Premium and Atrium Using EcoStruxure™ Control Expert

Fipio Bus
Setup Manual

(Original Document)

12/2018

35008155.11

www.schneider-electric.com

Schneider Electric

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

You agree not to reproduce, other than for your own personal, noncommercial use, all or part of this document on any medium whatsoever without permission of Schneider Electric, given in writing. You also agree not to establish any hypertext links to this document or its content. Schneider Electric does not grant any right or license for the personal and noncommercial use of the document or its content, except for a non-exclusive license to consult it on an "as is" basis, at your own risk. All other rights are reserved.

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.

© 2018 Schneider Electric. All rights reserved.

Table of Contents

	Safety Information	7
	About the Book	9
Part I	**Introduction to Communication by Fipio Bus**	11
Chapter 1	**Communication by Fipio bus**	13
	The Fipio Bus	14
	Examples of architecture	16
Part II	**Hardware Implementation for Fipio Bus**	19
Chapter 2	**Introduction to and implementation of Fipio bus devices**	21
2.1	Introduction to Fipio bus architectures	22
	Architecture of Fipio bus with connection auxiliaries (IP20)	23
	Architecture of Fipio Bus with Connection Auxiliaries (IP65)	24
	Architecture of Fipio bus with Connection Auxiliaries (IP67)	26
	Fipio Architecture with OZD FIP G3	29
2.2	Master Fipio communication built into processors	30
	Built-in Fipio Link on Premium/Atrium Processors	30
2.3	TSX FPP 10 PCMCIA card	32
	Introduction	33
	Description	35
	Connection of the TSX FPP 10 card	36
	Precautions to be taken when connecting PCMCIA card	37
	Installation	38
	Mounting Cards and Cables	39
	Displaying operating state of PCMCIA cards	42
	Visual Diagnostics for TSX FPP 10 PCMCIA Card	43
	Consumption of PCMCIA card	44
2.4	TSX FPC 10 module	45
	Description	45
2.5	TSX FPP 20 PCMCIA card	46
	Description	46
Part III	**Software Implementation for Fipio Bus**	47
Chapter 3	**Presentation and Specifications of the Fipio Bus**	49
3.1	Introduction to Fipio communication	50
	Addressing of Remote Module Language Objects on a FIPIO Bus	50

3.2 Characteristics 53
Hardware Compatibilities relating to the Bus Manager 54
Software Compatibilities relating to the Bus Manager 55
Software Compatibility: Network Transparency 56
Software Compatibility: Outside Communication 59
TBX Device Compatibilities 61
Momentum Device Compatibilities 63
Programming Terminal Compatibilities 65
Miscellaneous Device Compatibilities 66
Fipio Bus Operating Mode 69
Saturation of Fipio Messaging Service 71
Network Cycle Time: Mono Task Application 72
Network Cycle Time: Multitasking Application 74
Example of calculating the network cycle times of a multitasking application 77
Chapter 4 Installation methodology 79
Overview 79
Chapter 5 Configuring Fipio communication 81
5.1 General information on configuration 82
Configuration Mode 82
5.2 Fipio bus configuration 83
How to Access the Fipio Bus Configuration Screen 84
Fipio Bus Configuration Screen 85
How to Add a Device on the Bus 87
How to Add an Extension Module on the Bus 89
How to Delete/Move/ Duplicate a Bus Device 91
Changing the Fipio Communicator Type 95
Fipio Bus Displayed in the Project Browser 96
How to Access the Properties Screen of the Fipio Bus 97
Properties of the Fipio Bus: General Tab 99
Properties of the Fipio Bus: Expert tab 100
Expert Tab: Manual Mode 101
5.3 Configuration of devices on the Fipio bus 103
How to Access the Configuration Screen for a Fipio Device 104
Configuration Screen of a Fipio Device 105
How to Access Different Parameters Depending on the Type of Device 107

5.4 Management of Inputs/Outputs by Control Expert Tasks: 108
How to Access the Configuration of Control Expert Tasks in Fipio Mode 109
Management of Inputs/Outputs by Control Expert Tasks 110
5.5 Confirmation of the Fipio bus configuration 112
Refused Build 113
Examples of Validation Refusals 114
Chapter 6 Programming Fipio communication 117
Fipio Fault Processing 118
Examples of Detection of Explicit Exchange Saturation 119
Chapter 7 Debugging a Fipio communication 125
Debug Mode 126
How to Access the Debug Screens of Remote Devices 127
Debug Screen for a Device on the Fipio Bus 128
Chapter 8 Fipio communication diagnostics 131
Diagnostics Mode 132
How to Access the Fipio Bus Monitoring Screen. 133
Diagnostics Screen: Fipio Bus Monitoring. 134
How to Access the Device Monitoring Screen. 136
Diagnostics Screen: Device Monitoring. 137
How to Access the Device Monitoring Screen for a Single Device . . . 139
Diagnostics Screen: Device Monitoring. 140
How to Access the Communication Faults History Screen 142
Diagnostics Screen: Communication Errors History 143
Chapter 9 Language objects associated with Fipio communication 147
9.1 Language Objects and IODDTs for Fipio Communication 148
Presentation of Language Objects for Fipio Communication 149
Implicit Exchange Language Objects Associated with the Application-Specific Function 151
Explicit Exchange Language Objects Associated with the Application-Specific Function 152
Management of Exchanges and Reports with Explicit Objects 154
Details of Implicit Exchange Objects for Fipio Communication 158
9.2 The IODDT Type T_GEN_MOD Applicable to All Modules 159
Details of the Language Objects for the IODDT of Type T_GEN_MOD 159
Chapter 10 Fipio communication standard profiles 161
10.1 Introduction to standard Fipio communication profiles 162
Standard profiles: general 163
Designation of a Standard Profile 164
List of Catalog Standard Profiles Proposed by Control Expert 165

10.2 Configuration of the Fipio bus standard profiles . . . 167
Configuration Screen for a Standard Profile . . . 168
How to Modify the Parameters of a Standard Profile . . . 170
Adjustment Mode . . . 172
10.3 Debugging the Fipio bus standard profiles . . . 173
Debug Screen for a Standard Profile . . . 174
How to Modify the Debug Parameters for a Standard Profile . . . 176
10.4 Language objects associated with the Fipio bus standard profiles . . . 178
Details of the Language Objects for the IODDT of Type T_STDP_GEN . . . 179
Objects Accessible by Standard Profiles . . . 181
Implicit Exchange Language Objects . . . 183
Explicit Exchange Language Objects . . . 184
Language Objects Associated with Configuration . . . 186
Chapter 11 Fipio Agent . . . 187
11.1 Introduction to Fipio Agents . . . 188
Fipio Agent: General . . . 189
Periodic Data Exchange . . . 190
Special Cases . . . 191
Communication from a Fipio Agent . . . 192
11.2 Configuration of Fipio Agents . . . 194
How to Fipio Agent Parameters Using PCMCIA Cards . . . 195
Configuration Screen of a Fipio Agent . . . 197
11.3 Debugging Fipio Agents . . . 199
Debug Screen of a Fipio Agent . . . 199
11.4 Language objects associated with the Fipio Agents . . . 201
Implicit Exchange Language Objects . . . 202
Explicit Exchange Language Objects . . . 203
Explicit Exchange Management and Report . . . 204
Language Objects Associated with Configuration . . . 205
Index . . . 207

Safety Information

Important Information

NOTICE

Read these instructions carefully, and look at the equipment to become familiar with the device before trying to install, operate, service, 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 a hazardous situation which, if not avoided, **will result in** death or serious injury.

⚠ WARNING

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

⚠ CAUTION

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

NOTICE

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

PLEASE NOTE

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

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

About the Book

At a Glance

Document Scope

This manual describes the implementation of hardware and software for Fipio communication with Premium and Atrium PLCs.

Validity Note

This documentation is valid for EcoStruxure™ Control Expert 14.0 or later.

The technical characteristics of the devices described in the present document also appear online. To access the information online:

Step	Action
1	Go to the Schneider Electric home page *www.schneider-electric.com*.
2	In the **Search** box type the reference of a product or the name of a product range. ● Do not include blank spaces in the reference or product range. ● To get information on grouping similar modules, use asterisks (*).
3	If you entered a reference, go to the **Product Datasheets** search results and click on the reference that interests you. If you entered the name of a product range, go to the **Product Ranges** search results and click on the product range that interests you.
4	If more than one reference appears in the **Products** search results, click on the reference that interests you.
5	Depending on the size of your screen, you may need to scroll down to see the data sheet.
6	To save or print a data sheet as a .pdf file, click **Download XXX product datasheet**.

The characteristics that are presented in the present document should be the same as those characteristics that appear online. In line with our policy of constant improvement, we may revise content over time to improve clarity and accuracy. If you see a difference between the document and online information, use the online information as your reference.

Related Documents

Title of Documentation	Reference Number
Communication Services and Architectures, Reference Manual	35010500 (English), 35010501 (French), 35006176 (German), 35013966 (Italian), 35006177 (Spanish), 35012196 (Chinese)
EcoStruxure™ Control Expert, System Bits and Words, Reference Manual	EIO0000002135 (English), EIO0000002136 (French), EIO0000002137 (German), EIO0000002138 (Italian), EIO0000002139 (Spanish), EIO0000002140 (Chinese)

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

Product Related Information

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

The application of this product requires expertise in the design and programming of control systems. Only persons with such expertise should be allowed to program, install, alter, and apply this product.

Follow all local and national safety codes and standards.

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

Part I
Introduction to Communication by Fipio Bus

Chapter 1
Communication by Fipio bus

Subject of this Chapter

This Chapter introduces some general points concerning communication by Fipio bus.

What Is in This Chapter?

This chapter contains the following topics:

Topic	Page
The Fipio Bus	14
Examples of architecture	16

The Fipio Bus

General

Fipio is a field bus which allows the decentralization of the inputs/outputs of a PLC station and its industrial peripheral devices nearest to the operational part.

Fipio bus is part of the WorldFIP product range from Schneider Automation S.A.S.

From a PLC station whose processor has a built-in Fipio link, the Fipio bus is used to connect 1 to 127 devices such as:

- Momentum remote input/output modules (discrete, analog and counting modules),
- TBX IP20 remote input/output modules (discrete and analog modules),
- TBX IP65 and IP67 remote input/output modules (discrete modules),
- XBT-F terminals,
- Lexium, ATV 58 variable speed controllers,
- Devices which conform to standard profiles,
- Agent PLCs, PCs,
- Inductel stations,
- An AS-i gateway.

The Fipio field bus can be used in a single architecture (mono-station) or in a more complex architecture (multi-station) where several Fipio segments can be brought together by a local network at a higher level such as Fipway or Ethernet TCP_IP for example.

The Fipio protocol uses producer/consumer type exchanges and the bus is managed by a bus arbiter.

Manager PLC

A PLC station which has a built-in Fipio link is called a manager PLC *(see page 30)*.

Agent PLC

A PLC station with a PCMCIA card installed in a host slot is called an agent PLC *(see page 32)*.

Main Characteristics

Structure	
Nature	Open field bus, conforming to World FIP and FIP standards.
Topology	Devices linked by chaining or branching.
Access method	Managed by a bus arbiter
Communication	By exchange of variables which can be accessed by the user in the form of Control Expert objects and X-Way datagrams.
Privileged exchanges	Cyclical exchange of status variables and remote input/output commands

Transmission	
Binary rate	1 Mb/s.
Medium	Shielded twisted pair (150 Ohms of characteristic impedance) or fiber optic.

Configuration	
Number of connection points	128 logic connection points for whole architecture
Number of segments	15 maximum (in cascade format) using electrical or optical repeaters (14 maximum in cascade format).
Number of repeaters	An electrical or optical repeater is required every 1000 meters or every 32 devices.
PLC	One manager PLC (connection point address 0).
Programming terminal	One programming terminal (must be connected to connection point 63).
Length	The length of a segment depends on its type of branches: ● 1000 meters maximum without repeater. ● 1500 meters maximum between the devices which are the furthest apart.

⚠ WARNING

UNEXPECTED EQUIPMENT OPERATION

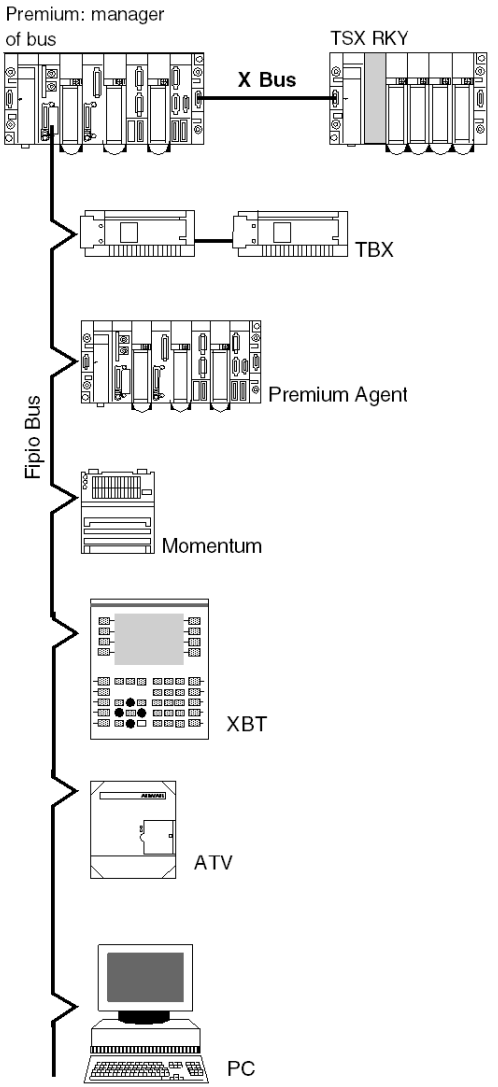
You are advised against connecting/disconnecting elements from the Fipio bus while powered up as certain devices may then not restart.

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

Examples of architecture

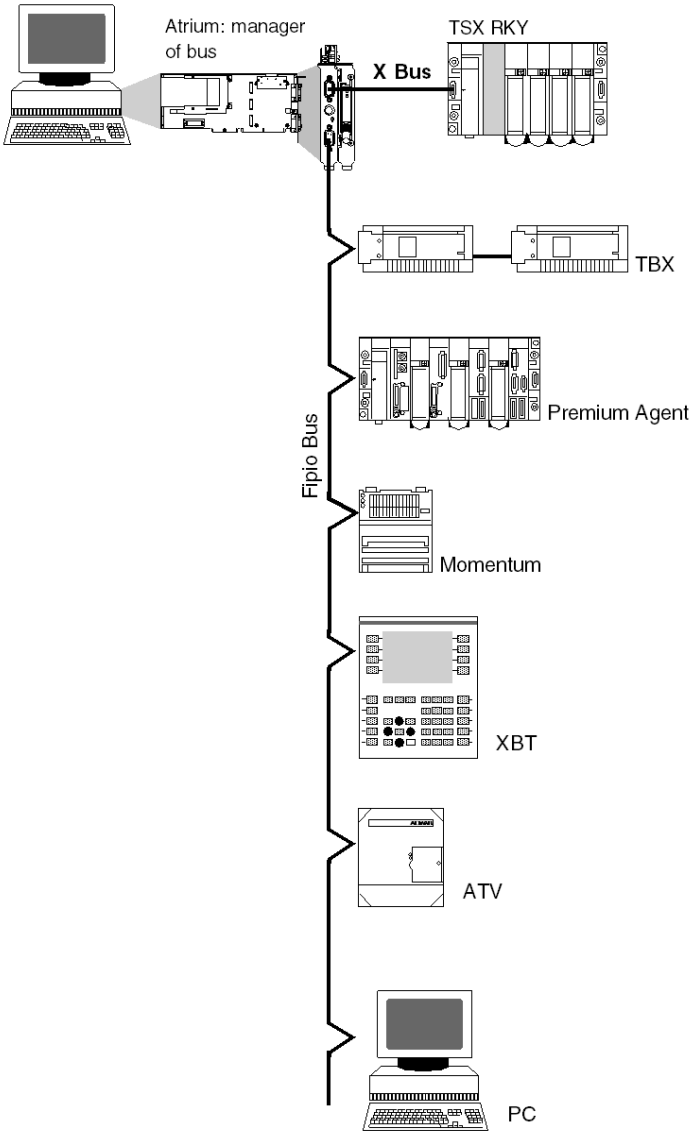
Premium station

Illustration:



Atrium station

Illustration:



Part II
Hardware Implementation for Fipio Bus

Chapter 2
Introduction to and implementation of Fipio bus devices

Subject of this Chapter

This Chapter introduces the different Fipio bus architectures and the implementation of devices.

What Is in This Chapter?

This chapter contains the following sections:

Section	Topic	Page
2.1	Introduction to Fipio bus architectures	22
2.2	Master Fipio communication built into processors	30
2.3	TSX FPP 10 PCMCIA card	32
2.4	TSX FPC 10 module	45
2.5	TSX FPP 20 PCMCIA card	46

Section 2.1
Introduction to Fipio bus architectures

Subject of this Section

This Section introduces some different Fipio bus architectures.

