Bits 0- 7: Modbus speed = 10,200 bits/s (LSB byte = 4) Bit 8: Modbus format = 8 data bits (bit at 1) Bit 9: Modbus format = 1 stop bit (bit at 0) Bits 10-11: Modbus format = No parity (bits at 0) Bits 12-15: Not used (bits at 0)
%MWlp.2.c\0.0.21	515 0x0203	Bits 0- 7: Modbus response timeout = 300 ms (3 x 100 ms) Bits 8-15: Number of frame retransmissions over the Modbus network = 2
%MWlp.2.c\0.0.22	30 0x001E	Bits 0- 7: Time taken by gateway to reconnect Modbus slave following a break in communication = 30 s (30 x 1 s) Bits 8-15: Not used (bits at 0)
%MWlp.2.c\0.0.23	514 0x0202	Bits 0- 7: Behavior on losing FIPIO network = Reset periodic command word values transmitted to Modbus slaves (LSB byte = 2) Bits 8-15: Behavior on losing Modbus network = Reset periodic control word values transmitted by Modbus slaves (LSB byte = 2)
%MWlp.2.c\0.0.24	0 0x0000	Not used (bits at 0)
%MWlp.2.c\0.0.25	23 070 0x5A1E	Bits 0- 7: Cycle time of fast periodic commands = 300 ms (30 x 10 ms) Bits 8-15: Cycle time of normal periodic commands = 900 ms (90 x 10 ms)
%MWlp.2.c\0.0.26	46 110 0xB41E	Bits 0- 7: Cycle time of fast periodic controls = 300 ms (30 x 10 ms) Bits 8-15: Cycle time of normal periodic controls = 1,800 ms (180 x 10 ms)
%MWlp.2.c\0.0.27	266 2#0000 0001 0000 1010	Cycle time settings for periodic commands n°1 to 16: Bit 0: Prd. command n°1 = fast (0)      Bit 5: Prd. command n°6 = fast (0) Bit 1: Prd. command n°2 = normal (1)    Bit 6: Prd. command n°7 = fast (0) Bit 2: Prd. command n°3 = fast (0)      Bit 7: Prd. command n°8 = fast (0) Bit 3: Prd. command n°4 = normal (1)    Bit 8: Prd. command n°9 = normal (1) Bit 4: Prd. command n°5 = fast (0)      Bits 9 to 15: Not used (bits at 0)
%MWlp.2.c\0.0.28	0 0x0000	Cycle time settings for periodic commands n°17 to 26: Not used
%MWlp.2.c\0.0.29	60 854 2#1110 1101 1011 0110	Cycle time settings for periodic controls n°1 to 16: Bit 0: Prd. control n° 1 = fast (0)      Bit 8: Prd. control n° 9 = normal (1) Bit 1: Prd. control n° 2 = normal (1)    Bit 9: Prd. control n°10 = fast (0) Bit 2: Prd. control n° 3 = normal (1)    Bit 10: Prd. control n°11 = normal (1) Bit 3: Prd. control n° 4 = fast (0)      Bit 11: Prd. control n°12 = normal (1) Bit 4: Prd. control n° 5 = normal (1)    Bit 12: Prd. control n°13 = fast (0) Bit 5: Prd. control n° 6 = normal (1)    Bit 13: Prd. control n°14 = normal (1) Bit 6: Prd. control n° 7 = fast (0)      Bit 14: Prd. control n°15 = normal (1) Bit 7: Prd. control n° 8 = normal (1)    Bit 15: Prd. control n°16 = normal (1)
%MWlp.2.c\0.0.30	0 0x0000	Cycle time settings for periodic controls n°17 to 26: Not used
%MWlp.2.c\0.0.31	7 690 0x1E0A	Modbus addresses: Slave n°1 (bits 0-7) = 10 / Slave n°2 (bits 8-15) = 30
%MWlp.2.c\0.0.32	10 764 0x2A0C	Modbus addresses: Slave n°3 (bits 0-7) = 12 / Slave n°4 (bits 8-15) = 42
%MWlp.2.c\0.0.33	33 0x0021	Modbus addresses: Slave n°5 (bits 0-7) = 33 / Slave n°6 (bits 8-15) = 0
%MWlp.2.c\0.0.34	0 0x0000	Modbus addresses: Slave n°7 (bits 0-7) = 0 / Slave n°8 (bits 8-15) = 0
%MWlp.2.c\0.0.35	0 0x0000	Not used (bits at 0)
.....	.....	.....
%MWlp.2.c\0.0.49	0 0x0000	Not used (bits at 0)

# Appendix B: LUFPI Gateway Settings

## FIPIO Objects Available for Programming

Only the Modbus slave control words (%IWp.2.c\0.0 to %IWp.2.c\0.0.26), the Modbus slave command words (%QWp.2.c\0.0 to %QWp.2.c\0.0.27) and the LUFPI gateway LAS service (%IWp.2.c\0.0.27) are discussed again here, as no other FIPIO objects are modified in the following example.

See Section 5, for a description of all the gateway's FIPIO objects.

**Inputs %IWp.2.c\0.0 to %IWp.2.c\0.0.26** correspond to the gateway's 26 periodic control words, plus one reserved word. As only control words n°1 to 16 are used, the only useful inputs are %IWp.2.c\0.0 to %IWp.2.c\0.0.15. Inputs %IWp.2.c\0.0.16 to %IWp.2.c\0.0.26 must therefore not be used:

LUFPI input	Corresponding periodic word	Corresponding slave and exchange (read)
%IWp.2.c\0.0	Periodic control word n° 1	Slave n°1 (TeSys U @10): Status of motor starter
.....	.....	.....
%IWp.2.c\0.0.15	Periodic control word n°16	Slave n°5 (ATV58 @33): Current in motor
%IWp.2.c\0.0.16	Periodic control word n°17	11 words not used (inputs at value 0x0000)
.....	.....	
%IWp.2.c\0.0.25	Periodic control word n°26	
%IWp.2.c\0.0.26	Reserved word	

**Outputs %QWp.2.c\0.0 to %QWp.2.c\0.0.27** correspond to the gateway's 26 periodic command words, plus two reserved words. As only command words n°1 to 9 are used, the only useful outputs are %QWp.2.c\0.0 to %QWp.2.c\0.0.8. Outputs %QWp.2.c\0.0.9 to %QWp.2.c\0.0.27 must therefore not be used:

LUFPI output	Corresponding periodic word	Corresponding slave and exchange (write)
%IWp.2.c\0.0	Periodic command word n° 1	Slave n°1 (TeSys U @10): Motor starter command
.....	.....	.....
%IWp.2.c\0.0.8	Periodic command word n° 9	Slave n°5 (ATV58 @33): Internal command reg. (app)
%IWp.2.c\0.0.9	Periodic command word n°10	19 words not used by the gateway
.....	.....	
%IWp.2.c\0.0.25	Periodic command word n°26	
%IWp.2.c\0.0.26	Reserved words	
%IWp.2.c\0.0.27		

**NOTE:** Only the mapping for the first and last inputs (or outputs) is shown here, as the intervening periodic words were described in the tables in Modbus Exchanges, Appendix B.