What Is in This Section?

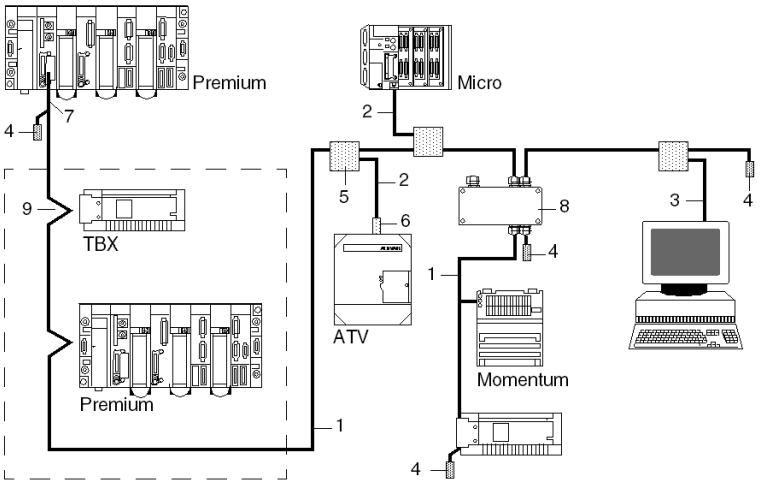
This section contains the following topics:

Topic	Page
Architecture of Fipio bus with connection auxiliaries (IP20)	23
Architecture of Fipio Bus with Connection Auxiliaries (IP65)	24
Architecture of Fipio bus with Connection Auxiliaries (IP67)	26
Fipio Architecture with OZD FIP G3	29

Architecture of Fipio bus with connection auxiliaries (IP20)

At a Glance

The architecture below shows the Fipio bus and the different devices (IP20) of which it is composed.



Description

The following table shows the devices on the bus:

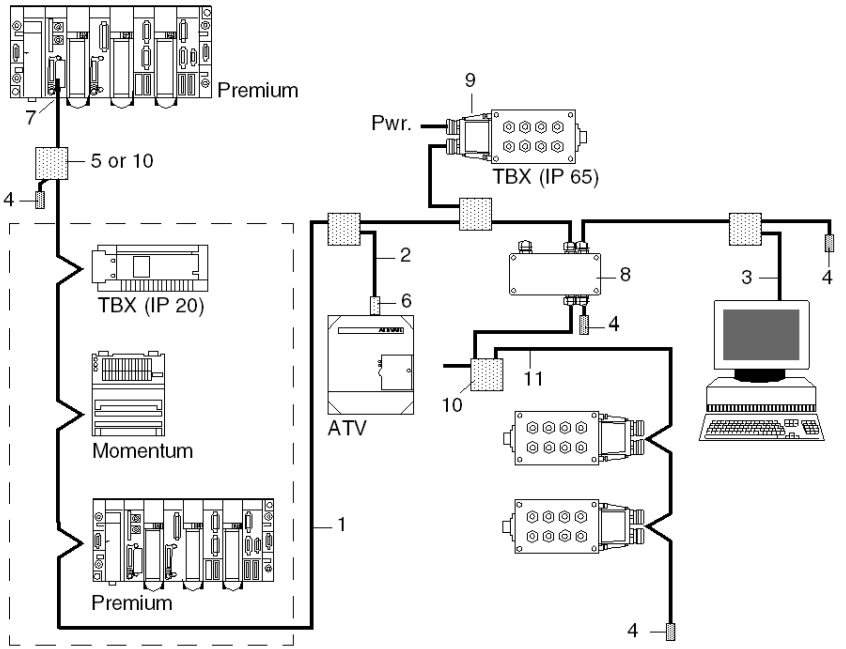
Number	Product reference	Description
1	TSX FP CA/CRxxx	Trunk cable
2	TSX FP CCxxx	Drop cable
3	TSX FP CE030	Connection cable for terminals and PCs
4	TSX FP ACC7	Line terminator
5	TSX FP ACC3 or TSX FP ACC4 or TSX FP ACC14	Branching devices
6	TSX FP ACC2	Connector for daisy-chaining or branching
7	TSX FP ACC12	Connector for Premium PLC
8	TSX FP ACC6 TSX FP ACC8	Electrical repeater or fiber optic repeater if fiber optic link used
9	TBX BLP 01	Connector for connecting TBXs

The Fipio reference manual (TSX DR FIP) specifies the characteristics of each device.

Architecture of Fipio Bus with Connection Auxiliaries (IP65)

At a Glance

The architecture below shows the Fipio bus and the different devices (IP65) of which it is composed.



Description

The following table shows the devices on the bus:

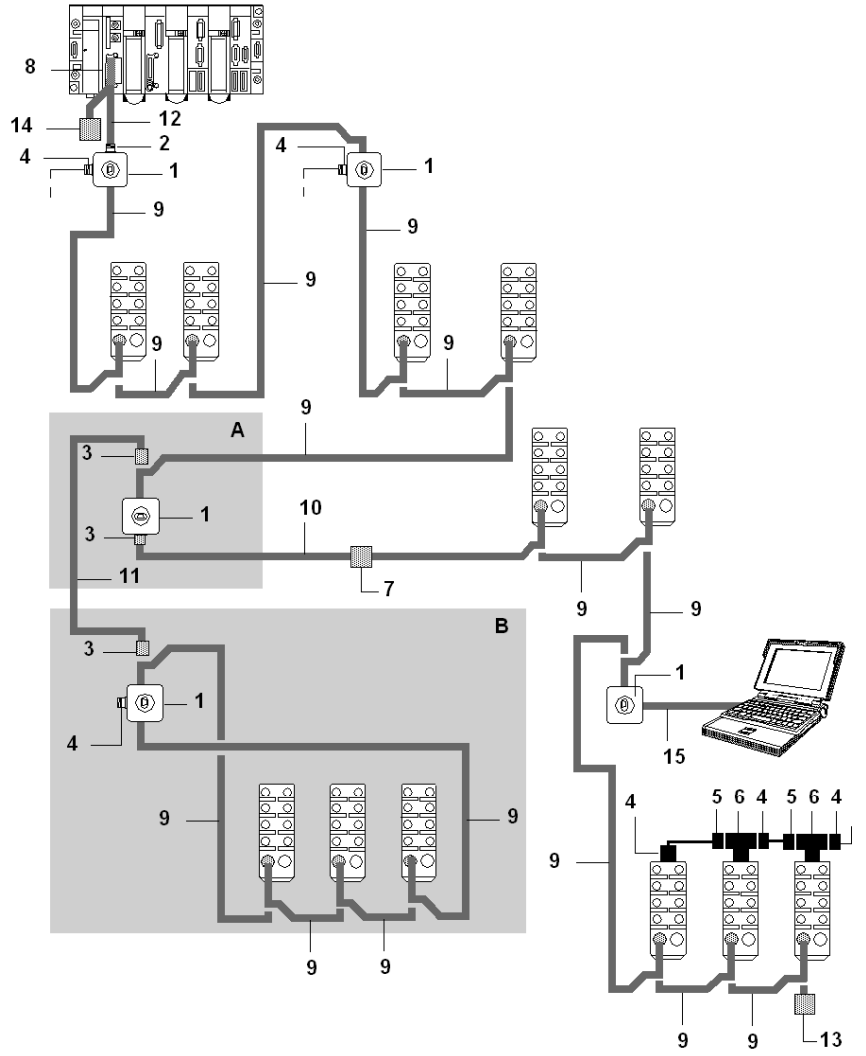
Number	Reference	Description
1	TSX FP CA/CR•••	Trunk cable
2	TSX FP CC•••	Branch cable
3	TSX FP CE030	Connection cable for terminals and PCs
4	TSX FP ACC7	Line terminator
5	TSX FP ACC4	Branching device
6	TSX FP ACC2	Connector for daisy-chaining or branching
7	TSX FP ACC12	Connector for Premium PLC
8	TSX FP ACC6 TSX FP ACC8	Electrical repeater or fiber optic repeater if fiber optic link used
9	TBX BLP 10	Connector for connecting IP 65 TBXs
10	TBX FP ACC10	Branching device
11	TSX FP CF•••	Remote power supply cable

The Fipio reference manual (TSX DR FIP) specifies the characteristics of each device.

Architecture of Fipio bus with Connection Auxiliaries (IP67)

Illustration

The diagram below shows the possible architectures which may be used to connect IP67 sealed I/O modules, as well as the accessories used, to the Fipio bus.



A Branch head.
B IP67 branch.

Description

The following table presents the various accessories that make it possible to assemble the desired configuration whilst observing all the operating constraints of the Fipio bus.

Number	Part Number	Type of connection	Description
1	**TSX EF ACC 99**	Branch box.	This IP65 branch box can be used to perform 3 functions: ● Connection of a 24 VDC power supply, ● Creation of an IP67 branch, ● Connection of a programming terminal.
2	**TSX EF CF 01**	IP67 female connector, type M23.	IP67 female solder connector with M23-type ring, 6 contacts for Fipio bus connection and 24 VDC power supply. This type of independent connector is designed to be fitted to one end of **TSX FP CA/CC/CP•00/CR** cables.
3	**TSX EF CM 01**	IP67 male connector, type M23.	IP67 male solder connector with M23-type ring, 6 contacts for Fipio bus connection and 24 VDC power supply. This type of independent connector is designed to be fitted to one end of **TSX FP CA/CC/CP•00/CR** cables.
4	**TSX EF CF 03**	IP67 female connector, type 7/8" (24 VDC power supply).	7/8" type IP67 female screw connector, 5 contacts for Fipio bus connection and 24 VDC power supply. This type of connector is designed to be used for connecting the 24 VDC power supply of **TSX ESF/EMF** output or mixed modules and/or of the **TSX EF ACC 99** branch box. The maximum authorized current in each contact on the connector is 8 A.
5	**TSX EF CM 03**	IP67 male connector, type 7/8" (24 VDC power supply).	7/8" type IP67 male screw connector, 5 contacts for linked connection using the **TSX EF CT 03** T connector, designed to supply the preactuators with 24 VDC.
6	**TSX EF CT 03**	T connector (for linking 24 VDC power supply).	7/8" type IP67 T connector, 5 contacts for linking the 24 V power supply of the preactuators. The maximum authorized current for each contact on this connector is 8 A.
7	**TSX EF CF 02**	IP67 female connector, type M23.	M23, type IP67 female threaded solder connector, 6 contacts for extending Fipio **TSX FP CA/CC/CP•00/CR** cables.
8	**TSX FP ACC 2** or **TSX FP ACC 12**	SUB-D 9 type female connector.	SUB-D type connectors with 9 contacts for Fipio/Fipway connection to PLCs. Can be used for linked or branched connection (90° high or low output, 45° high or low output).
9	**TSX EF ACC 2•••** and **TSX EF ACC 2••••**	IP67 compound-filled Fipio cable for linking of modules.	Main compound-filled cable of a predefined length, 150 Ω twisted shielded pair, and a 1.5 mm^2 power supply pair. Equipped with an M23 male connector and an M23 double female/female connector.

Number	Part Number	Type of connection	Description
10	**TSX FP CP •00**	IP67 Fipio cable (1 cable pair, 2 1.5 mm^2 power supply conductors, in 100 m or 500 m).	Main IP67, Fipio cable, 150 Ω twisted shielded pair, and a 1.5 mm^2 power supply pair (Ø 9.5 mm). Free wire ends to be fitted with IP 67 M23 type **TSX EF CF 01/02** and **TSX EF CM 01** solder connectors.
11	**TSX FP CC •00**	Fipio branch cable (2 pairs, in 100 m or 500 m).	Branch cable, 150 Ω double shielded twisted pair (Ø 8 mm) for standard environments and building interiors. Free wire ends to be fitted with IP67 M23 type **TSX CF 01/02** and **TSX CM 01** soldered connectors.
12	**TSX FP CA •00**	Fipio link cable (1 pair, in 100 m, 200 m or 500 m).	Main cable, one 150 Ω shielded twisted pair (Ø 8 mm) for standard environments and building interiors. Free wire ends to be fitted with IP 67 M23 type **TSX EF CF 01/02** and **TSX EF CM 01** solder connectors.
13	**TSX EF ACC 7**	IP67 M23 type line terminator.	IP67 M23 type line terminator, to be positioned, depending on use, at one or both ends of the section (male type for direct connection).
14	**TSX FP ACC 7**	Line terminator	IP20 line terminator (batch of 2), to be positioned, depending on use, at one or both ends of the section.
15	**TSX FP CG 0•0**	PLC or programming terminal link cable	Branch connection cable for **TSX FP 10/20 PCMCIA** interface card for Premium, FT 2000 terminal and PC-compatible PLCs. This cable allows a PLC or a programming terminal to be connected to a **TSX EF ACC 99** branch box.

Other Accessories

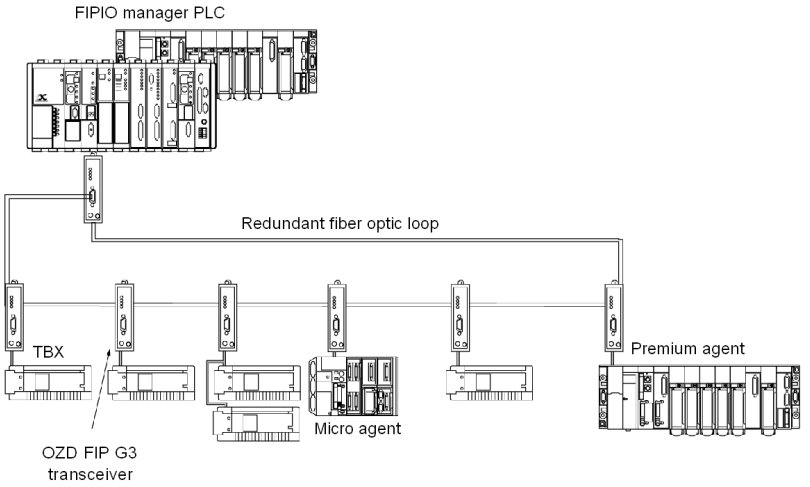
The following table shows the other accessories available.

Part Number	Type of connection	Description
XZ-LG 101	Identification labels.	Strip of 10 additional identification labels (a strip is already supplied with each module).
XZ-LG 102	IP67 sealing plugs.	Batch of 10 M12 sealing plugs for unused I/O connectors. Each module is supplied with two plugs.
170 XTS 050 00	IP67 sealing plug.	Sealing plug for M23 connector.

Fipio Architecture with OZD FIP G3

Description

Example of a Fipio architecture:



Most Schneider products which operate on the Fipio bus can be connected to the OZD FIP G3 transceiver:

- As Fipio manager devices:
  - Premium: processors with an integrated FIPIO link: TSX P57•54 and TSX PCXI57 354 programmed with Control Expert (version V1.1 or later).
- As Fipio bus agent devices:
  - TBX equipped with an LEP 030 communication module
  - Magelis XBT equipped with a TSX FPP 10 card
  - PC equipped with a TSX FPC 10 or PCMCIA FPP20 ISA card
  - Momentum
  - TSX Micro TSX 372• and TSX Premium TSX/PMX/PCX 57 equipped with a TSX FPP 010 card
  - Third-party devices which conform to the standard Fipio profiles defined in the FipConnect connection program.

NOTE: TSX FP ACC6 and TSX FP ACC8M repeaters are not compatible with architectures using OZD FIP G3 transceivers.

NOTE: Fipio configurations in WorldFip mode are excluded from these architectures.

NOTE: The TSX DM OZD 01 manual gives all the characteristics of HIRSCHMANN OZD FIP G3 transceivers.

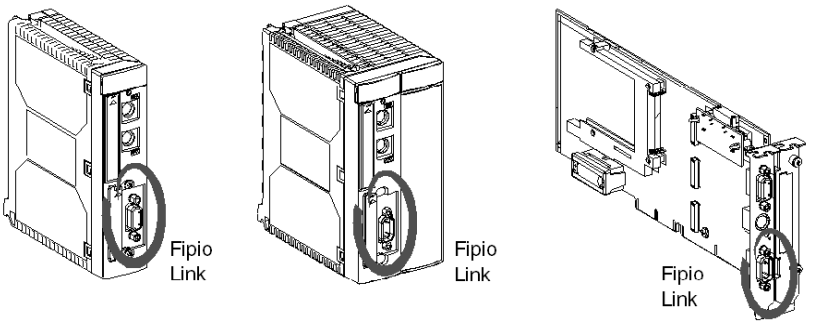
Section 2.2
Master Fipio communication built into processors

Built-in Fipio Link on Premium/Atrium Processors

General

Some Premium *(see Premium and Atrium using EcoStruxure™ Control Expert, Processors, racks and power supply modules, Implementation Manual)* and Atrium *(see Premium and Atrium using EcoStruxure™ Control Expert, Processors, racks and power supply modules, Implementation Manual)* processors have a built-in master Fipio link which makes it possible to connect the PLC station to a Fipio bus.