Only the first 5 bits of the input word assigned to the list of active slaves (LAS) service are actually used. A Modbus slave is present if the corresponding bit is set to 1.

See also Section 5.2 for more details on the list of active slaves service.

### LAS service (%IWp.2.c\0.0.27)

Bit	Modbus slave	Address
X0	TeSys U motor starter	10
X1	TeSys U motor starter	30
X2	Altistart 48 soft start	12
X3	Altistart 48 soft start	42
X4	Altivar 58 speed drive	33
X5-X15	Not used (bits at 0)	—

# Appendix C: Standard Configuration

The configuration described below is the standard configuration used when implementing the LUF1 gateway.

**NOTE:** This chapter mainly gives the user information about the performances obtained on the downstream Modbus network. It allows users to decide whether, for example, they should change the timing of periodic exchanges with one or more of the TeSys U motor starters (see Chapter 7).

## Configuring Modbus Exchanges

The LUF1 gateway carries out two types of exchanges with each of the 8 TeSys U motor starters. These exchanges are periodic and allow you to control and monitor the motor starter.

Function	Modbus function	Number of bytes (1)	Exchange between the LUF1 gateway and the TeSys U motor starter
0x03	Read Holding Register	11.5 + 10.5	Periodic reading (300 ms period) of the TeSys U motor starter's status register (address 455 = 0x01C7) only
0x10	Preset Multiple Registers	14.5 + 11.5	Periodic writing (300 ms period) of the TeSys U motor starter's status register (address 704 = 0x02C0) only

(1) Number of bytes in the Query + number of bytes in the Response, plus a silence time of 3.5 characters for each of these two frames. Each byte will be transmitted in the form of a group of 10 bits (8 data bits, 1 start bit and 1 stop bit). These values allow you to calculate the approximate amount of traffic on the downstream Modbus network as follows:

Volume of periodic traffic (300 ms period) ..... [ (11.5 + 10.5) + (14.5 + 11.5) ] x (8 + 1 + 1) = 480 bits

For 1 TeSys U motor starter ..... 1 x 480 x (1 000 ÷ 300) = 1,600 bits/s

For 8 TeSys U motor starters ..... 8 x 480 x (1 000 ÷ 300) = 12,800 bits/s

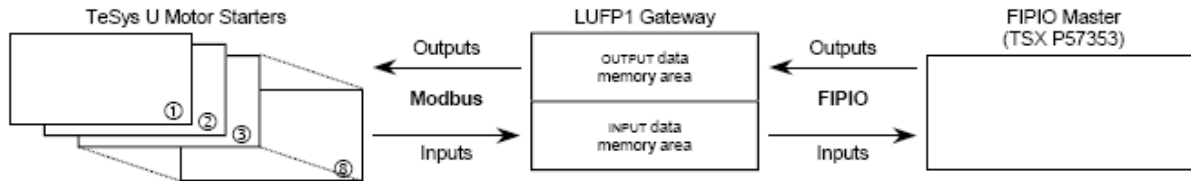
As a result, on a network operating at 9,600 bits/s, you will need to considerably increase the cycle time for all or some of the periodic Modbus commands. On the other hand, at a speed of 19,200 bits/s (the default speed), the available bandwidth is sufficient to support acceptable communications, even with the occasional degraded mode (retransmission of frames), and to allow the use of aperiodic parameter exchanges (using the PKW service described in Section 5.3).

# Appendix C: Standard Configuration

## Content of the Gateway's DPRAM Memory

The LUFFP1 gateway's DPRAM memory contains all of the data exchanged between the gateway and the 8 TeSys U motor starters.

The flow of data between the TeSys U motor starters, the gateway and the FIPIO master is shown below, in order to highlight the role of the gateway's memory in these exchanges:



**FED C32 P** Configuring the gateway with this profile limits the total number of available words to 26, i.e. 52 bytes from among the 52 input bytes and the 52 output bytes.

**FED C32** This profile makes available all of the gateway's 52 input bytes and 52 output bytes.

## Input Data Memory Area

The gateway has 52 input bytes. Only the first 16 bytes are used.

Service	Address	Size	Description
Periodic communications — Control of TeSys U motor starters	0x0000	1 word	Value of status register for motor starter ①
	0x0002	1 word	Value of status register for motor starter ②
	0x0004	1 word	Value of status register for motor starter ③
	0x0006	1 word	Value of status register for motor starter ④
	0x0008	1 word	Value of status register for motor starter ⑤
	0x000A	1 word	Value of status register for motor starter ⑥
	0x000C	1 word	Value of status register for motor starter ⑦
	0x000E	1 word	Value of status register for motor starter ⑧
—	0x0010	1 byte	Free input area (36 bytes)
	...	...	
	0x0033	1 byte	
—	0x0034	1 byte	Reserved memory locations (2 bytes, always at 0x0000)
	0x0035	1 byte	
LAS service (1)	0x0036	1 word	Reserved memory locations (2 bytes)
PKW service (RESPONSE) (1)	0x0038	1 byte	Reserved memory locations (8 bytes)
	...	...	
	0x003F	1 byte	
—	0x0040	1 byte	Unusable input area (448 bytes)
	...	...	
	0x01FF	1 byte	

- (1) The data for these services are not actually stored in these gateway memory locations, but the mapping between the gateway memory and the corresponding FIPIO inputs / outputs (see Section 5.1.3) means that these locations are not to be used. If they were, the mapping would overwrite the stored data with data from the services.