Illustration:



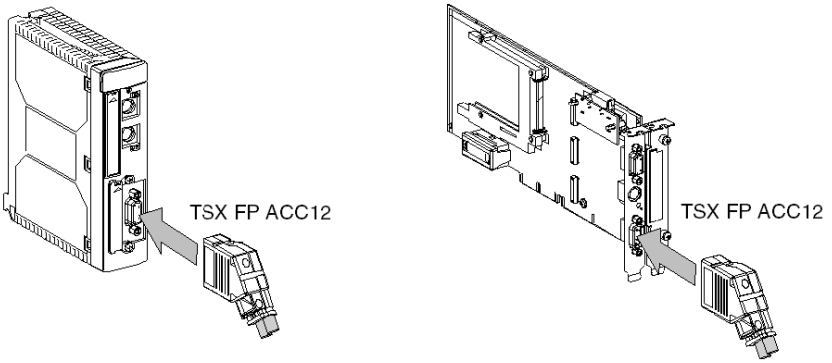
Premium processors

Atrium processor

Connecting to Fipio Bus

The processor has a SUB-D 9-pin connector which is used to connect it to the Fipio bus using a **TSX FP ACC12** connector.

Illustration:



NOTE: The master Fipio link built into processors should not be taken into account when counting the station channels.

Section 2.3
TSX FPP 10 PCMCIA card

Subject of this Section

This Section deals with the installation of a TSX FPP 10 PCMCIA card.

What Is in This Section?

This section contains the following topics:

Topic	Page
Introduction	33
Description	35
Connection of the TSX FPP 10 card	36
Precautions to be taken when connecting PCMCIA card	37
Installation	38
Mounting Cards and Cables	39
Displaying operating state of PCMCIA cards	42
Visual Diagnostics for TSX FPP 10 PCMCIA Card	43
Consumption of PCMCIA card	44

Introduction

General

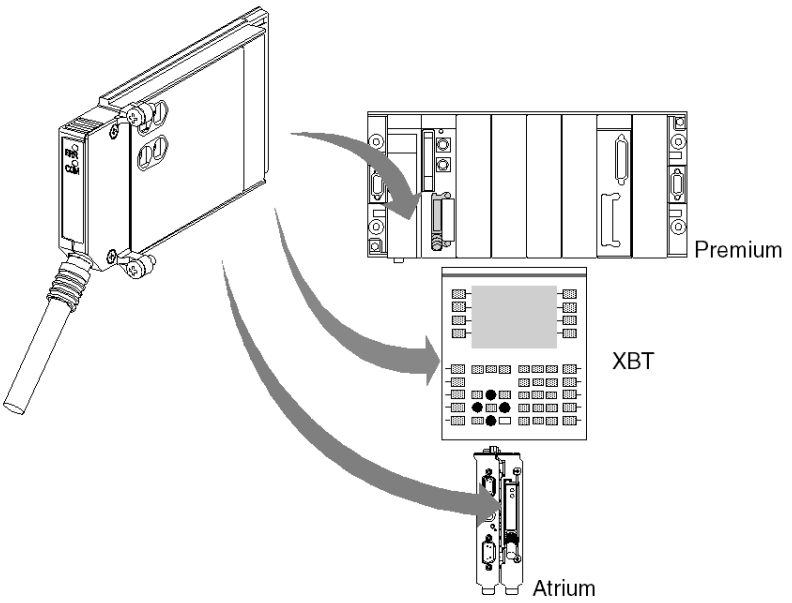
Premium/Atrium PLC stations connect to communication networks, buses and links via PCMCIA communication cards.

The card to be connected is a metal unit whose dimensions comply with PCMCIA extended type III format.

The PCMCIA card is installed in the host slot of the processor of Premium PLCs.

The PCMCIA card can also be used in devices which have a type III host slot, such as **XBT** terminals or third-party devices, such as PC compatible devices, for example.

Illustration:



NOTE: It is prohibited to connect PCMCIA cards when the power is switched on.

Definition

An agent PLC is a PLC with a connected PCMCIA card.

TSX FPP 10 card

The PCMCIA card used to connect a Premium/Atrium PLC station to a Fipio bus is the **TSX FPP 10**.

The PCMCIA card is installed, operated and maintained using Control Expert programming and operation software for all PLCs in the Premium/Atrium family.

Description

General

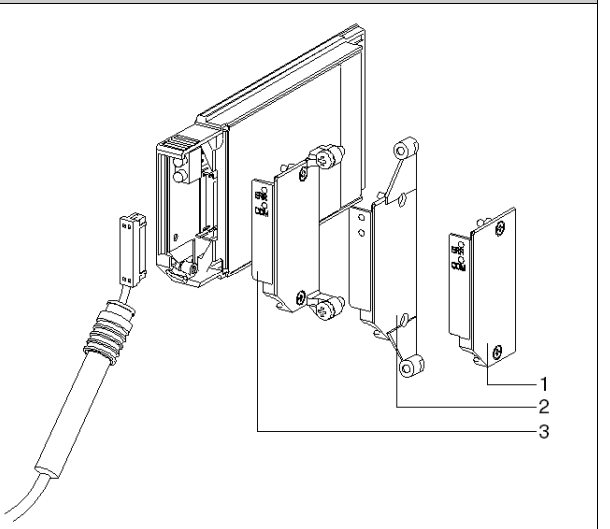
PCMCIA type III (extended) communication cards are built into a metal unit with the following dimensions:

- Length: 85.5 mm
- Width: 51 mm
- Height: 10 mm

The front of the card is designed to display the operating status of communications as well as the physical connection to the network.

Mechanical configuration

The mechanical configuration of the card must be adapted by mounting a removable cover, depending on the type of installation desired:

Type of installation	Configuration	Illustration
Installation on a Premium type processor.	Removable cover with wings. Screws are provided to fix it to the host module (marked 3 on illustration)	
Installation on an Atrium type processor.	Removable cover with wings. Screws are provided to fix it to the Atrium processor (marked 2 on illustration).	
Installation on a PC compatible device.	Removable cover (marked 1 on illustration).	

NOTE: The covers with wings, mounted on PCMCIA cards, prevent any accidental removal when switched on and guarantee that the card remains in good working order.
The two covers (1) and (3) are provided with the PCMCIA card. The cover (2) is provided with the Atrium processor.

Connection to the network is achieved by connecting the link cable to the front of the card. A guidance system is used to prevent anything being mounted incorrectly.

The product reference label informs the user of the type of physical layer supported by the card.

Connection of the TSX FPP 10 card

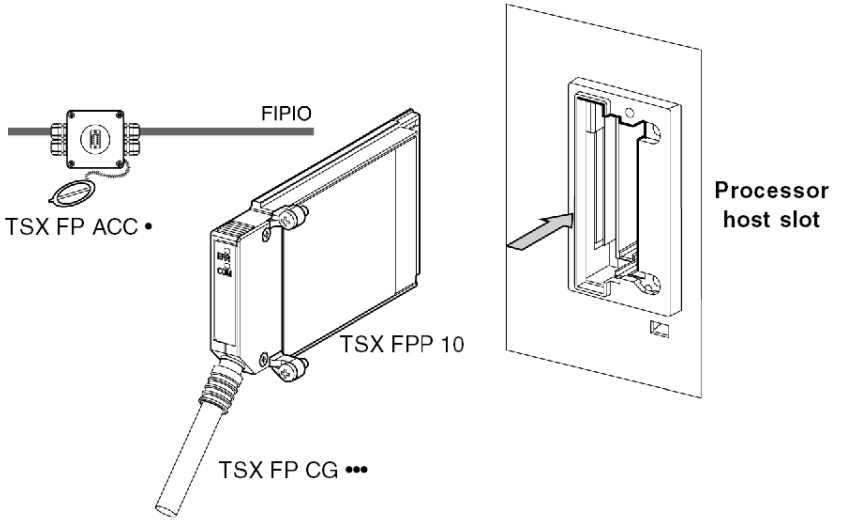
General

The **TSX FPP 10** PCMCIA card is connected to the Fipway bus using a connector such as a **TSX FP ACC 3** or **TSX FP ACC 4**.

To connect the PCMCIA card to the ACC3/ACC4 connector the user may use:

- Either a 1 m cable, product reference **TSX FP CG 010**.
- Or a 3 m cable, product reference **TSX FP CG 030**.

Items required for connecting a Premium/Atrium PLC to the remote inputs/outputs Fipio bus:



Precautions to be taken when connecting PCMCIA card

General

NOTICE
PCMCIA CARD DESTRUCTION Cables (**TSX FP CG 010/030**) are to be connected and disconnected from the PCMCIA card only **when the power is switched off** **Failure to follow these instructions can result in equipment damage.**

⚠ CAUTION
PCMCIA CARD DESTRUCTION Power off the module before inserting or removing the PCMCIA card. **Failure to follow these instructions can result in injury or equipment damage.**

When removing or inserting the card, the unit is not guaranteed to be operational.

The PCMCIA card must be equipped with a PLC version cover and be screwed into the reception module before the power is switched on (see *Mechanical configuration, page 35*).

The ferrule, placed in direct contact with the PCMCIA card device, is used to drain electrical interference carried by the link cable braids.

Installation

Installation

The **TSX FPP 10** card can only be installed in the host slot of processors. It cannot be installed in the module **TSX SCM 21601**

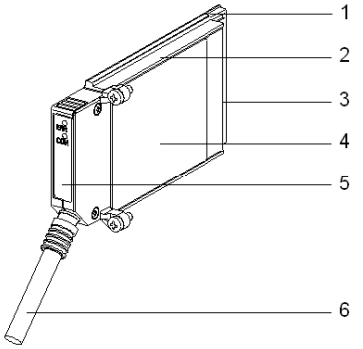
Application-specific channel and network connection

The installation of the card in the processor does not count as an application-specific channel or as a network connection.

Mounting Cards and Cables

PCMCIA Card Details

Illustration:

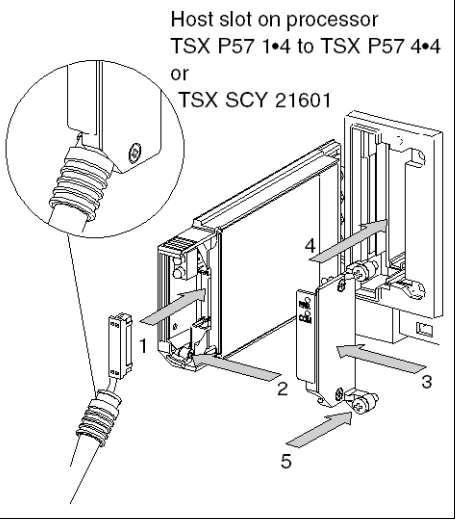


PCMCIA cards are made up of the following elements:

Number	Designation	Comments
1	Equipped card	Receives electronic components.
2	Body made of zamac	-
3	PCMCIA connector	Connector with 20 connection points.
4	Upper cover	Houses the product reference label which shows the type of PCMCIA card
5	Removable cover	Ensures the card is displayed in its slot. The names of the two LEDs are printed on the front of the removable cover. This cover also allows the PCMCIA card to be fastened to the processor.
6	Link cable with ferrule	The ferrule placed on the end of the cable nearest to the PCMCIA card prevents the cable being pinched by the removable cover. This ferule also eliminates the risk of causing a bending radius which can damage the quality of the link.

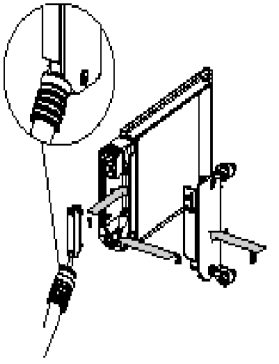
Assembly for TSX P57 1•4 to TSX P57 5•4 Processors

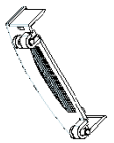
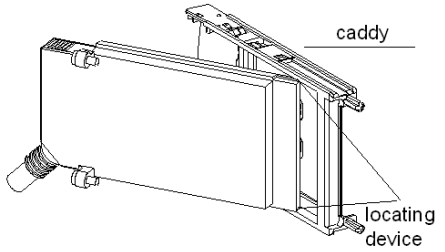
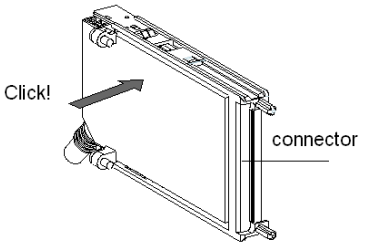
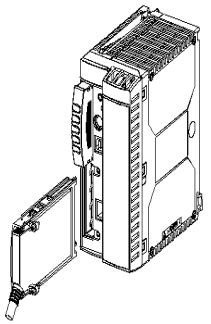
To assemble the transmission card for the processor or the TSX SCY 21601, first remove the cover which is screwed on the device and then follow the instructions below:

Step	Action	Illustration
1	Connect the cable	
2	Place the appropriate cover on the device, taking care to insert the ferrule in the slot provided in order to fix the cable to the card.	
3	Screw on the cover.	
4	Insert the card in the slot provided in the host device.	
5	Screw in the card to stop it being moved when switched on, and to ensure it functions effectively.	

Assembly for TSX P57 5•4 Processors

To assemble the card in TSX P57 5•4 type processors, follow the instructions below:

Step	Action	Illustration
1	Connect the cable.	
2	Place the appropriate cover on the device, taking care to insert the ferrule in the slot provided in order to fix the cable to the card.	
3	Screw on the cover.	

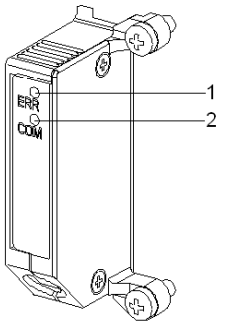
Step	Action	Illustration
4	Remove the plastic cover from the caddy.	
5	Guide the card into the caddy from an oblique angle using the 2 locating devices.	
6	Slide the card into the caddy until it stops. The card is now firmly attached to the caddy.	
7	Insert the assembly (caddy and card) in the slot provided in the host device.	
8	Screw in the card to stop it being moved when switched on, and to ensure it functions effectively.	

Displaying operating state of PCMCIA cards

General

Two diagnostics LEDs are located on the front of the card. They inform the user on the status of exchanges between the device supporting the PCMCIA card and related devices.

Illustration

Number	Description	Diagram
1	"ERR" Error LED (normally off) displays errors. It is red	
2	The "COM" communication LED displays line activity. This LED is yellow	

Visual Diagnostics for TSX FPP 10 PCMCIA Card

General

Depending on their state, the LEDs of the PCMCIA card indicate the operating mode for communication, as well as the card diagnostics.

TSX FPP 10 card

State of LEDs:

ERR	COM	Meaning	Corrective actions
○	○	Device switched off No dialog	Check supply, card inoperational
○	▒	Normal operation	-
●	(1)	Serious fault	Change the card
▒	○	Functional fault	Check the configuration and the connection to the communication bus
▒	▒	Functional fault	Check the configuration
Key:			
○ Off ● On ▒ Flashing (1) = Indifferent state			

Consumption of PCMCIA card

TSX FPP 10

Table of consumption:

Voltage	Typical Current	Maximum Current	Power dissipation
5 volts	280 mA	330 mA	1.65 W max.

Section 2.4
TSX FPC 10 module

Description

General

The **TSX FPC 10** module can be used for connecting the following devices to the Fipio bus:

- FTX 517 workstations
- CCX 77/87 supervision terminals
- any machine fitted with an ISA bus

The module is the same shape as a half-format PC extension card.

It is connected to the Fipio bus via a **TSX FP CE030** cable and a **TSX FP ACC3, TSX FP ACC4 or TBX FP ACC10** branching device

Section 2.5
TSX FPP 20 PCMCIA card

Description

General

The **TSX FPP 20** card is used to connect a programming terminal to a PLC station via the Fipio bus.

The card can also be used to connect a Premium/Atrium PLC station to a Fipway network.

The programming terminal must have a type III (extended) PCMCIA slot.

To install the card, see the manual entitled Fipway Communication.

Part III
Software Implementation for Fipio Bus

Subject of this Part

This part presents the principles for configuring and using Fipio bus communication using Control Expert software.

What Is in This Part?

This part contains the following chapters:

Chapter	Chapter Name	Page
3	Presentation and Specifications of the Fipio Bus	49
4	Installation methodology	79
5	Configuring Fipio communication	81
6	Programming Fipio communication	117
7	Debugging a Fipio communication	125
8	Fipio communication diagnostics	131
9	Language objects associated with Fipio communication	147
10	Fipio communication standard profiles	161
11	Fipio Agent	187

Chapter 3
Presentation and Specifications of the Fipio Bus

Subject of this Chapter

This chapter presents communication by Fipio bus and its specifications.

What Is in This Chapter?