# Appendix C: Standard Configuration

## Output Data Memory Area

The gateway has 52 output bytes. Only the first 16 bytes are used.

Service	Address	Size	Description
Periodic communications — Command of TeSys U motor starters	0x0200	1 word	Value of command register for motor starter ①
	0x0202	1 word	Value of command register for motor starter ②
	0x0204	1 word	Value of command register for motor starter ③
	0x0206	1 word	Value of command register for motor starter ④
	0x0208	1 word	Value of command register for motor starter ⑤
	0x020A	1 word	Value of command register for motor starter ⑥
	0x020C	1 word	Value of command register for motor starter ⑦
	0x020E	1 word	Value of command register for motor starter ⑧
—	0x0210 ... 0x0233	1 byte ... 1 byte	Free output area (36 bytes)
—	0x0234 ... 0x0237	1 byte ... 1 byte	Reserved memory locations (4 bytes not used by gateway)
PKW service (COMMAND) (1)	0x0238 ... 0x023F	1 byte ... 1 byte	Reserved memory locations (8 bytes)
—	0x0240 ... 0x03FF	1 byte ... 1 byte	Unusable output area (448 bytes)

(1) See note on “PKW service (RESPONSE)” in the previous table.

## Total Number of Modbus Queries and Responses

The total number of Modbus queries and responses is 32 (2 periodic queries and 2 periodic responses for each of the 8 TeSys U motor starters). Since the total number of Modbus queries and responses one can configure for a single gateway is limited to 52 (whichever profile is used: FED C32 or FED C32 P), there are 20 spare Modbus queries and responses (i.e. the equivalent of 10 Modbus commands).

This reserve is therefore enough for the addition of any single Modbus command to each of the TeSys U motor starters, as this requires 16 Modbus queries and responses (1 query and 1 response for each of the 8 motor starters).

# Appendix D: Sample Use under PL7 PRO



A practical example can be found on the <http://www.schneider-electric.com> website.








- The first file, "LUFP1\_FEDC32\_Example.cfg", is an ABC-LUFP Config Tool file; its content corresponds to the system architecture described in Section 4.1.1. The gateway must therefore be configured using profile **FED C32** under PL7 PRO (see Chapter 6). This ABC-LUFP Config Tool file corresponds to the standard configuration used in this guide.
- The second file, "lufp1 – exemple du tutorial fedc32p tsx57252.stx ", is a PL7 PRO file and provides the example for a TSX Premium PLC with a TSX 57252 processor where the LUFP1 is configured under PL7.
- The third file, "lufp1\_tutorial\_en\_fedc32p\_tsx57353.stx", is a PL7 PRO file and provides the example for a TSX Premium PLC with a TSX 57353 processor where the LUFP1 is configured under PL7.
- The fourth file, "lufp1\_tutorial\_en\_fedc32\_tsx57252.stx", is a PL7 PRO file and provides the example for a TSX Premium PLC with a TSX 57252 processor where the LUFP1 is configured under ABC-LUFP Config Tool. The configuration "LUFP1 - FED C32 - Exemple.cfg" needs to be loaded onto the LUFP1.
- The fifth file, "lufp1\_tutorial\_en\_fedc32\_tsx57353.stx", is a PL7 PRO file and provides the example for a TSX Premium PLC with a TSX 57353 processor where the LUFP1 is configured under ABC-LUFP Config Tool. The configuration "LUFP1\_FEDC32\_Example.cfg" needs to be loaded onto the LUFP1.

The content and application of these examples is dealt with in the following sections.

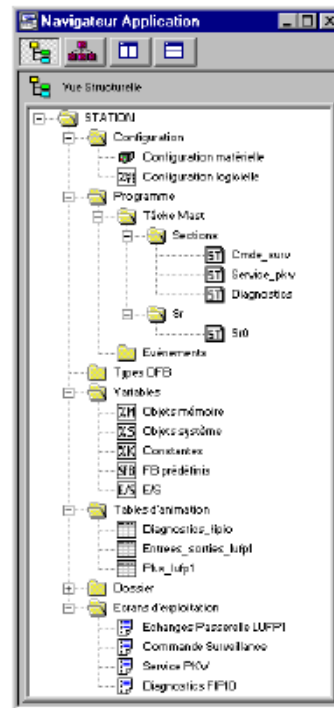
The configuration of the first file corresponds exactly to what is described in the previous sections; its content is therefore not detailed here. However, the PL7 PRO file is described below, in terms of the structure of the program sections used and the related operating screens.

## Overview of the "LUFP1\_tutorial" example

In this example, the various program sections and sub-programs (icon ) and operations screens (icon ) are organized as follows:

- Command and control screen, 8 TeSys U motor starters:
  -  Cmd\_mon
  -  Sr0
  -  Command Control
- Using the indexed periodic variables (PKW) service to read or write the value of a register:
  -  Pkw\_service
  -  PKW Service
- FIPIO and LAS service diagnostic commands:
  -  Diagnostics
  -  FIPIO Diagnostics

Each of the above groupings is described in a separate section.




## Appendix D: Sample Use under PL7 PRO

---

This description remains concise, because it is only meant to describe the overall operation of the program and the way to use the related screen. If you need a greater degree of detail, don't hesitate to review the actual content of the example under PL7 PRO.

The source code contains numerous remarks to help you understand how it works. Each "program" file starts with a short description of the way it operates, and each line includes a comment.

Each variable used has a symbol whose name describes its use. Use the keyboard shortcuts "Ctrl+E" and "Ctrl+F" to display the variables by address (e.g. %MW100) or by symbol (e.g. Pkw\_checked\_boxes\_dn).

Three animation tables (icon ) have been created, "FIPIO Diagnostics", "lufp1\_inputs\_outputs" and "lufp1\_Pkw".

However, the operating screens provided are more than sufficient to present the application data used throughout this example.

### Displaying LUF P1 Gateway Inputs / Outputs

The "LUF P1 gateway exchanges" operating screen comprises two separate panes:

- Hexadecimal display of gateway input data (%IW0.2.2\0.0 to %IW0.2.2\0.0.31) in the **INPUTS** frame. These inputs are named and grouped in the same way as in this guide (see Section 4.2.7). Of course, the display of these input data is correct only if the standard gateway configuration is used.