This chapter contains the following sections:

Section	Topic	Page
3.1	Introduction to Fipio communication	50
3.2	Characteristics	53

Section 3.1
Introduction to Fipio communication

Addressing of Remote Module Language Objects on a FIPIO Bus

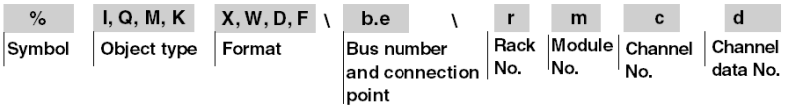
At a Glance

The addressing of the main bit and word objects of remote modules on the Fipio bus is performed on a geographical basis. That means that it depends on:

- The connection point,
- The type of module (base or extended),
- The channel number.

Illustration

Addressing is defined in the following way:



Syntax

The table below describes the different elements that make up addressing.

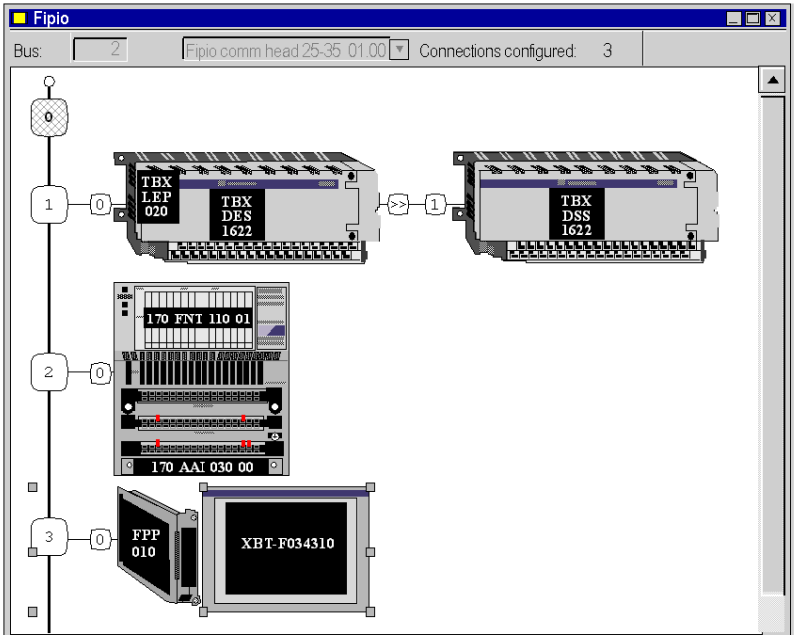
Family	Element	Values	Meaning
Symbol	%	-	-
Object type	I Q	- -	Image of the physical input of the module. Image of the physical output of the module. This information is exchanged automatically for each cycle of the task to which they are attached.
	M	-	Internal variable This read or write information is exchanged at the request of the project.
	K	-	Internal constant This configuration information is available as read only.

Family	Element	Values	Meaning
Format (size)	X	-	Boolean For Boolean objects the X can be omitted.
	W	16 Bit	Single length.
	D	32 Bit	Double length.
	F	32 Bit	Floating point. The floating point format used is the IEEE Std 754-1985 standard (equivalent to IEC 559).
Module/channel address and connection point	b	2	Bus number.
	e	1 to 127	Connection point number.
Rack No.	r	0	Virtual rack number:
Module No.	m	0 or 1	0: base module, 1: extension module.
Channel No.	c	0 to 999 or MOD	MOD: channel reserved for managing the module and parameters common to all the channels.
Channel data No.	d	0 to 999 or ERR	ERR: Used to read a module or channel fault.

Examples

The table below shows some examples of addressing objects.

Object	Meaning
%MW\2.1\0.0.5.2	Status word at rank 2 of the image bit of input 5 of the remote input base module situated at connection point 1 on the Fipio bus.
%I\2.1\0.0.7	Image bit of input 7 of remote input base module situated at connection point 1 on the Fipio bus.
%Q\2.1\0.1.2	Image bit of output 2 of the remote output extension module situated at connection point 1 on the Fipio bus.
%I\2.2\0.0.MOD.ERR	Fault information for Momentum module situated at connection point 2 on the Fipio bus.
%I\2.3\0.0.0.ERR	Fault information for channel 0 of Magelis module situated at connection point 3 on the Fipio bus.



Section 3.2
Characteristics

Aim of this Section

This Section introduces the characteristics and constraints of using Fipio communication.

What Is in This Section?

This section contains the following topics:

Topic	Page
Hardware Compatibilities relating to the Bus Manager	54
Software Compatibilities relating to the Bus Manager	55
Software Compatibility: Network Transparency	56
Software Compatibility: Outside Communication	59
TBX Device Compatibilities	61
Momentum Device Compatibilities	63
Programming Terminal Compatibilities	65
Miscellaneous Device Compatibilities	66
Fipio Bus Operating Mode	69
Saturation of Fipio Messaging Service	71
Network Cycle Time: Mono Task Application	72
Network Cycle Time: Multitasking Application	74
Example of calculating the network cycle times of a multitasking application	77

Hardware Compatibilities relating to the Bus Manager

General

The binary rate of the bus is 1 Mbits/s, with a length that can reach 15 Km with optical or electric repeaters.

The Fipio bus is linked to the Hirschmann optical transmitters, which is used to create medium hot-standby architectures.

Availability

This type of communication is available using built-in links to the Premium PLCs.

Capacity

Processors with a built-in Fipio link capable of managing 128 connection points (limited to 64 for TSX P57 154) on the bus (maximum addresses 0 to 127).

The bus manager is connected to address 0, with address 63 reserved for the programming and diagnostics terminal.

These processors feature a RAM that enables them to save the configuration of the devices linked to the bus.

Software Compatibilities relating to the Bus Manager

Explicit Exchanges Limits

Processors with a built-in Fipio link can be used to simultaneously activate 24 explicit exchange functions.

An exchange request sent to a Fipio device can take several MAST task cycles. It is therefore essential to manage the words of the exchange management parameters for every explicit variable exchange, including `READ_STS(IODDT_VAR1)` and `READ_STS(IODDT_VAR2)`. This requires IODDT-type IODDT_VAR1 and IODDT_VAR2.

If for example 24 functions are being processed simultaneously, or if 25 functions are triggered in a same task cycle, then all new functions (or the 25th) will not be served and will end in error.

For a `READ_STS` exchange, given that the 25th exchange is not possible, the report raises a communication fault in word %MW\2.e\0.m.c.2

For exchanges `WRITE_CMD`, `READ_PARAM`, `WRITE_PARAM`, `RESTORE_PARAM`, the report raises a communication fault in word %MW\2.e\0.m.c.1. A probable cause is the lack of system resources to provide the service.

Reminders

System word %SW155 contains the number of explicit exchanges in progress. Bit %SW116 .2 switches to 1 if the maximum number of simultaneous explicit exchanges is reached or exceeded.

Software Compatibility: Network Transparency

At a Glance

Message exchanges from a remote station to the Fipio bus, or vice-versa, are possible as long as certain conditions are met.

Transparency Restrictions

Inter-station communication to Fipio is only available if the bus arbiter is a Premium-type PLC.

Exchanges from a third-party device to a PLC connected as a Fipio agent are only possible if the PLC is Premium. There is no communication to a Micro-type PLC.

Connection point **63** is a privileged connection point, which does not need to be declared in Control Expert configuration.

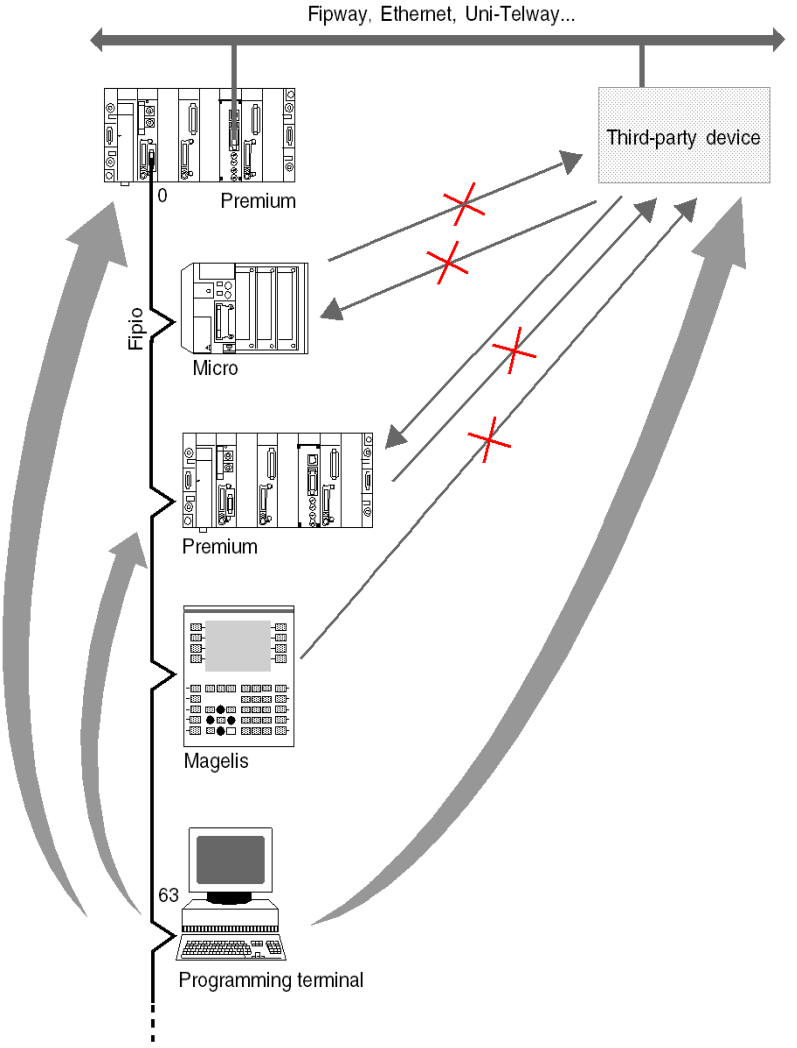
The programming terminal can only be physically connected to connection point 63.

It is impossible to download a project from the programming terminal connected to address 63 in the bus arbiter from the Fipio bus. In fact, downloading causes the PLC to switch to STOP, which makes Fipio communication no longer managed by the bus arbiter.

NOTE: You must not configure managing Fipio communication (built-in link) and Fipio agent communication (PCMCIA TSX FPP 10 card) on the same PLC

Illustration

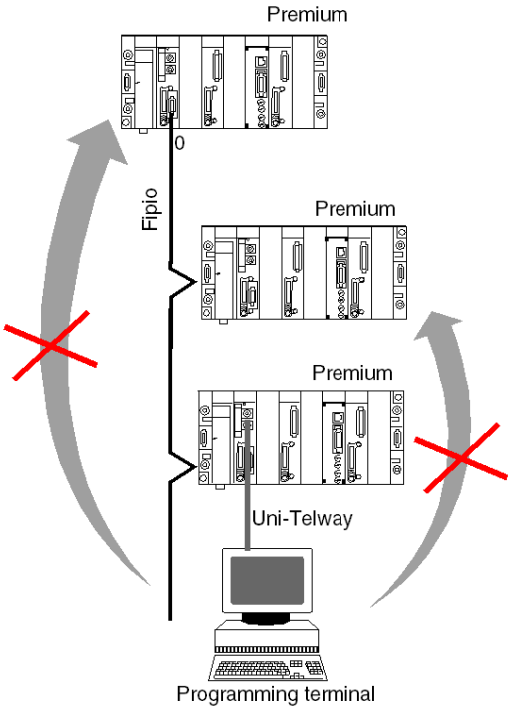
The following illustration shows the different possible exchanges between a Fipio bus, a network or another bus.



In the above example, the micro PLC cannot communicate with the third-party device. Here, you must use the master *(see page 59)* PLC. The programming terminal can access the master and Fipio Agents as well as third-party devices via the master.

Illustration

The following illustration shows the restrictions on exchanges between a terminal connected to the terminal port of a Premium agent.



In the above example, the programming terminal cannot communicate with the managing PLC and the other PLC agents present on the Fipio bus.

Software Compatibility: Outside Communication

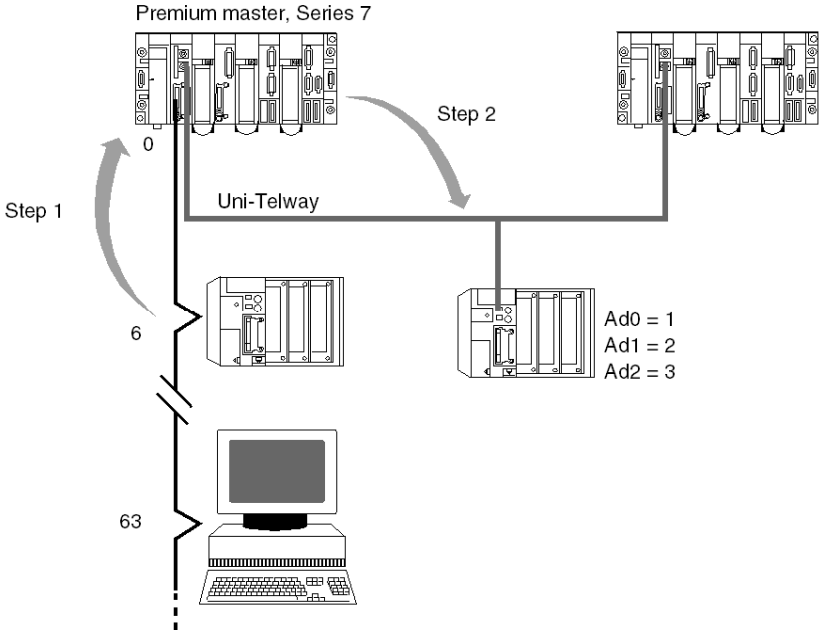
At a Glance

To communicate from one station connected to the Fipio bus to an outside device, the transmitter station must send to the master PLC and not the destination device (requests are not sent directly).

NOTE: The master PLC project transfers requests to the destination device.

Example

The following example shows an exchange from the Fipio bus to the Uni-Telway bus. Exchanges in the other direction use an identical procedure.



Step 1

The transmitter station, connected to Fipio connection point 6, sends the request to the master PLC. The destination address is the address of the master PLC.

```
WRITE_VAR (ADDR('\2.0\SYS'), '%MW',0,10, %MW10:10, %MW100:4)
```

The report is provided by the master PLC.

Step 2

When the master PLC project receives the request from the transmitter station, it sends it to the destination device according to the Uni-Telway protocol.

```
WRITE_VAR (ADDR('0.0.0.1'), '%MW',100,10, %MW10:10, %MW40:4)
```

The report is addressed to the master PLC.

TBX Device Compatibilities

At a Glance

The following compatibility information concerns:

- The address settings supported by the different connectable devices,
- The memory size these devices occupy in the configuration RAM.

Rules of Use

The device connected to a connection point, if it is modular, must have a uniform application: Discrete base and extension, or analog base and extension (no Discrete/analog mix).

The device (base and extension) is controlled by a single FAST or MAST Control Expert task (no group assigning of channel groups of a single TBX to different tasks).

Address Settings

Fipio address settings depend on the type of device:

Type of Device	FIPIO addresses	Comments
Compact TSXs	1...31	TBX C•
Dust and damp proof TBXs	1...62, 64...127	TBX E•
Modular TBXs	1...62, 64...127	TBX LEP •••
AS-i gateway	1...62, 64...127	TBX SAP 10

Memory Space

Each processor *(see page 97)* has a memory space available for Fipio devices. This memory space equals **95744** bytes for TSX P57 154/254/354 processors and **214528** bytes for SX P57 454/554 processors.

Each Fipio device connected to the bus uses part of this memory space. The memory used by the devices connected to the Fipio bus must be less than the processor's available memory. When calculating the memory used, you must add an offset of 1424 bytes.

The following table lists the memory values used in bytes for each TBX module.

Module	Base	Extended	Maximum number of connection points		
			TSX P57 154	TSX P57 254/354	TSX P57 454/554
AES 400, ASS 200	1332	-	62	70	126
		272	2 or 4 channels		
			59	59	126
AMS 620	1584	-	59	59	126
		272	2 channels		
			50	50	100
			4 channels		
			50	50	84
		528	8 channels		
			44	44	63
CEP 1622, CSP 1622, CSP 1625	1152	-	10 or 12 channels		
			31	31	31
DES 16••, DMS 16••, DSS 16••	1152	-	62	81	126
		144	62	64	64
DSS 1235	1152	144	62	72	85
DMS 1025	1152	144	62	72	102
EEP/ESP 08C22/1622	1152	-	62	64	126
TBX SAP 10	1808	-	52	52	117

Example:

For a TSX P57 154 processor:

Two TBX AES 200 (base and extension) and a TBX AMS 620 (base) are connected to the Fipio bus.

The memory used equals 4612, i.e.: 1332+272+1584+1424 (offset).

In this case, the memory used by the devices (4612) is less than the processor's memory (95744).

Momentum Device Compatibilities

At a Glance

The following compatibility information concerns:

- The address settings supported by the different connectable devices,
- The memory size these devices occupy in the configuration RAM.

Rules of Use

The implementation of Momentum devices on Fipio is only possible using the communication module: 170 FNT 110 01.

Address Range

The address ranges for Momentum devices vary depending on the type of processor used. The address range is:

- [1..62] for TSX P57 154 processors,
- [1..62] and [64..99] for the other processors (TSX P57 254/354/454/554).

Memory Space

Each processor *(see page 97)* has a memory space available for Fipio devices. This memory space equals **95744** bytes for TSX P57 154/254/354 processors and **214528** bytes for SX P57 454/554 processors.

Each Fipio device connected to the bus uses part of this memory space. The memory used by the devices connected to the Fipio bus must be less than the processor's available memory. When calculating the memory used, you must add an offset of 1424 bytes.

The following table lists the memory values used in bytes for each Momentum module.

Module	Base	Maximum number of connections		
		TSX P57 154	TSX P57 254/354	TSX P57 454/554
ADI, ADM, ADO	832	62	98	98
ARM 370 30	832	62	98	98
AAI 520 40 00	1808	52	52	98
AMM 090 00, AAI 030 00, AAO 120 00, AAO 921 00	1808	52	52	98
AAI 140 00	2304	40	40	92
AEC 920 00	1808	52	52	98
OTHER_FED	1280	62	62	99
OTHER_FED_P	2304	40	40	92

Module	Base	Maximum number of connections		
OTHER_FRD	832	62	98	98
OTHER_FRD_P	1744	54	54	99
OTHER_FSD	896	62	98	98
OTHER_FSD_P	1808	52	52	98

Example:

Or a TSX P57 154 processor.

Two momentums of type ADI and one momentum of type standard profile OTHER_FSD are connected to the Fipio bus.

The memory used equals 3984, i.e.: 832+832+896+1424 (offset).

In this case, the memory used by the devices (3984) is less than the processor’s memory (95744).

Programming Terminal Compatibilities

At a Glance

The following compatibility information concerns the address settings supported by the different connectable devices.

Rules of Use

A programming terminal equipped with a TSX FPC 10 card or PCMCIA card TSX FPP 20 must be connected to Fipio address 63.

Control Expert operates in online mode with the Fipio manager PLC.

If the manager PLC has a project in which one or more agent devices are configured, Control Expert communicates with the Premium agent Fipio PLCs.

Exchanges from a third-party device to a PLC connected as a Fipio agent are only possible if the PLC is Premium. There is no communication to a Micro-type PLC.

It is impossible to download a Control Expert project to the manager from a programming terminal. In fact, downloading causes the PLC to switch to STOP, which makes Fipio communication no longer managed by the bus arbiter. However, it is possible to download a Control Expert project to a Fipio agent PLC.

Miscellaneous Device Compatibilities

At a Glance

The following compatibility information concerns:

- The address settings supported by the different connectable devices,
- The memory size these devices occupy in the configuration RAM.

Address Range

The table below shows the Fipio address range a device can use depending on the type of processor used:

Type of device	Type of processor		
	TSX P57 154	TSX P57 254/354	TSX P57 454/554
Advantys STB	1...62	1...62, 64...127	1...62, 64...127
Altivar (ATV-38/58/58F/68)	1...62	1...62	1...62
Analog TBX (AES, AMS, ASS)	1...62	1...62, 64...127	1...62, 64...127
Gateway (TBX SAP 10)	1...62	1...62, 64...127	1...62, 64...127
Inductel	1...62	1...62	1...62
Lexium	1...62	1...62	1...62
Magelis	1...62	1...62	1...62
Momentum	1...62	1...62, 64...99	1...62, 64...99
Standard profiles (FED, FDR, FSD)	1...62	1...62, 64...127	1...62, 64...127
Discrete TBX (DES, DMS, DSS)	1...62	1...62, 64...127	1...62, 64...127
Discrete TBX compact (CEP and CSP)	1...31	1...31	1...31
TSX (37 and 57)	1...62	1...62	1...62
Waterproof IP65 (EEP, ESP)	1...62	1...62, 64...127	1...62, 64...127
Waterproof IP67 (EEF, ESF, EMF)	1...62	1...62, 64...99	1...62, 64...99

Memory Space

Each processor *(see page 97)* has a memory space available for Fipio devices. This memory space equals **95744** bytes for TSX P57 154/254/354 processors and **214528** bytes for TSX P57 454/554 processors.

Each Fipio device connected to the bus uses part of this memory space. The memory used by the devices connected to the Fipio bus must be less than the processor's available memory. When calculating the memory used, you must add an offset of 1424 bytes.

The following table lists the memory values used in bytes for each device.

Module	Base
ATV38 with VW58311 card	1280
ATV58/58F with VW-A58301 card	1808
ATV58/58F PKW with VW58311 card	1280
ATV 68	1280
Magelis	1424
EEF 08D2, EEF 16D2	832
ESF 08T22 EMF 16DT2	1808
FED C32	1280
FED C32P	2304
FED M32	1424
FED M32P	2448
FRD C2	832
FSD C8	896
FSD C8P	1808
FSD M8	1040
FSD M8P	1952
Premium/Micro Agent	1424
Lexium	1424
Inductel	1808
Advantys STB 2 words	832
Advantys STB 8 words	896
Advantys STB 32 words	1280

Example:

Or a TSX P57 154 processor.

Two Magelis and a standard profile FSD C8P are connected to the Fipio bus.

The memory used equals 6080, i.e.: 1424+1424+1808+1424 (offset).

In this case, the memory used by the devices (6080) is less than the processor’s memory (95744).

Fipio Bus Operating Mode

At a Glance

Fipio bus operating mode is linked to the processor's operating mode and to the physical state of the bus.

If the number of physical errors detected by the processors endangers the nominal operation of the devices connected to the bus, then the processor stops all bus activity. In this case, certain connected devices apply fallback values to the inputs.

Operating Mode

The operating mode is as follows:

Situation	Description
1	Cold start of the processor with a Fipio project: the bus is automatically started, the inputs/outputs are checked and controlled.
2	Downloading a Fipio project: the bus is automatically started, the inputs/outputs are checked and controlled.
3	When a communication issue is detected by the processor (equipment powered off, Fipio cable disconnected, ...) the status bit corresponding to the equipement in the system words %SW128 to %SW143 *(see EcoStruxure™ Control Expert, System Bits and Words, Reference Manual)* is reset.The equipment status bits should be tested before their input/output values are used.
4	If the processor detects serious physical errors, the Fipio bus is automatically stopped. The connected devices apply the fallback values and the processor no longer checks inputs/outputs. An example of a serious physical error is disconnection of the line end jack, disconnection of the Fipio terminal block from the processor, etc.
5	Following an automatic stopping (serious errors detected by the processor): ● The `%SW144.2` bit is set to 1: the processor periodically executes an attempt to restart by trying to check and control the inputs/outputs of the connected devices. This is automatic mode. ● The `%SW144.2` bit is set to 0: to restart the bus, a user action must be carried out by program or in the programming terminal. Set the `%SW144.0` and `%SW144.1` bits to 1. This is manual mode.
6	Warm restart: The bus is automatically restarted if it was not intentionally stopped though an action on the `%SW144` system word 0 and 1 bits. The bus is set to the previous power outage state.
7	If `%S0` = 1 the system words are initialized, and if the bus was stopped, it is restarted.

Additional Information Concerning Manual Mode

If %SW144.0 is set to 0 (in the programming terminal or by program), then the producer/consumer (PC) function stops. No further variables or messages are exchanged on the bus.

If %SW144.1 is set to 0 (in the programming terminal or by program), then the bus arbiter (BA) function stops. No further variable or message scanning is carried out on the bus.

To completely and intentionally stop Fipio bus activity, the %SW144 system word 0 and 1 bits must be set to 0 simultaneously in the programming terminal or by the program.

Additional Information Concerning Automatic Mode

If the Fipio bus is stopped without intentional action on the %SW144 system word, this means that the processor has detected physical errors that inhibit nominal operation of the bus and of the connected devices.

In this case and only in this case, if the %SW144.2 bit is set to 1 (automatic start), then the processor will periodically and automatically attempt to restart exchanges on the bus.

If the %SW144.2 bit is set to 0 following a stop of the bus due to physical errors, then the %SW144.0 and %SW144.1 bits must be reset to 1 by the program or in the programming terminal in order to restart the Fipio bus. In this case, the bus will restart **if and only if the physical error that caused the stop has disappeared**.

Example

If the %SW144.2 bit is set to 1, when the Fipio terminal block is disconnected/reconnected from/to the processor, the bus will be restarted automatically.

The devices will be checked again and will be able to apply program values.

If the %SW144.2 bit is set to 0, when the Fipio bus is disconnected/reconnected from/to the processor, the devices connected to the bus will not be checked and will be unable to apply program values.

The devices will be checked again if the user sets the %SW144.0 and %SW144.1 bits to 1.

Saturation of Fipio Messaging Service

Messaging

The Fipio messaging service is used for communication between:

- the bus arbiter and a Premium or Micro PLC agent (equipped with a TSX FPP 10 card) or between agents,
- Control Expert and Fipio agents (Premium or Micro PLC),
- the bus arbiter with devices such as the HMI terminal (XBT), variable speed controllers (Altivar) or Lexium drives.

If the messaging service is saturated, the system returns an increasing number of time-out errors. Start by increasing the bandwidth allocated to the messaging service. If the problem persists, decrease *(see page 101)* the message rate.

This problem mainly occurs on TSX P57 554 processors which have a very short cycle time.

Network Cycle Time: Mono Task Application

At a Glance

The network cycle time for a mono task application is calculated for the following configuration:

- the bus length is 1 Km,
- the values corresponding to silence time, slot time and bandwidth are the default values (automatic mode).

Calculation of Network Cycle Time

For an application where all devices are configured in the same task, the value of the network cycle time of the task, in milliseconds, is obtained using the following formula (provided for information purposes):

$$TCR_TASK = 1.45 + \Sigma(K \text{ x equivalent number of devices from the same family})$$

Value of Coefficient K

Coefficient K will be defined as follows:

Coefficient K	Family	Device
0.5	TBX	Discrete
0.9	TBX	Analog
0.4	Momentum	Discrete
0.6	Momentum	Analog
0.6	ATV-58	All
1.5	TSX 37/57	All
0.6	AS-i gateway	TSX SAP 10
1.5	Magelis	All
0.4	STD_P	FRD
0.6	STD_P	FSP
1.5	STD_P	FED

Example

The example is configured as follows:

- All the elements are configured in the application's MAST task.
- The configuration consists of:
  - 10 discrete TBXs,
  - 11 analog TBXs,
  - 12 discrete Momentums.

The MAST task cycle time is calculated as follows:

TCR_MAST = 1.45 + ((0.5 x 10) + (0.9 x 11) + (0.4 x 12)) = 21.15 ms

The resulting network cycle time value is 21 ms.

Network Cycle Time: Multitasking Application

At a Glance

For a multitasking application with two periodic tasks, and if the devices are configured in each of the tasks, the value of the network cycle times depends on the configuration of the different tasks.

The cycle time values calculated with the following algorithm are the maximum values. The actual values are less than or equal to these values.

Convention

The following table shows all the parameters involved in the calculation of the network cycle time.

Parameter	Definition
P_{max}	Value of the largest task period (MAST or FAST)
P_{min}	Value of the smallest task period (MAST or FAST)
T_{max}	Task with period P_{max}
T_{min}	Task with period P_{min}
R	Ratio between the periods (This ratio must be a whole number; it will be rounded off to the nearest whole number if necessary).
NE_{max}	Number of devices from the same family configured in T_{max}
NE_{min}	Number of devices from the same family configured in T_{min}
NE_Equip	Equivalent number of devices from the same family
TCR_T_{max}	Network cycle time for the task T_{max}
TCR_T_{min}	Network cycle time for the task T_{min}

Calculation Algorithm

To calculate the network cycle time, the algorithm is as follows:

Phase	Description
1	Calculation of R $$R = \frac{Pmax}{Pmin}$$
2	Calculation of NE_Equip (for each device family) $$NE_Equip = ((R \times NEmin) + NEmax)$$
3	Calculation of TCR_T_{max} $$TCR_Tmax = 1{,}15 + \sum (K \times NE_Equip)$$

Phase	Description
4	Calculation of TCR_T_{min} $$TCR_Tmin = \frac{TCR_Tmax}{R}$$

Value of Coefficient K

Coefficient K will be defined as follows:

Coefficient K	Family	Device
0.5	TBX	Discrete
0.5	TBX	Discrete Compact
0.9	TBX	Analog
0.5	TBX IP65	All
0.4	TBX IP67	All
0.4	Momentum	Discrete
0.6	Momentum	Analog
0.6	Momentum expert	170 AEC 920
0.6	Altivar	All
1.5	TSX 37/57	All
0.6	AS-i gateway	TSX SAP 10
1.5	Magelis	All
0.4	STD_P	FRD
0.6	STD_P	FSP
1.5	STD_P	FED
1.5	Lexium	All
0.6	Inductel	All
0.4	Advantys	2W
0.6	Advantys	8W
1.5	Advantys	32W

Interpretation of the Results

To refresh the inputs/outputs on the bus, the following constraints are imposed:

- For the periodic tasks in controlled mode, the network cycle time of the task must be less than the task period.
- For the periodic tasks in free mode, the network cycle time of the task must be less than the task watchdog.

Example of calculating the network cycle times of a multitasking application

Introduction

The configuration of the example is as follows:

- Task periods:
  - MAST: 60 ms
  - FAST: 15 ms
- The configuration comprises:
  - 6 discrete TBXs configured in the MAST task and 4 discrete TBXs configured in the FAST task,
  - 6 analog TBXs configured in the MAST task and 4 analog TBXs configured in the FAST task,
  - 6 discrete Momentums configured in the MAST task and 4 discrete Momentums configured in the FAST task,
  - 6 analog Momentums configured in the MAST task and 4 analog Momentums configured in the FAST task,

Convention

Depending on the configuration, the application of conventions involves:

- P_{max} = 60 and P_{min} = 15
- $T_{max} = T_{MAST}$ and $T_{min} = T_{FAST}$
- $TCR_T_{max} = TCR_T_{MAST}$ and $TCR_T_{min} = TCR_T_{FAST}$

Algorithm

The following table shows the algorithm:

Stage	Description
1	Calculation of R R = 60 / 15 = 4
2	Calculation of the equivalent number of devices NE_TBX_TOR = (R * 4 + 6) = 22 NE_TBX_ANA = (R * 4 + 6) = 22 NE_MOMENTUM_TOR = (R * 4 + 6) = 22 NE_MOMENTUM_ANA = (R * 4 + 6) = 22
3	Calculation of the network cycle times TCR_T_{MAST} TCR_T_{MAST} = 1.45 + ((0.5 * 22) + (0.9 * 22) + (0.4 * 22) + (0.6 * 22)) TCR_T_{MAST} = 54.25 ms
4	Calculation of the network cycle times TCR_T_{FAST} TCR_T_{FAST} = 54.25 / 4 TCR_T_{FAST} = 13.56 ms

Conclusions

The network cycle times of each equal:

- TCR_T_{MAST} = 55 ms,
- TCR_T_{FAST} = 14 ms.

The time period for MAST and FAST tasks are sufficient to refresh the inputs/outputs.

Chapter 4
Installation methodology

Overview

Introduction

The software installation for the modules is carried out from the various Control Expert editors:

- In offline mode,
- In online mode.

The following order of installation phases is recommended but it is possible to change the order of certain phases (for example, starting with the configuration phase).

Installation Phases with Processor

The following table shows the various phases of installation with the processor.

Phase	Description	Mode
Declaration of variables	Declaration of IODDT-type variables for the application-specific modules and variables of the project.	Offline (1)
Programming	Project programming.	Offline (1)
Configuration	Declaration of modules.	Offline
	Module channel configuration.	
	Entry of configuration parameters.	
Association	Association of IODDTs with the channels configured (variable editor).	Offline (1)
Generation	Project generation (analysis and editing of links).	Offline
Transfer	Transfer project to PLC.	Online
Adjustment / Debugging	Project debugging from debug screens, animation tables.	Online
	Modifying the program and adjustment parameters.	
Documentation	Building documentation file and printing miscellaneous information relating to the project.	Online (1)
Operation/Diagnostic	Displaying miscellaneous information necessary for supervisory control of the project.	Online
	Diagnostic of project and modules.	
Key:		
(1)	These various phases can also be performed in the other mode.	

Chapter 5
Configuring Fipio communication

Subject of this Chapter

This chapter describes the Configuration process during set-up of Fipio communication.

What Is in This Chapter?