In the case of the standard configuration, these input data comprise the periodic Modbus data (command and control of TeSys U motor starters), the Modbus aperiodic data (indexed periodic variables / PKW service) and the input word of the list of active slaves (LAS) service, as well as the free and reserved memory locations.

- Hexadecimal display of gateway output data (%QW0.2.2\0.0 to %QW0.2.2\0.0.31) in the **OUTPUTS** frame. *As above.*

### Command and Control of the 8 TeSys U Motor Starters

The "Command Control" operating screen (see illustration on next page) allows you to monitor the status of the 8 TeSys U motor starters, numbered from 1 to 8. It also enables you to command them individually by means of a number of buttons.

Registers 455 and 704 on each of the 8 TeSys U motor starters are used to conduct this control and command function:

#### 455 – TeSys U status register (IEC61915)

Bit 0 ..... Motor starter ready  
Bit 1 ..... Contactor in ON position ( **1** )  
Bit 2 ..... Fault (*trip or dropout*)  
Bit 3 ..... Alarm present  
Bit 4 ..... Specific: *Tripped* ( **I>>** )  
Bit 5 ..... Specific: Fault reset authorized  
Bit 6 ..... Specific: A1-A2 supplied  
Bit 7 ..... Specific: Motor running  
Bits 8-13 .. Motor current (2#10 0000 = 200%)  
Bit 14 ..... Reserved: Local control  
Bit 15 ..... Ramping (motor starting)

#### 704 – Command register (IEC61915)

Bit 0 ..... Reserved: Run forward  
Bit 1 ..... Reserved: Run reverse  
Bit 2 ..... Reserved (stop)  
Bit 3 ..... Reset  
Bit 4 ..... Reserved (emergency start)  
Bit 5 ..... Self test: Triggering test (*trip*)  
Bit 6 ..... Reserved (low speed)  
Bits 7-11 .. Reserved by IEC61915 standard  
Bit 12 ..... Specific: Overload (*shunt trip*)  
Bit 13 ..... Specific: Pause (reserved for adjustment)  
Bits 14-15 Specific: Reserved

# Appendix D: Sample Use under PL7 PRO

These statuses and commands are grouped into two sections: "General Status", for the general operating mode of the motor starters, and "Motor" for that of the controlled motors. The last section, "DEBUG COMM.", displays the two registers %IW and %QW used for each motor starter.

This screen is shown opposite, but only for the first motor starter, as it is identical to the 7 others.

Most of the displays in this operating screen are directly linked to the %MW registers, which in turn contain the values of registers %IW4.0.1 to %IW4.0.8 and %QW4.0.1 to %QW4.0.8 (status and command registers of the TeSys U motor starters). Only the indirect commands and statuses are described below.

The "Cmd\_mon" program performs the following tasks:

- It copies the values of words %MW20 to %MW27 into output registers %QW0.2.2\0.0 to %QW0.2.2\0.0.7 and copies the values of input registers %IW0.2.2\0.0 to %IW0.2.2\0.0.7 into words %MW10 to %MW17.

These data are copied because word bit extraction can be carried out on %MW-indexed words, but not on %IW-indexed words. The sub-program "Sr0" makes extensive use of word indexing, as it can be used to manage any motor starter, the latter being designated by the word %MW0. E.g. "%MW10[%MW0]:X13" is authorized, but not "%IW0.2.2\0.0 [%MW0]:X13".

TeSys U		n°1
<b>General</b>		1
Ready — ON		<input checked="" type="checkbox"/> <input checked="" type="checkbox"/>
PAUSE — Start	<input type="checkbox"/>	
— Stop	<input type="checkbox"/>	
Alarm — Fault	<input type="checkbox"/>	<input type="checkbox"/>
? Fault reset ?	<input type="button" value="NO"/>	
Fault reset		
Tripped	<input type="button" value="SHUNT"/>	
Shunt Trip	<input type="button" value="TEST"/>	
Test Trip		
<b>MOTOR</b>		
Motor current	<input type="text" value="50"/>	
Motor current (%)	<input type="text" value="156.25"/>	
Run Forward	<input type="button" value="FOR"/>	
Run Reverse	<input type="button" value="REV"/>	
Stopping	<input type="button" value="STOP"/>	
<b>DEBUG COMM.</b>		
TeSys U command	<input type="text" value="16#0001"/>	
TeSys U status	<input type="text" value="16#32C3"/>	

Furthermore, the input and output words are copied one by one, as PL7 PRO does not support expressions such as "%IW0.2.2\0.0:8".

- It runs the "Sr0" sub-program calling loop to command and control the 8 TeSys U motor starters. At each iteration of the loop, the word %MW0 ("Module") takes a value from 0 to 7 in order to serve as an index for words %MW10 to %MW17 (inputs) and %MW20 to %MW27 (outputs).

The "Sr0" sub-program is called once per motor starter from the "Cmd\_mon" program. Each of these calls should be made with a different value in the word %MW0 (between 0 and 7) as it is used to index the status word and the command word of the same motor starter. The sub-program is used by the screen shown above to perform the following tasks:

- Determining the overall status of the motor starter with:  in case of triggering (tripped),  if the contactor on the motor starter is in the ON position, and  in all other cases.
- Using the commands generated by buttons  and  to switch the motor starter into or out of pause mode (bit 13 of the command word).

**NOTE:** The "pause" mode should not be used for a normal application; use the ,  and  buttons instead (see next page).

- Activating the motor starter fault reset command (bit 3 of the command word) if the user pressed the  button and the fault LED is on: . Once this command has been activated, it is cancelled as soon as the LED goes off: .
- Activating the motor starter self test command (trip) (bit 5 of the command word) if the user pressed the  button and the "Tripped" LED is off: . Once the command has been activated, it is cancelled as soon as the LED goes on: .

## Appendix D: Sample Use under PL7 PRO

- Evaluating the value of the motor current, given as a percentage of the IR current value (unit = % FLA). Bits 8 (LSB) to 13 (MSB) of the motor starter status word are extracted and the appropriate unit, i.e. 3.125% FLA, is then used to evaluate the current. The maximum value is therefore 63, or 196.875% FLA.
- Using the **FOR**, **REV** and **STOP** buttons exclusively to enable only one of the three following commands at a time, while resetting the other two to zero, in descending order of priority: stop (bit 2 of the command word), run forward (bit 0), and run reverse (bit 1).