This chapter contains the following sections:

Section	Topic	Page
5.1	General information on configuration	82
5.2	Fipio bus configuration	83
5.3	Configuration of devices on the Fipio bus	103
5.4	Management of Inputs/Outputs by Control Expert Tasks:	108
5.5	Confirmation of the Fipio bus configuration	112

Section 5.1
General information on configuration

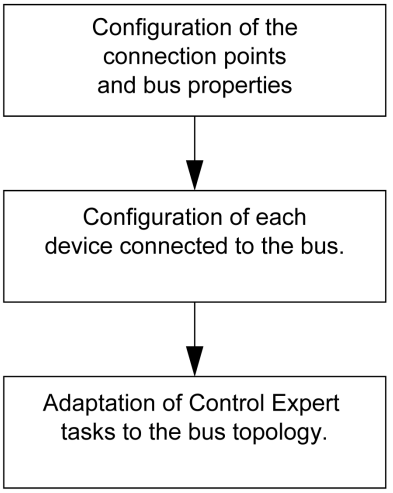
Configuration Mode

At a Glance

The creation of a project implementing a Fipio bus requires the configuration of several elements from the Control Expert software.

Methodology

The following methodology describes the different configuration phases of the Fipio bus and devices:



This phase allows you to define the type of device connected to the bus, and the bus management characteristics (e.g.: length, bandwidth, etc.).

This phase allows you to configure each device (e.g.: selection of the Control Expert task to manage the device, inputs, and outputs, etc.).

This phase allows you to define how the inputs/outputs on the bus are managed by the Control Expert tasks (e.g.: cyclic or periodic task, period, controlled or free mode, etc.).

Section 5.2
Fipio bus configuration

Aim of this Section

This section introduces the Fipio bus configuration principles.

What Is in This Section?

This section contains the following topics:

Topic	Page
How to Access the Fipio Bus Configuration Screen	84
Fipio Bus Configuration Screen	85
How to Add a Device on the Bus	87
How to Add an Extension Module on the Bus	89
How to Delete/Move/ Duplicate a Bus Device	91
Changing the Fipio Communicator Type	95
Fipio Bus Displayed in the Project Browser	96
How to Access the Properties Screen of the Fipio Bus	97
Properties of the Fipio Bus: General Tab	99
Properties of the Fipio Bus: Expert tab	100
Expert Tab: Manual Mode	101

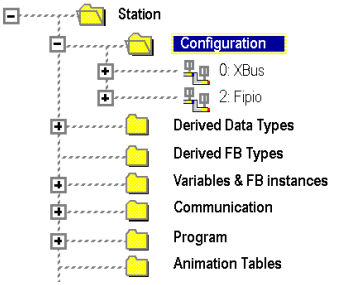
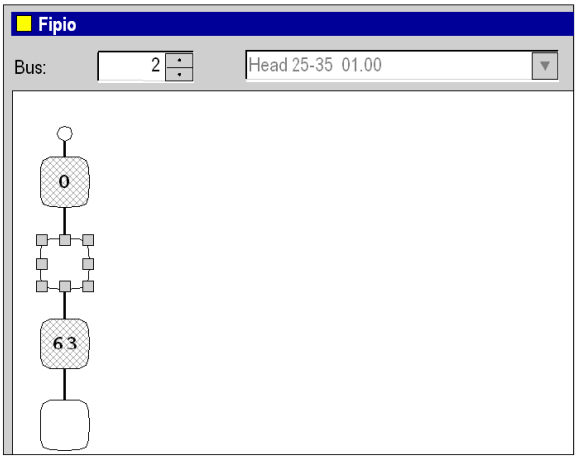
How to Access the Fipio Bus Configuration Screen

At a Glance

This describes how to access the configuration screen of the Fipio bus for a Premium or Atrium PLC with a built-in Fipio link.

Procedure

To access the Fipio field bus, perform the following actions:

Step	Action
1	From the project navigator, deploy the **Configuration** directory. **Result**: the following screen appears 
2	Select the Fipio sub-directory and select the **Open**command using the contextual menu. **Result**: the **Fipio** window appears 

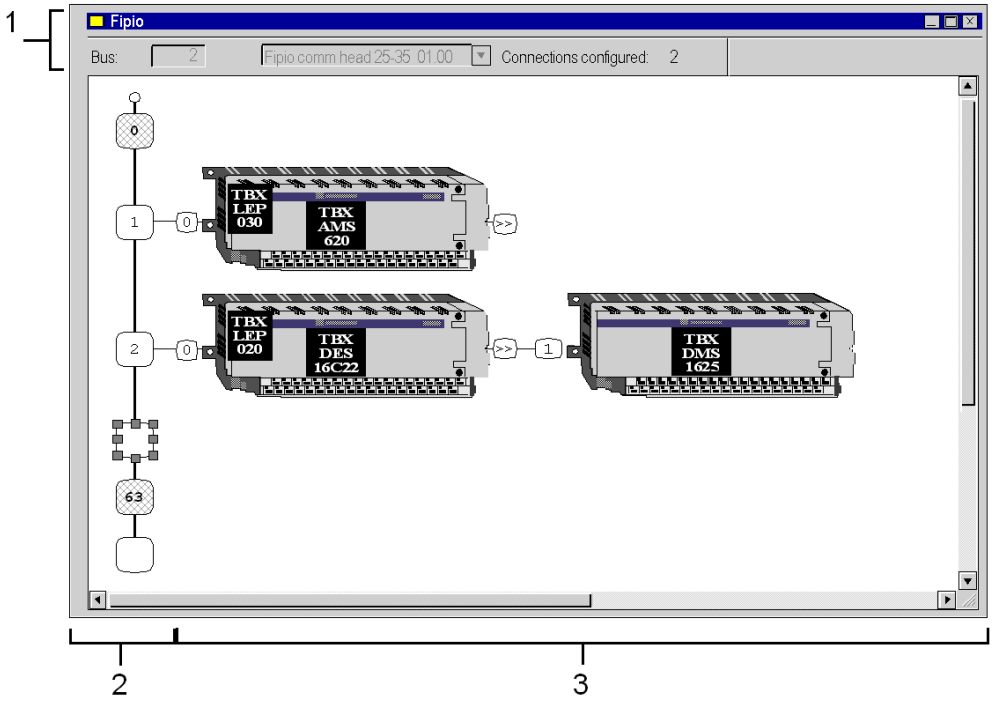
Fipio Bus Configuration Screen

At a Glance

This screen is used to declare devices which are connected to the bus.

Illustration

The Fipio communication screen looks like this:



Elements and Functions

This table describes the different areas that make up the configuration screen:

Number	Element	Function
1	Bus	Bus number.
	Connections configured	Indicates the number of connection points configured (0 and 63 are not counted).
2	Logical address area	This area consists of: ● slot 0 (manager), ● the addresses of each device, ● free addresses between 0 and 63 and over 63, ● address 63 is reserved for the programming terminal.
3	Module area	This area contains the devices (base and extension modules, and the communicator if necessary) which are connected to the bus.

Gray squares on the bus indicates the two reserved addresses:

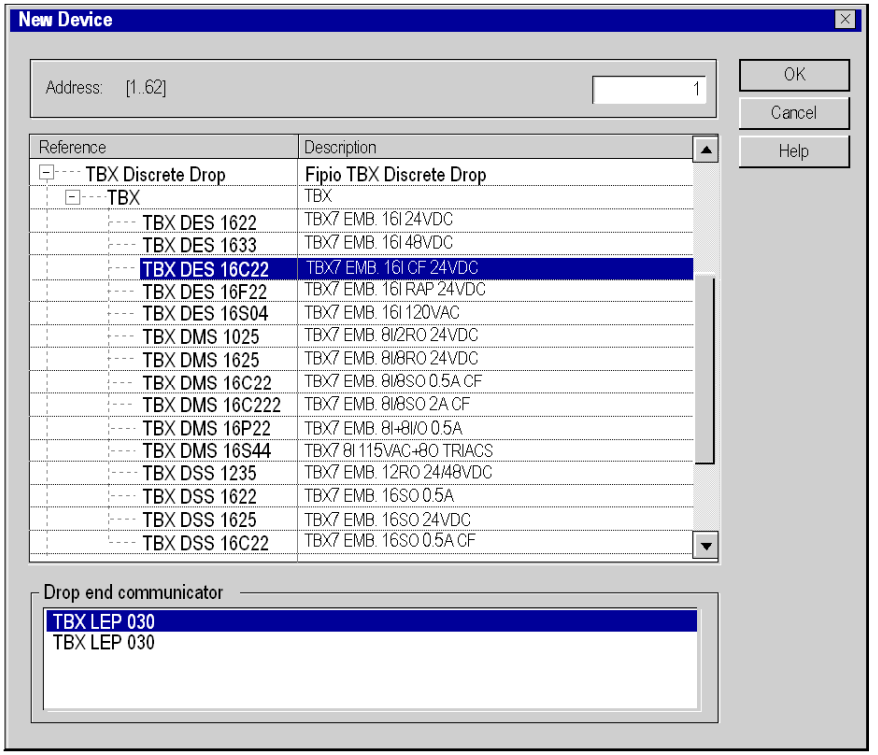
- 0 for the bus manager,
- 63 for the privileged terminal.

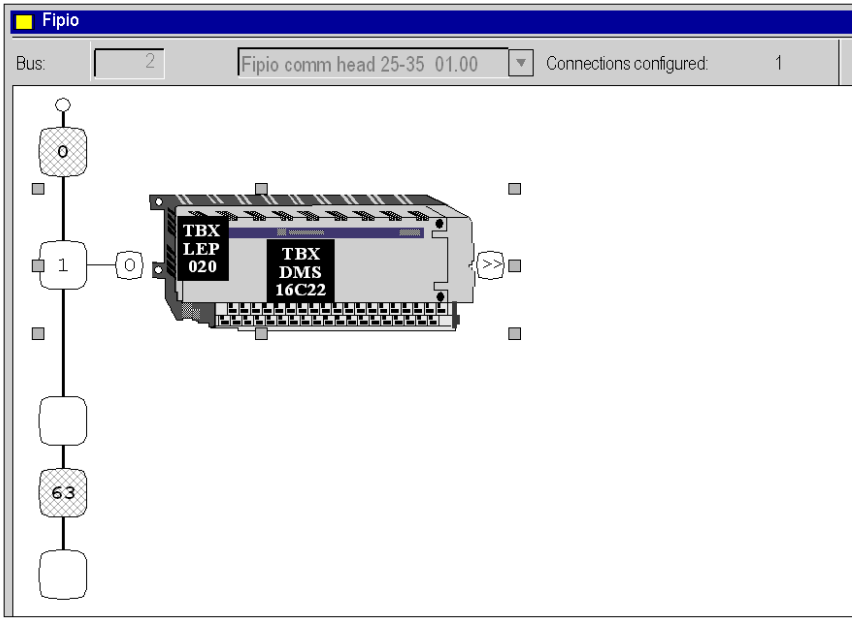
Available connection points are indicated by a white square with rounded edges.

How to Add a Device on the Bus

Procedure

This operation is used to add, via the software, a device connected to the Fipio bus.

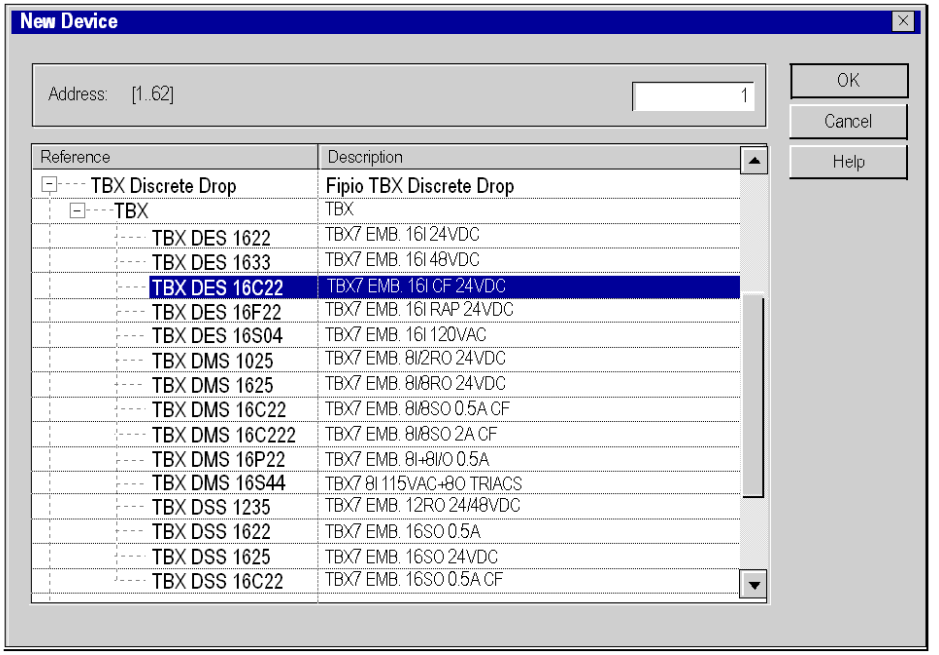
Step	Action
1	Access the Fipio *(see page 84)* configuration screen.
2	Right-click on the logical address of the connection point, at the location where the module has to be connected (available addresses from 1 to 62 and 64 to 127, addresses 0 and 63 being reserved by the system). **Result**: the **New Device** screen appears. 
3	Enter the number of the connection point corresponding to the address. By default, the Control Expert software offers the first free consecutive address.
4	In the **Reference** field, enter the type of device to connect to the bus.

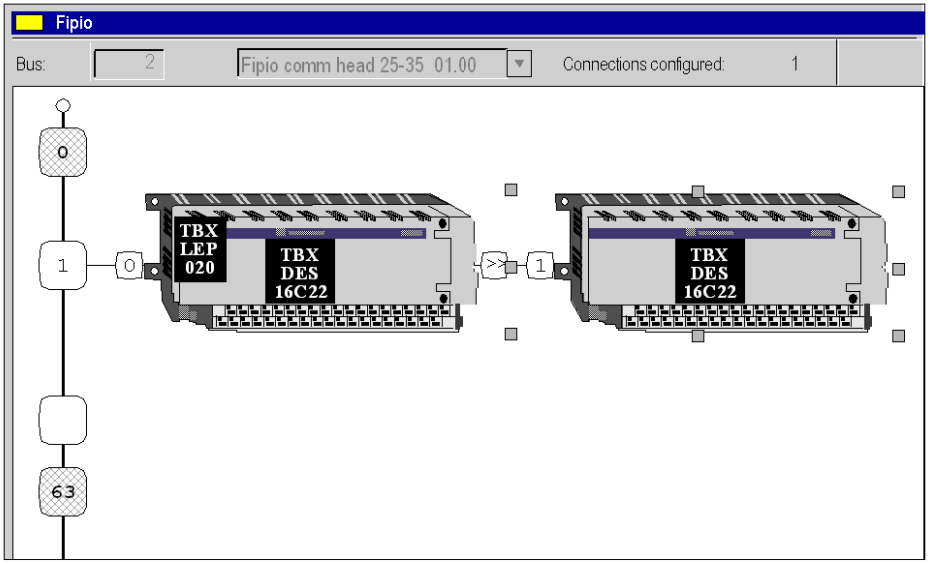
Step	Action
5	In the **Communicator** field, select the element type enabling communication on the Fipio bus. For modules with built-in communicators, this window does not appear.
6	Validate with **Ok**. **Result:** the module is declared. 

How to Add an Extension Module on the Bus

Procedure

This operation is used to add, via the software, an extension module to a base module which is connected to the Fipio bus.

Step	Action
1	Access the Fipio *(see page 84)* configuration screen.
2	On the base module, select the connection point which is shown as  .
3	From the contextual menu, select **New Device**. **Result**: the **New Device** screen appears. 
4	In the **Reference** field, select the type of device to connect to the base module.

Step	Action
5	Validate with **Ok**. **Result:** the module is declared. 

How to Delete/Move/ Duplicate a Bus Device

Procedure for Deleting a Device

This operation is used to delete, via the software, a device connected to the Fipio bus.

Step	Action
1	Access the Fipio *(see page 84)* configuration screen.
2	Select the connection point to be deleted. A dotted frame surrounds the selected connection point.
3	Select the **Edit → Delete** command.

Procedure to Follow to Delete an Extension Module

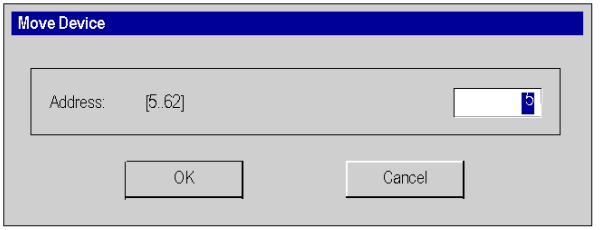
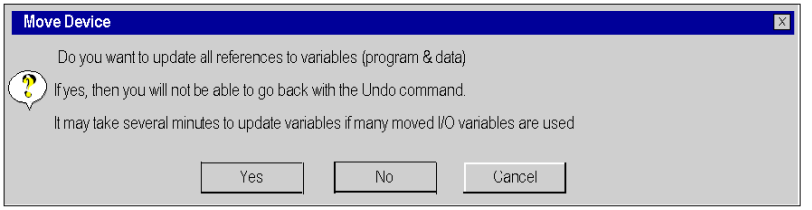
This operation is used to delete, via the software, an extension module connected to a base module.