### Using the Indexed Periodic Variables (PKW) Service

The “**PKW service**” operation screen allows the user to command the reading or the writing of a register on a Modbus slave, on the gateway, or on all the Modbus slaves at once (broadcast). The screen is divided up into a number of frames, as illustrated below:

<ul style="list-style-type: none"><li>• The first frame, “DN (Device Number)”, is used to select the station to be polled by the PKW service. Only one box can be checked at any given time. Depending on the box selected when the “SEND command” button is clicked, the DN field (MSB of %QW\0.2.2\0.0.29) of the PKW service command will be updated accordingly (see Section 5.3.2). The “Other address:” box allows the user to type in an address ranging from 1 to 247.</li></ul>	<table border="1"><tr><td><b>DN (Device Number)</b> <input type="checkbox"/> 1st Modbus Slave <input type="checkbox"/> TeSys U n°1 motor starter <input type="checkbox"/> TeSys U n°2 motor starter <input checked="" type="checkbox"/> TeSys U n°3 motor starter <input type="checkbox"/> TeSys U n°4 motor starter <input type="checkbox"/> TeSys U n°5 motor starter <input type="checkbox"/> TeSys U n°6 motor starter <input type="checkbox"/> TeSys U n°7 motor starter <input type="checkbox"/> TeSys U n°8 motor starter <input type="checkbox"/> Other address : <input type="text" value="0"/> <input type="checkbox"/> LUFPP1 gateway <input type="checkbox"/> Modbus broadcast</td><td><b>PKE (Register Address)</b> <input type="text" value="0"/></td></tr><tr><td></td><td><b>PWE (Value to write)</b> <input type="text" value="0"/></td></tr><tr><td></td><td><b>R/W (Read/Write)</b> <input type="checkbox"/> RESET command <input checked="" type="checkbox"/> Read a register <input type="checkbox"/> Write a register</td></tr><tr><td colspan="2"><b>ENVOI commande</b></td></tr></table>	<b>DN (Device Number)</b> <input type="checkbox"/> 1st Modbus Slave <input type="checkbox"/> TeSys U n°1 motor starter <input type="checkbox"/> TeSys U n°2 motor starter <input checked="" type="checkbox"/> TeSys U n°3 motor starter <input type="checkbox"/> TeSys U n°4 motor starter <input type="checkbox"/> TeSys U n°5 motor starter <input type="checkbox"/> TeSys U n°6 motor starter <input type="checkbox"/> TeSys U n°7 motor starter <input type="checkbox"/> TeSys U n°8 motor starter <input type="checkbox"/> Other address : <input type="text" value="0"/> <input type="checkbox"/> LUFPP1 gateway <input type="checkbox"/> Modbus broadcast	<b>PKE (Register Address)</b> <input type="text" value="0"/>		<b>PWE (Value to write)</b> <input type="text" value="0"/>		<b>R/W (Read/Write)</b> <input type="checkbox"/> RESET command <input checked="" type="checkbox"/> Read a register <input type="checkbox"/> Write a register	<b>ENVOI commande</b>	
<b>DN (Device Number)</b> <input type="checkbox"/> 1st Modbus Slave <input type="checkbox"/> TeSys U n°1 motor starter <input type="checkbox"/> TeSys U n°2 motor starter <input checked="" type="checkbox"/> TeSys U n°3 motor starter <input type="checkbox"/> TeSys U n°4 motor starter <input type="checkbox"/> TeSys U n°5 motor starter <input type="checkbox"/> TeSys U n°6 motor starter <input type="checkbox"/> TeSys U n°7 motor starter <input type="checkbox"/> TeSys U n°8 motor starter <input type="checkbox"/> Other address : <input type="text" value="0"/> <input type="checkbox"/> LUFPP1 gateway <input type="checkbox"/> Modbus broadcast	<b>PKE (Register Address)</b> <input type="text" value="0"/>								
	<b>PWE (Value to write)</b> <input type="text" value="0"/>								
	<b>R/W (Read/Write)</b> <input type="checkbox"/> RESET command <input checked="" type="checkbox"/> Read a register <input type="checkbox"/> Write a register								
<b>ENVOI commande</b>									

- The second frame, “PKE (Address Register)”, allows you to set the address of the register to be read / written. When the “SEND command” button is clicked, the value currently entered in this frame is copied into the PKE field (%QW\0.2.2\0.0.28) of the PKW service command.
- The third frame, “PWE (Value to be written)”, will only be used for write commands on registers. When the “SEND command” button is clicked, the write command is copied into the first word (%Q\0.2.2\0.0.30) in the PWE field of the PKW service command, providing the “Write a register” box is checked.
- The fourth frame, “R/W (Read / Write)”, allows you to select the type of command generated by the PKW service: Read, Write or Reset the response. Only one box can be checked at any given time.
- The “SEND command” button triggers the update of the gateway’s periodic output data (%QW\0.2.2\0.0.28 to %QW\0.2.2\0.0.31) so that the gateway generates a command that matches the settings and values in the frames above. These updates are performed in the “**Pkw\_service**” program. Of course, using periodic data means that a new command can only be generated when one of the values involved changes from one command to the next. This is why the “Reset” command is useful, for example, so that the same command can be repeated several times.

**E.g.** In the above example, the frames are configured to issue a read command (0x52) on the register located at address 455 (0x01C7) of TeSys U motor starter n°3 (0x03).

# Appendix D: Sample Use under PL7 PRO

- The last frame, beneath and apart from the others, displays the output data transmitted to the gateway in order to generate the corresponding PKW command, and the input data from the gateway's response to this command.

The following **example** contains the PKW service output and input data for the instances described at the bottom of the previous page. The reading value is 0x01C3.