Step	Action
1	Access the Fipio *(see page 84)* configuration screen.
2	Select the extension module to be deleted. A dotted frame surrounds the selected module.
3	Select the **Edit → Delete** command.

Procedure for Moving a Device

Moving a device does not involve a physical move on the bus, but rather a change in the device address logic. This triggers modification of the address of I/O objects in the program and movement of the variables associated with these objects.

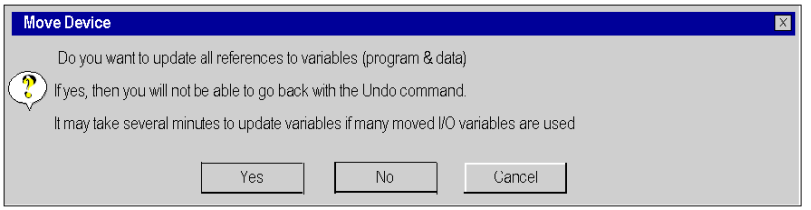
Step	Action
1	Access the Fipio *(see page 84)* configuration screen.
2	Select the connection point to be moved. A dotted frame surrounds the selected connection point.

Step	Action
3	**Drag and Drop** the connection point to be moved to an empty connection point. **Result**: the **Move Device** screen appears. 
4	Enter the number of the destination connection point.
5	Confirm the new connection point by pressing **OK**. **Result**: the **Move Device** screen appears. 
6	Confirm the modification by pressing **Yes** to modify the addresses of the I/O objects in the program and move the variables associated with these objects.

Procedure to Follow to Move an Extension Module

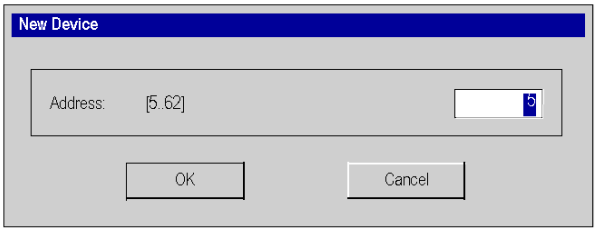
Moving an extension module does not involve a physical move on the bus, but rather a change in the device address logic. This triggers modification of the address of I/O objects in the program and movement of the variables associated with these objects.

Step	Action
1	Access the Fipio *(see page 84)* configuration screen.
2	Select the extension module to be moved. A dotted frame surrounds the selected connection point.

Step	Action
3	**Drag and Drop** the connection point to be moved to an empty connection point in the base module. **Result**: the **Move Device** screen appears. 
4	Confirm the modification by pressing **Yes** to modify the addresses of the I/O objects in the program and move the variables associated with these objects.

Procedure for Duplicating a Device

This feature is similar to the function for moving a device.

Step	Action
1	Access the Fipio *(see page 84)* configuration screen.
2	Select the connection point to be duplicated. A dotted frame surrounds the selected connection point.
3	Select the **Edit → Copy** command.
4	Select the destination connection point.
5	Select the **Edit → Paste** command. **Result**: the **New Device** screen appears. 
6	Enter the number of the destination connection point.
7	Confirm the new connection point by pressing **OK**.

Procedure to Follow to Duplicate an Extension Module

The following table shows the procedure for duplicating an extension module.

Step	Action
1	Access the Fipio *(see page 84)* configuration screen.
2	Select the extension module to be duplicated. A dotted frame surrounds the selected connection point.
3	Select the **Edit** → **Copy** command.
4	Select the connection point of the destination base module.
5	Select the **Edit** → **Paste** command. **Result**: the **New Device** screen appears.

Changing the Fipio Communicator Type

At a Glance

You can change the type of Fipio communicator for TBX IP20 discrete input/output modules.

Procedure

The following table shows the procedure for changing the Fipio communicator:

Step	Action
1	Access the Fipio *(see page 104)* configuration screen.
2	Select the Fipio communicator to be changed.
3	From the contextual menu, select the product reference of the new communicator (**TBX LEP 020** or **TBX LEP 030**).

Fipio Bus Displayed in the Project Browser

At a Glance

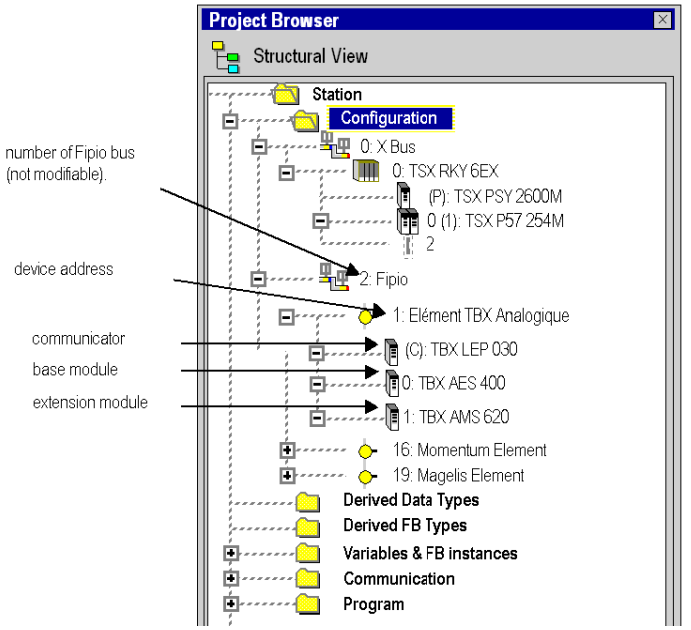
If the processor has a built-in Fipio link, the Fipio bus is shown in the **configuration** directory of the project browser with the number **2**. **This value cannot be modified**.

The project browser displays all the Fipio devices in the Fipio configuration screen *(see page 85)*. Each device is displayed with its address number. Depending on the type of device configured, the following items may be displayed below the device:

- a communicator,
- a base module,
- an extension module.

Whenever a Fipio device is added/removed to/from the Fipio bus, it also appears/disappears on/from the Fipio bus in the project browser.

The following illustration shows the Fipio bus with its devices in the project browser:



How to Access the Properties Screen of the Fipio Bus

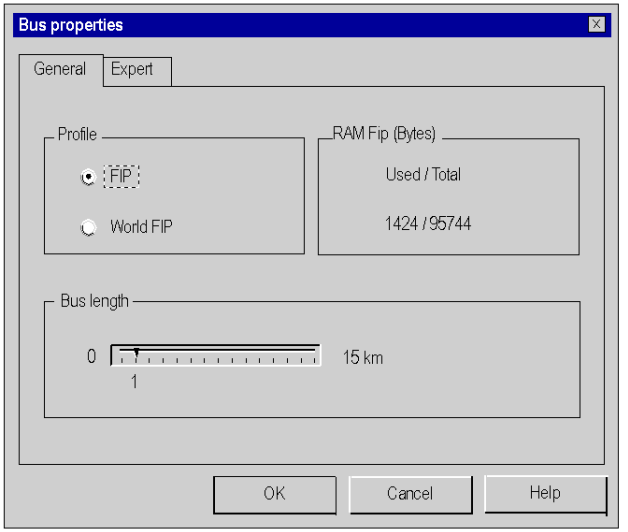
At a Glance

In a Fipio bus, the input/output exchange manager is the bus arbiter. This is in charge of managing access to the bus of each connected device.

The screen showing the properties of the bus lets you adapt the management parameters according to the devices connected.

How to Access the Properties

The following table shows the procedure for accessing the bus properties screen:

Step	Action
1	From the project navigator, deploy the **Configuration** directory.
2	Select the **Fipio** sub-directory.
3	Select the **Properties** command of the contextual menu. **Result**: the **Properties** screen appears. 

NOTE: Where a OZD FIP G3 is used, it is essential to set this value to 7km, whatever the actual length of the fiber optic.

Description of the Tabs

This table describes the different tabs which make up the properties screen:

Tab	Function
General	This tab is used to: ● Select the type of profile managed by the bus arbiter, ● View the RAM occupation of the devices, ● Adjust the length of the bus.
Expert	This tab is used to select the management mode. For manual management, it is used to adjust the management parameters depending on the connected devices.

Properties of the Fipio Bus: General Tab

At a Glance

With this tab, you can access three types of information:

- The profile,
- The RAM Fip,
- The bus length.

The Profile

The profile is chosen according to the devices which are connected to the Fipio bus. It can be in either of two versions:

- FIP:
  - ❍ This profile is chosen by default.
  - ❍ It is compatible with all the devices of Schneider and partners.
- WorldFip:
  - ❍ The selection of this mode allows you to operate the Fipio bus in accordance with the EN 50170 standard.

NOTE: The WorldFip profile mode can be used with the following restrictions:

- No TSX FP ACC6 or TSX FP ACC8 repeaters,
- Use of TBX LEP 020 version 1.4 minimum and TBX LEP 030 version 1.2 minimum,
- Use of the PCMCIA TSX FPP 10 card, version 1.8 minimum.

RAM Fip

This is the memory used by the configured Fipio devices in relation to the memory available in the processor.

The bus length

With this cursor you can increase or decrease the bus length. The default value is 1 Km and the adjustment increment is 1 Km.

This length must be adjusted because it is taken into account in the calculation of the propagation time for the variables and messages on the bus and consequently in the calculation of the network cycle times.

NOTE: At each modification of the length of the Fipio bus, all the connected devices must be powered off, then on after the new project has been downloaded (in the PLC).

Properties of the Fipio Bus: Expert tab

At a Glance

With this tab, you have access to the management mode concerning the calculation of:

- reversal and silence time,
- aperiodic bandwidths.

NOTE: At each modification of the parameters below, all the connected devices must be powered off, then on after the new project has been downloaded (to the PLC).

Automatic Mode

NOTE: This mode is the recommended mode.

The reversal and silence times and the bandwidths cannot be modified. They are calculated according to the bus parameters and device configuration data.

Manual Mode

No monitoring is carried out in manual mode. Modifying these parameters can cause the device connected to the bus not to work.

This mode is used to enter the reversal and silence times and the bandwidths.

It is mainly reserved for specific projects which require these values to be modified (example: use of Hirschmann transceivers).

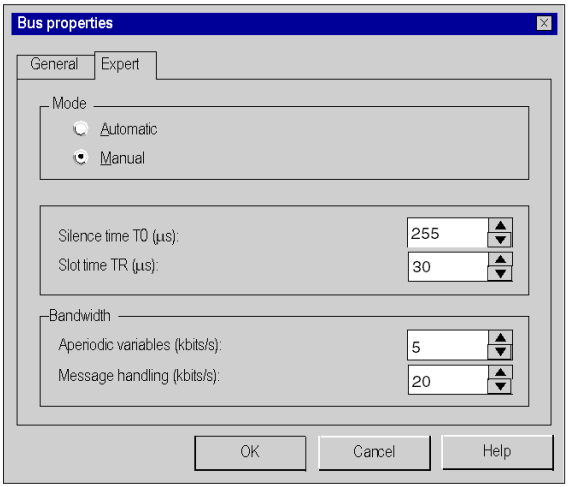
Expert Tab: Manual Mode

At a Glance

By selecting the manual mode, you have access to the following information:

- The reversal and silence times,
- The aperiodic bandwidths.

Illustration:



Aperiodic variables

The following table gives a description of the times:

Reference	Description
Silence time delay T0	This defines the maximum time between the transmission of the last byte of a frame and the transmission of the first byte of the following frame. This time is expressed in micro-seconds. Its value must be greater than the reversal time TR and less than 255.
Reversal time TR	This defines the minimum time between the transmission of the last byte of a frame and the transmission of the first byte of the following frame. This time is expressed in micro-seconds. Its value must be greater than or equal to 1 and less than the silence time T0.

NOTE: In the case of using an OZD FIP G3, it is essential to adjust the value of T0 = 255µs and TR = 30µs for all configurations.

Aperiodic Bandwidths

There are two types of exchange on a Fipio bus:

- Exchanges of cyclic variables which are reserved for the process data,
- Exchanges of aperiodic variables which concern the configuration of connected devices, the adjustment and the messaging.

A decrease or increase of the bandwidth of the aperiodic exchanges (Variables and Messages) slows down or accelerates the output of the configuration or adjustment variables which can circulate on the bus. The same applies to a decrease or increase of the aperiodic bandwidth of the messaging facility.

The following table gives a description of the aperiodic bandwidths:

Reference	Description
Aperiodic variables	This defines the output of the aperiodic variables which can circulate on the bus. This bandwidth is expressed in Kbits/s. Its value can be adjusted from 1 to 200 Kbits/s.
Messaging	This defines the output of the aperiodic messages which can circulate on the bus. This bandwidth is expressed in Kbits/s. Its value can be adjusted from 1 to 200 Kbits/s.

NOTE: An increase in the aperiodic bandwidth increases the network cycle time.

Section 5.3
Configuration of devices on the Fipio bus

Aim of this Section

This section introduces the configuration principles for Fipio bus devices.

What Is in This Section?

This section contains the following topics:

Topic	Page
How to Access the Configuration Screen for a Fipio Device	104
Configuration Screen of a Fipio Device	105
How to Access Different Parameters Depending on the Type of Device	107

How to Access the Configuration Screen for a Fipio Device

At a Glance

When a device has been declared, its configuration can be accessed in the same way as for an in-rack module.

This screen is used to select the task which exchanges the inputs/outputs: FAST or MAST. It can also be used to modify the configuration or adjustment parameters specific to the module.

Procedure

The following table describes the procedure for accessing the configuration of a device on the Fipio bus.

Step	Action
1	Access the Fipio *(see page 84)* configuration screen.
2	Select the module to be configured.
3	Using the contextual menu, click **Open Module**.

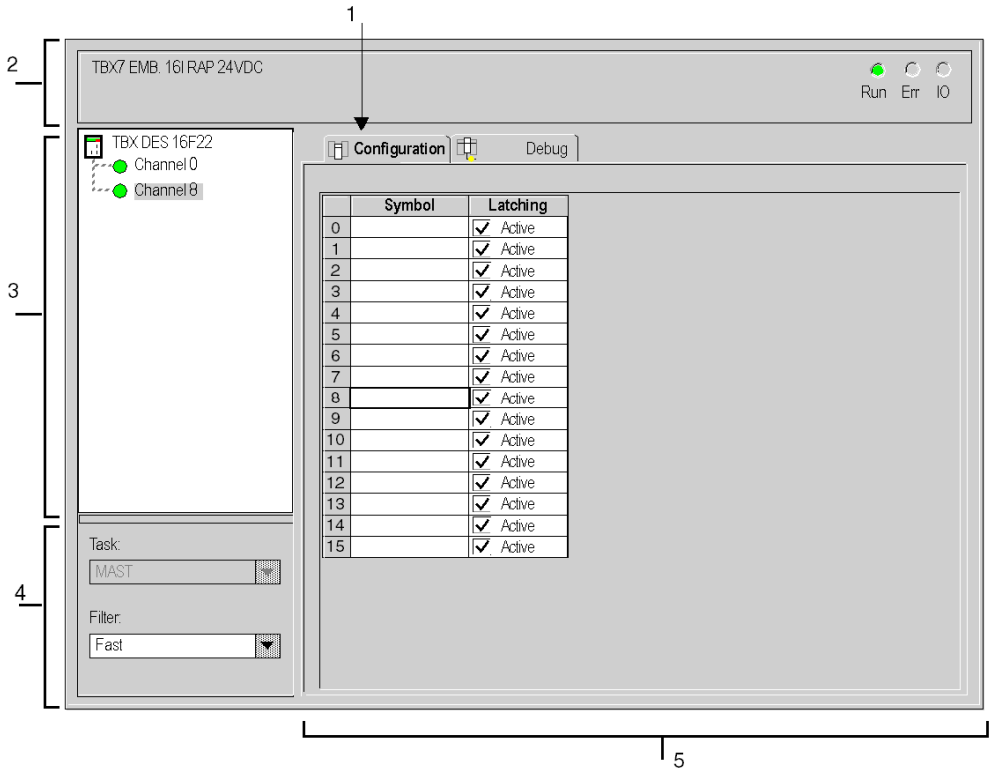
Configuration Screen of a Fipio Device

At a Glance

This screen, featuring several zones, is used to declare the communication channel and configure the parameters of the selected device for Fipio communication.

Illustration

The following figure shows an example of a configuration screen.



Description

The next table shows the various elements of the configuration screen and their functions.