%QWp.2.c\0.0.28	16#01C7	PKE - Adresse paramètre DN - Device Number (MSB)+ R/W - Function Code (LSB)	16#01C7	%IWp.2.c\0.0.28
%QWp.2.c\0.0.29	16#0352		16#0352	%IWp.2.c\0.0.29
%QWp.2.c\0.0.30	16#0000	PWE - Parameter Value (n°1)	16#00C3	%IWp.2.c\0.0.30
%QWp.2.c\0.0.31	16#0000	PWE - Parameter Value (n°1)	16#0000	%IWp.2.c\0.0.31

The “Pkw\_service” program includes the ST instructions that use the values entered in the “PKW service” frame, described on the previous page, to update the PLC outputs for the PKW service command on the gateway (%QWp.2.c\0.0.28 to %QWp.2.c\0.0.31). The program handles the following tasks:

- It tests the checkboxes in the “DN (Device Number)” and “R/W (Read / Write)” frames. If, in either frame, none of the boxes is checked, one of them will be selected by default (DN = 1st Modbus slave; R/W = Read a register).
- It compares the values over two consecutive PLC cycles in order to keep only one box checked at any given time in each frame.
- When it detects a click on the “SEND command” button, it updates the local variables, followed by the outputs corresponding to the gateway's PKW service. The purpose of this two-stage update procedure is to isolate the use of the outputs from the rest of the program.

## LUFP1 Gateway Diagnostics

The “FIPIO Diagnostics” operating screen provides a graphic example of the use of the list of active slaves (LAS) service, specific to the gateway, and of the FIPIO diagnostic objects that a TSX Premium PLC assigns to the stations on its FIPIO network. See the PL7 PRO user manual or online help for more details about using FIPIO diagnostics on Premium PLCs.

The screen is divided up into three separate frames, as illustrated below:

- List of active Modbus slaves:** Each “TeSys U n°” in this frame indicates that the corresponding TeSys U motor starter is present. The hexadecimal value in the top right corner is that of the FIPIO input for the list of active slaves (LAS) service, namely %IWp.2.c\0.0.27. See Section 5.2 for a detailed description of the LAS service.

In this example, bits 2 and 4 of the LAS service input word have been reset by the gateway. Modbus slaves n°3 and 5 are therefore absent.

### List of modbus Active Slaves

2#0000000011101011

LUFP1

TeSys U n°1

TeSys U n°2

TeSys U n°4

TeSys U n°6

TeSys U n°7

TeSys U n°8

The “LUFP1” box uses a system bit on the Premium PLC to indicate whether the gateway is present on, or absent from, the FIPIO network (colored blue or red accordingly). This system bit (%SW128:X2 in this example) is described in Section 5.4.1.

# Appendix D: Sample Use under PL7 PRO

- Diagnostics: Implicit exchanges:** The first two rows of the table in this frame correspond to the “Module fault” (%I\p.2.c\0.MOD.ERR) and “Channel fault” (%I\p.2.c\0.0.ERR) described at the beginning of Section 5.4. The last two rows correspond to the words %MW\p.2.c\0.0 and %MW\p.2.c\0.0.1, both described in Section 5.5.

If the FIPIO connection between the PLC and the gateway is lost, the first two rows signal the presence of just such a fault (see example).

Diagnostics : Implicit Exchanges		
%I\p.2.c\0 0.MOD.ERR		
%I\p.2.c\0.0 0.0.ERR		
%MW\p.2.c\0.0 0.0 16#0000	Status of the exchange management	• No exchanges
%MW\p.2.c\0.0.1 0.0.1 16#0000	Exchange Feedback	<ul style="list-style-type: none"> <li>• Status reading : OK</li> <li>• Command parameter received : ACCEPTED</li> <li>• Adjust parameter received : ACCEPTED</li> <li>• Configuration parameter received : ACCEPTED</li> </ul>
%I\p.2.c\0 0.MOD.ERR	Module Fault	Module Fault (at least one channel is in default, ...) > Communication breakdown between the PLC and the LUFF1
%I\p.2.c\0.0 0.0.ERR	Channel Fault	Channel Fault > Communication breakdown between the PLC and the LUFF1

If these faults arise, the “Explicit exchanges”, refreshed every 5 seconds (see below), indicate the presence of an “Internal fault” (Standard channel status) and “Gateway absent” (Module status). A sample screen is shown below.

- Diagnostics: Explicit exchanges:** Both rows of the table in this frame are refreshed every 5 seconds. They indicate the nature of any errors detected either by the gateway or by the FIPIO master.

See Sections 5.4.2 and 5.4.3 for a description of the explicit objects introduced here.

The example opposite illustrates one type of error (gateway out of supply).

Diagnostics : Explicit exchanges		
%MW\p.2.c\0 0 MOD.2 16#0040	Module Status	• Gateway is missing
%MW\p.2.c\0.0 0 0.2 16#0010	Standard Channel Status	• Internal Fault

The “**Diagnostics**” program includes all the ST instructions for generating the explicit FIPIO diagnostic commands. The program handles the following tasks:

- A timeout (%TM4) is used to generate two explicit diagnostic commands every 5 seconds. One of these commands reads the “Module status” and the other reads the “Standard channel status”.
- A second timeout (%TM5) is used to prolong the “Read status in progress” alert (%MW\p.2.c\0.0:X0) by 500 ms, so that it can be seen in the “Exchange management status” line of the “**FIPIO Diagnostics**” screen. This indicates the activity triggered by the two explicit diagnostic commands (see previous point).

# Appendix E: Modbus Commands

Only the Modbus commands shown in the table opposite are supported by the gateway. The structure of the query and response frames for each of these commands is described in the rest of this chapter.

Function code		Broadcast (1)	Modbus command
3	0x03	—	Read Holding Registers
6	0x06	Yes	Preset Single Register
16	0x10	Yes	Preset Multiple Registers

(1) The content of this column shows whether the command can be added (“Yes”) or not (“—”) to the list of commands for a broadcaster node, known as “Broadcaster” in ABC-LUFP Config Tool.

In the following sections, each byte in the query and response frames of a Modbus command is described, in sequence, with the exception of the fields opposite: these are always present in the queries and responses of Modbus commands.

The first two bytes of these frames correspond to the “Slave Address” and “Function” fields. The last two bytes are the two “Checksum” bytes.

Slave Address	- Non-modifiable (Modbus address: 1 to 247. Addresses 65, 126, and 127 prohibited)
Function	- Non-modifiable (Modbus command code)
... Other fields...	... Specific Modbus command data...
Checksum (Lo)	- Type of error check
Checksum (Hi)	- Number of the 1st byte checked

The following descriptions of the Modbus frames are mainly intended to help you to configure the gateway’s Modbus exchanges using ABC-LUFP Config Tool. See the documentation of each Modbus slave to check for any restrictions on the use of these frames (number of registers which can be read or written in a single Modbus command, for example).

It is preferable to get hold of a standard Modbus document, such as the *Modicon Modbus Protocol Reference Guide* (ref. PI-MBUS-300 Rev. J), to see how the elements displayed in ABC-LUFP Config Tool map match the content of the corresponding Modbus frames. Here is a mapping example for a complete frame (including the start and end of frame fields shown above), based on the “**Read Holding Registers**” Command (0x03) section.

	Elements under ABC-LUFP Config Tool	Modbus frame fields	Size
<b>Modbus query</b>	Slave Address	Slave no.	1 byte
	Function code	Function no.	1 byte
	Starting Register Address	No. of 1st word (MSB / LSB)	2 bytes
	Number of Registers	Number of words (MSB / LSB)	2 bytes
	Checksum	CRC16 (LSB / MSB)	2 bytes
<b>Modbus response</b>	Slave Address	Slave no.	1 byte
	Function code	Function no.	1 byte
	Byte count	Number of bytes read	1 byte
	Data	Value of 1st word (MSB / LSB)	2 bytes
		Value of last word (MSB / LSB)	2 bytes
	Checksum	CRC16 (LSB / MSB)	2 bytes

Section 7.12 also shows a few examples of mapping between the elements displayed in ABC-LUFP Config Tool and the corresponding Modbus frame fields.



# Appendix E: Modbus Commands

**NOTE:** Here, the notions of “input” and “output” (and assimilated concepts) are meaningless, in as far all Modbus commands have access to the entire memory of the Modbus slave. We continue to use these terms, however, in order to conform to the terminology of the standard Modbus documentation.

## “Read Holding Registers” Command (0x03)

Frame	Field	Value or properties
Query	Starting Address (MSB)	- Address of the 1st output / internal register
	Starting Address (LSB)	
	Number of points (MSB)	- Number of output / internal registers
	Number of points (LSB)	
Response	Byte count	- Number of data bytes = Number of output / internal registers x 2
	Data (first register / MSB)	- Byte swap = “No swapping” (or “Swap 2 bytes”)
	Data (first register / LSB)	
	.....	- Data length = Value of the “Byte count” field
	Data (last register / MSB)	- Data location = Address in the gateway’s input memory
	Data (last register / LSB)	

## “Preset Single Register” Command (0x06)

Frame	Field	Value or properties
Query	Register (MSB)	- Address of the output / internal register
	Register (LSB)	
	Preset data (MSB)	- Byte swap = “No swapping” (or “Swap 2 bytes”)
	Preset data (LSB)	- Data length = 0x0002 - Data location = Address in the gateway’s output memory
Response	Register (MSB)	- Byte swap = “No swapping” (or “Swap 2 bytes”)
	Register (LSB)	- Data length = 0x0002
	Preset data (MSB)	- Data location = Address in the gateway’s input memory
	Preset data (LSB)	<b>NOTE:</b> These data are an echo to the query: in most cases there is no need to feed them back to the FIPIO master.

**NOTE:** Instead of placing the echo of the response to the “Preset Single Register” Command (0x06) in the addresses reserved for FIPIO inputs (0x0000-0x0033), you can put it at 0x0400. This is what the gateway does automatically when you configure periodic command words.

## “Preset Multiple Registers” Command (0x10)

Frame	Field	Value or properties
Query	Starting Address (MSB)	- Address of the 1st output / internal register
	Starting Address (LSB)	
	No. of Registers (MSB)	- Number of output / internal registers
	No. of Registers (LSB)	
	Byte Count	- Number of data bytes = Number of output / internal registers x 2
	Data (first register / MSB)	- Byte swap = “No swapping” (or “Swap 2 bytes”)
	Data (first register / LSB)	
	.....	- Data length = Value of the “Byte count” field
	Data (last register / MSB)	- Data location = Address in the gateway’s output memory
	Data (last register / LSB)	
Response	Starting Address (MSB)	- Address of the 1st output / internal register
	Starting Address (LSB)	
	No. of Registers (MSB)	- Number of output / internal registers
	No. of Registers (LSB)	

# Appendix E: Modbus Commands

## Modbus Protocol Exception Responses

When it cannot process a command issued by a Modbus query, a slave sends an exception response instead of the normal response to the query.

### **WARNING**

#### **UNATTENDED OPERATION OF THE SYSTEM**

With standard Modbus commands, the LUF1 gateway considers that all the exception responses it receives from Modbus slaves are incorrect responses. As a result, it will carry out the re-transmissions configured for the queries involved.

**Failure to follow this instruction can result in death, serious injury, or equipment damage.**

The structure of an exception response is independent of the Modbus command associated with the “Function” field of the query involved. The complete frame of an exception response is shown below:

Slave Address	Modbus address (1 to 247; addresses 65, 126 and 127 prohibited): The value of this field is identical to that of the “Slave Address” field in the query involved.
Function	Command code, with exception indicator: The value of this field is set to 0x80 + the value of the “Function” field in the query involved.
Exception Code	Code indicating the nature of the error which has caused the exception response (see table below).
Checksum (Lo)	Error check
Checksum (Hi)	

Code	Exception	Description of the exception
0x01	ILLEGAL FUNCTION	The query’s “Function” command is not implemented in the Modbus slave software, or it is unable to process it for the moment.
0x02	ILLEGAL DATA ADDRESS	The combination of the query’s “Starting Address” and “No. of Registers” fields (or assimilated fields) gives access to one or more addresses which are not accessible on the Modbus slave.
0x03	ILLEGAL DATA VALUE	The value of one of the Modbus query’s fields is outside the authorized range. This error does not affect the content of the “Data” (or assimilated) fields, as it only governs the fields used in managing the Modbus protocol.
0x04	SLAVE DEVICE FAILURE	An unrecoverable failure has occurred when processing the command.
0x05 (1)	ACKNOWLEDGE	The Modbus slave informs the gateway that it has accepted the command (acknowledgement); but that it will take too long to process it and it cannot afford to wait for the completion of this process before sending a response. The gateway will need to transmit further queries in order to determine whether the command has finished.
0x06 (1)	SLAVE DEVICE BUSY	The Modbus slave informs the gateway that it is already running a command and cannot therefore run the one transmitted to it. The gateway will need to retransmit the query later.
0x07 (1)	NEGATIVE ACKNOWLEDGE	The Modbus slave informs the gateway that it cannot process the requested command. This exception only affects commands 13 and 14 (0x0D and 0x0E). These functions are not part of the standard Modbus commands and are not described in this document.
0x08 (1)	MEMORY PARITY ERROR	The Modbus slave informs the gateway that it has detected a parity error on access to its own memory. This exception only affects standard commands 20 and 21 (0x14 and 0x15), which are not supported by the gateway.

(1) See the standard Modbus documentation for further information about these various scenarios.

# Index

---

- 2**
  - 2-way TSXSCA62 subscriber connector, 19
- A**
  - Address, 23
  - Architecture, 9, 28
- C**
  - Cable
    - type A, 20, 22
  - Communications
    - aperiodic, 44
    - periodic, 44
  - Cycle time, 29
- D**
  - Data exchanged, 11
  - Diagnostic LEDs, 12
  - DIN Rail, 13
- E**
  - End of line connector, 22
- G**
  - Gateway's address, 23
  - GSD file, 36
- L**
  - LEDs, 25
  - Line connector, 22
  - Line termination, 21
  - LU9GC03 hub, 19
- M**
  - Modbus cable, 19
- P**
  - PL7 PRO, 34, 39, 40, 42, 68, 77, 121
  - Product Related Warning, 5
  - Profibus-DP master, 34
  - Protective Earth, 13
- R**
  - Related documents, 5
  - RJ45 connector, 12
- S**
  - SyCon, 28, 34, 35, 37, 73, 76, 116, 121
- T**
  - SCA junction boxes, 17
  - Topology
    - bus, 16
  - TSXCA50 SCA junction box, 19
  - Type A cable, 20
- V**
  - VW3 A8 306 RC double termination, 19
  - VW3 A8 306 TF3 T-junction box, 19

# Glossary

---

<b>0x****</b>	Value expressed in hexadecimal, which is equivalent to the H****, ****h and 16#**** notations, sometimes used in other documents. <b>NOTE:</b> The ABC-LUFP Config Tool software uses the 0x**** notation. e.g. 0x0100 = 16#0100 = 256.
<b>2#**** ****</b>	Value expressed in binary. The number of ‘.’ digits depends on the size of the item of data represented. Each nibble (group of 4 bits) is separated from the other nibbles by a space. Examples: byte 2#0010 0111 = 39, word 2#0110 1001 1101 0001 = 0x69D1 = 27089.
<b>ABC-LUFP Config Tool</b>	Name of the PC software used to configure and monitor the LUFP7 Profibus-DP/Modbus Gateway.
<b>ASIC</b>	Integrated circuits specific to a given user and application, covering two major families: pre-characterised processes and pre-distributed networks.
<b>ATS</b>	Abbreviation of “Altistart” (soft start- soft stop unit).
<b>ATV</b>	Abbreviation of “Altivar” (drive).
<b>CRC</b>	Cyclical Redundancy Check.
<b>LED</b>	Light-Emitting Diode.
<b>DP</b>	Decentralised Periphery (remote I/O). Profibus version or <i>protocol</i> meant for quick communication with remote I/O. This is the only Profibus protocol supported by the LUFP7 gateway.
<b>DPM1</b>	Class 1 DP master: the central automatic control of a Profibus-DP network. It resets and controls I/O transfers and slave diagnostics on the network. One can have several DPM1 stations on a given Profibus-DP network, each one steering its own slaves.
<b>DPM2</b>	Class 2 DP master: The programming, configuration, and diagnostic device of a Profibus-DP network.
<b>Fieldbus</b>	A term referring to the upstream Profibus-DP network in ABC-LUFP Config Tool.
<b>FMS</b>	Profibus-FMS messaging system that defines the objects and application services applicable to these objects. By extension, the Profibus version or <i>protocol</i> dedicated to complex and advanced communication tasks at the cell level. This protocol is not supported by the LUFP7 gateway.
<b>GSD</b>	Electronic equipment database, also called GSD <i>file</i> . This term designates the format of the files (“.gsd” extension) that are used by a Profibus master configuration and adjustment tool to configure their exchanges according to that same protocol.
<b>Handshake</b>	An old term referring to the two registers used for initialising and carrying out diagnostics of the LUFP7 gateway. This term has been replaced by the expression “Control/Status Byte”.
<b>LRC</b>	Longitudinal Redundancy Check.
<b>LSB</b>	Least significant byte in a 16-bit word.

# Glossary

---

<b>MSB</b>	Most significant byte in a 16-bit word.
<b>Node</b>	A term referring to the connection point of a Modbus slave under ABC-LUFP Config Tool.
<b>PA</b>	Profibus version or protocol dedicated to process automation. This protocol is not supported by the LUFP7 gateway.
<b>PDP</b>	Profibus-DP (see “DP” above).
<b>PI</b>	Profibus International. This term designates the international organization for users of the Profibus protocol. It is responsible for federating Profibus skills centres, scattered throughout the 20 largest industrial countries. The list of user groups of the Profibus protocol is available on the Profibus web site, at <a href="http://www.profibus.com/">http://www.profibus.com/</a> . To get general-purpose support on Profibus, please email to PI, at: <a href="mailto:Profibus_international@compuserve.com">Profibus_international@compuserve.com</a> .
<b>PNO</b>	This term designates the national and local associations of Profibus protocol users.
<b>PPO</b>	Parameter Process data Object. This term designates the type and size of the data exchanged between a Profibus master and slave. In the case of the LUFP7 gateway, PPOs are not used to configure its exchanges on the Profibus network.
<b>Profibus</b>	PROcess Field BUS.
<b>PSU</b>	Power supply.
<b>Sub-Network</b>	A term referring to the downstream Modbus network under ABC-LUFP Config Tool.
<b>TSDI</b>	Initiating station request time.
<b>TSDR</b>	Answering station response time.
<b>XML</b>	EXtensible Markup Language. The language used by ABC-LUFP Config Tool to import/export the configuration of a Modbus slave.