Number	Element	Function
1	Tabs	The tab in the foreground indicates the mode in progress (**Configuration** in this example). Every mode can be selected using the respective tab. The available modes are: ● **Configuration**, ● **Adjust**, ● **Debug** which can be accessed only in online mode, ● **Calibration** which can be accessed only in online mode, ● **Fault** which can be accessed only in online mode.
2	**Module** area	Gives a reminder of the device's shortened name.
3	**Channel** area	Is used: ● By clicking on the reference number, to display the tabs: ❍ **Description** which gives the characteristics of the device. ❍ **I/O Objects** *(see EcoStruxure™ Control Expert, Operating Modes)* which is used to presymbolize the input/output objects. ❍ **Fault** which shows the device faults (in online mode). ● To select the channel or group of channels to be configured, ● To display the **Symbol**, name of the channel defined by the user (using the variable editor).
4	**General parameters** area	This area is used to define: ● The type of task (**MAST** or **FAST**), ● The channel filtering, ● The fallback mode, ● The reactivation, ● etc.
5	**Configuration** area	This area is used to set up the channels of the devices. For certain types of devices, this area is not accessible.

NOTE: Refer to the documentation of each device for information on general, configuration, adjustment and debugging parameters.

How to Access Different Parameters Depending on the Type of Device

How to Access Inputs/Outputs

The following procedure shows how to access the input or output parameters of a module. It can be used in both offline and online mode.

Step	Action
1	Access the configuration screen of the Fipio *(see page 104)* device.
2	Select the **Configuration** tab.
3	Select the channel which corresponds to the inputs or the outputs or the inputs/outputs in the **module** area **Result**: the configuration screen corresponding to the input/output type is displayed in the **configuration** area.

How to Set the Configuration Parameters

The following procedure describes how to modify the configuration parameters of a module. It can be used in both offline and online mode.

Step	Action
1	Access the configuration screen of a Fipio *(see page 104)* device.
2	Select the **Configuration** tab.
3	Select the channel in the **module** area.
4	Depending on the type of device, you can do the following in the **configuration** area: ● check or uncheck a box and/or, ● select a value from a drop-down list and/or, ● modify a value directly.
5	Confirm by pressing **Enter**.

How to Assign a Module to a Control Expert Task

The following procedure shows how to assign a module to a Control Expert task. It can be used in offline mode only.

Step	Action
1	Access the configuration screen of a Fipio *(see page 104)* device.
2	Select the **Configuration** tab.
3	Select the channel in the **module** area.
4	In the **General parameters** area, select **Task** → **MAST** or **FAST**.

Section 5.4
Management of Inputs/Outputs by Control Expert Tasks:

Aim of this Section

This section introduces the configuration principles for Control Expert tasks in Fipio mode.

What Is in This Section?

This section contains the following topics:

Topic	Page
How to Access the Configuration of Control Expert Tasks in Fipio Mode	109
Management of Inputs/Outputs by Control Expert Tasks	110

How to Access the Configuration of Control Expert Tasks in Fipio Mode

At a Glance

The following elements are configured in the **Properties** window of the MAST or FAST task:

- the type of task (cyclic or periodic), as well as its watchdog and execution time,
- processing of Fipio inputs/outputs in relation to the task cycle (controlled or free).

The network cycle time is calculated using the elements provided above and the bus and Fipio device configuration.

NOTE: The Network Cycle Time is only significant where the project was generated after all modifications to the bus.

Procedure

The following table describes the procedure for accessing the MAST or FAST task configuration in Fipio mode:

Step	Action
1	From the project browser select the **MAST** or **FAST** sub-directory of the **Program** directory.
2	Select **Properties** from the contextual menu. **Result**: the **MAST Properties** or **FAST Properties** window is displayed.

Management of Inputs/Outputs by Control Expert Tasks

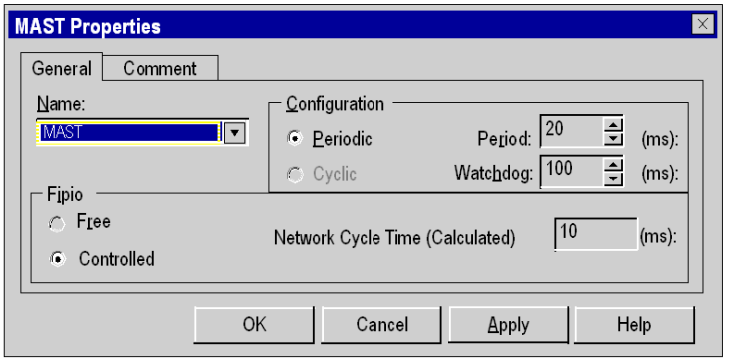
At a Glance

The MAST or FAST task screen is divided into several areas. It allows you to manage Fipio inputs/outputs according to:

- the task type (MAST or FAST),
- the execution type (cyclic or periodic) and period,
- the watchdog,

Controlled Fipio Mode

This mode can only be accessed for a Control Expert task declared as periodic.



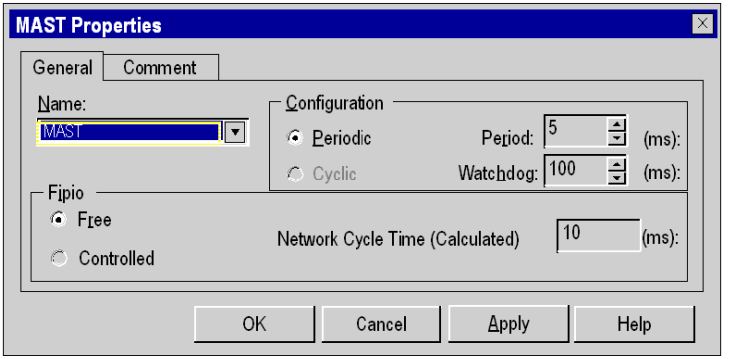
In this mode, the refreshing of inputs/outputs is correlated with the task period. The system guarantees that inputs/outputs are updated in a single period. Only the inputs/outputs associated with this task are refreshed.

The calculated network cycle time is used to adjust the PLC (MAST or FAST) task period.

This period must be greater than or equal to the calculated network cycle time.

Free Fipio Mode

This mode can be accessed for a Control Expert task declared as periodic.



In this mode, no restriction is imposed on the task period. The period of the PLC task (MAST or FAST) can be less than the network cycle time.

If a refresh is performed in a network cycle, the task can be executed without updating the inputs/outputs.

Selecting this mode gives you the possibility of having the lowest possible task times for projects where speed is critical.

Network Cycle Time

Network cycle time (calculated): this time is the network cycle time associated with a task. It is only displayed in the window if project generation is executed.

Section 5.5
Confirmation of the Fipio bus configuration

Aim of this Section

This Section introduces the main instances of Fipio bus configuration refusal.

What Is in This Section?

This section contains the following topics:

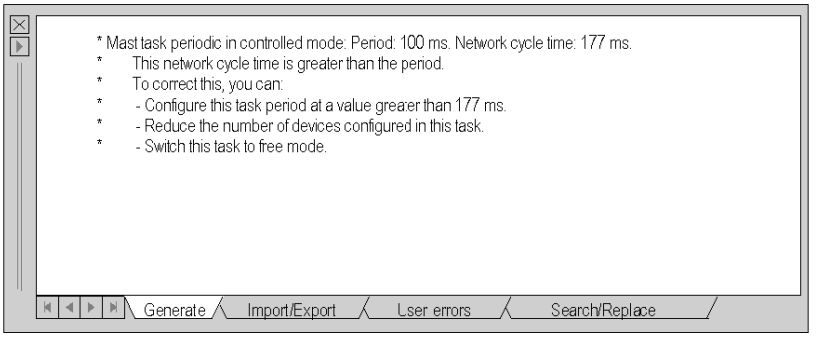
Topic	Page
Refused Build	113
Examples of Validation Refusals	114

Refused Build

At a Glance

When you use the **Build Project** or **Rebuild All Project** command, Control Expert checks the constraints relating to the project. If the constraints relating to task data for the devices configured and the bus parameters are not observed, the scan table is not generated.

In this case, a message is displayed in the output window



The configuration as a whole remains invalid until the problem is corrected by the user. This configuration cannot be transferred to the PLC.

If the task constraints are not observed, the cause of the failure and the recommendations on how to correct it, are provided in the properties.

Depending on the recommendations, you have two possible ways of correcting the configuration:

- Access the task properties screen to modify the Control Expert task management parameters.
- Access the Fipio configuration screen to modify the distribution of devices in the MAST or FAST tasks.

Examples of Validation Refusals

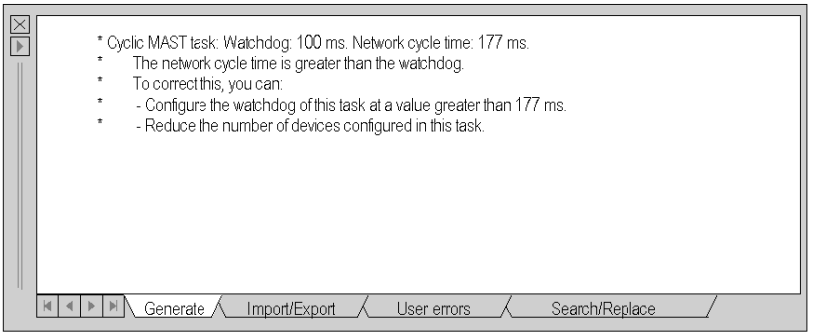
At a Glance

The following examples cover the main reasons for refusal that you are likely to encounter.

Example 1

The MAST task is configured in cyclic mode.

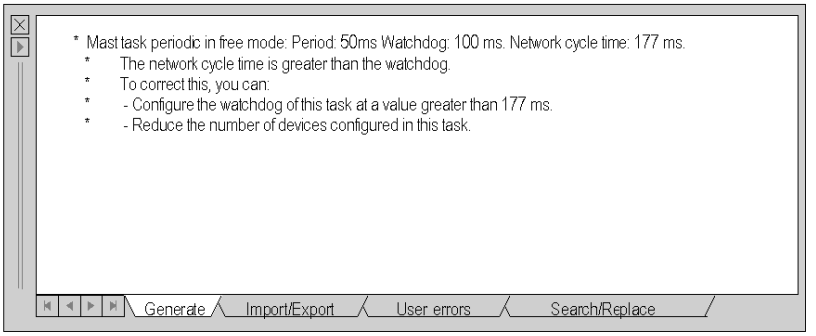
This network cycle time is greater than the watchdog time.



Example 2

The MAST or FAST task is configured in free periodic mode.

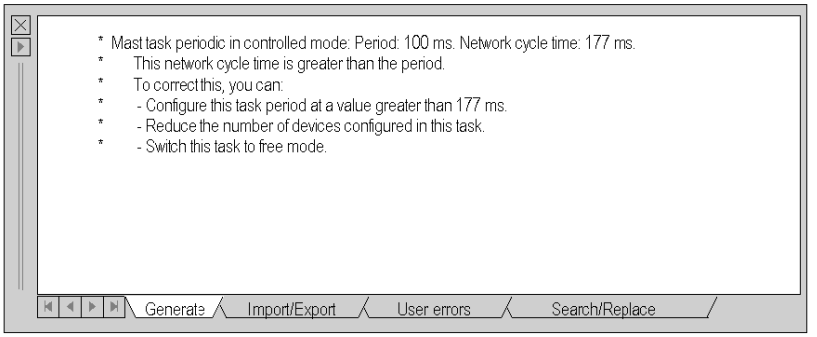
This network cycle time is greater than the watchdog time.



Example 3

The MAST or FAST task is configured in controlled periodic mode.

This network cycle time is greater than the period.



Chapter 6
Programming Fipio communication

Subject of this Chapter

This Chapter describes the Programming process during set-up Fipio communication.

What Is in This Chapter?

This chapter contains the following topics:

Topic	Page
Fipio Fault Processing	118
Examples of Detection of Explicit Exchange Saturation	119

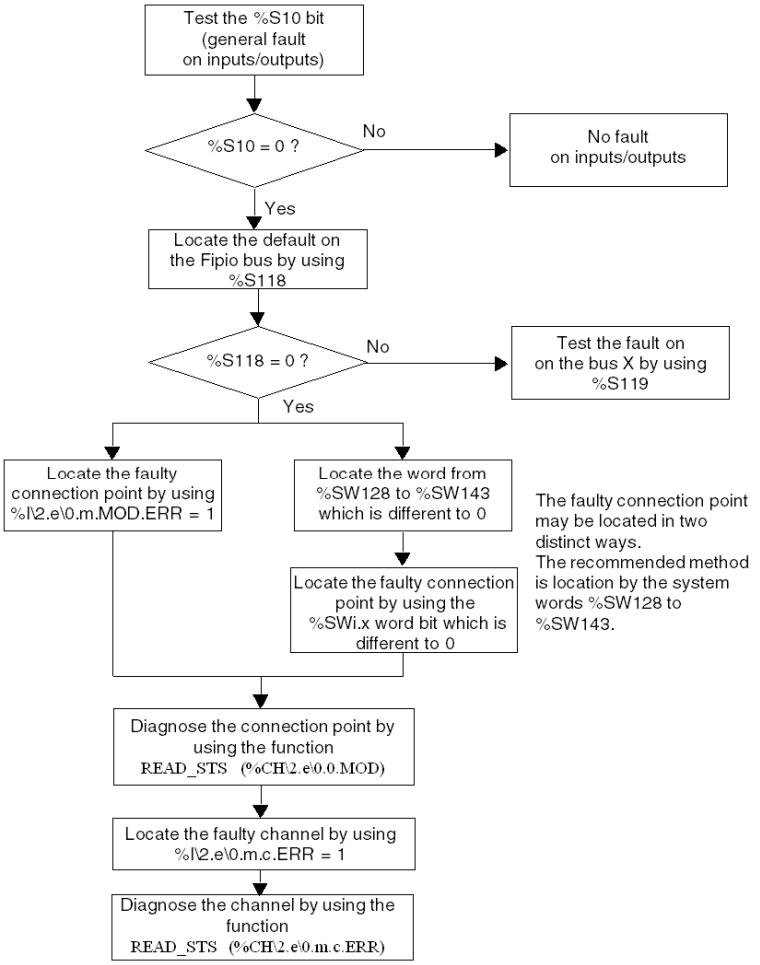
Fipio Fault Processing

At a Glance

Fipio faults can be processed from the debug and diagnostics screens. It can be advantageous to use the program to perform this processing.

Algorithm

The algorithm below is an example of an application algorithm that can be used to detect and manage faults on the Fipio bus



Examples of Detection of Explicit Exchange Saturation

At a Glance

The examples provided below are in Structured Text language, but they are transposable to any of the languages available in Control Expert.

They should only be used if the Control Expert project is likely to activate more than 24 explicit functions simultaneously and if you wish to relaunch a request in the event of explicit exchange saturation.

Management of READ_STS for Module

```
IF  (NOT  (%MW\2.2\0.0.MOD.0.0))  AND  (%I\2.2\0.0.MOD.ERR)  THEN
(*no request in progress*)
   SET (%M0); (*latching of a request*)
   (*bit indicating explicit exchange saturation reset to 0*)
   %SW116 :=%SW116 AND 2#1111111111111011;
   READ_STS (%CH\2.2\0.0.MOD);
   IF (%SW116.2) AND (%MW\2.2\0.0.MOD.2.1) THEN
      RESET (%M0); (*retry READ_STS, saturation occurred*)
      (*%MW\2.2\0.0.MOD.2.1 signals a functional fault*)
   END_IF;
END_IF;
IF(%M0 AND NOT (%MW\2.2\0.0.MOD.0.0)) THEN
   (*the %MW\2.2\0.0.MOD.2 can be used*)
   RESET (%M0);
END_IF;
```

Management of READ_STS for Channel

```
IF (NOT (%MW\2.2\0.0.0.0.0)) AND (%I\2.2\0.0.0.ERR) THEN (*no
request in progress*)
   SET (%M1); (*latching of a request for channels 0 to 7*)
   (*bit indicating explicit exchange saturation reset to 0*)
   %SW116 :=%SW116 AND 2#1111111111111011;
   READ_STS (%CH\2.2\0.0.0);
   IF (%SW116 .2) AND (%MW\2.2\0.0.0.2.6) THEN
      RESET (%M1); (*retry READ_STS, saturation occurred*)
      (*%MW\2.2\0.0.0.2.6 signals a communication fault*)
   END_IF;
END_IF;
IF(%M1 AND NOT (%MW\2.2\0.0.0.0.0)) THEN
   (*the %MW\2.2\0.0.0.2 can be used*)
   RESET (%M1);
END_IF;
```

READ_PARAM at Connection Point 2

```
IF (NOT (%MW\2.2\0.0.0.2.0)) AND RE(%M10) THEN (*no request
in progress*);
   SET (%M2); (*latching of a request for channels 0 to 7*)
   (*bit indicating explicit exchange saturation reset to 0*)
   %SW116 :=%SW116 AND 2#1111111111111011;
   READ_PARAM (%CH\2.2\0.0.0); (*parameter request*)
   IF ((%MW\2.2\0.0.0.1.2) AND (%SW116.2)) THEN
      (*exchange refused*)
      RESET (%M2); (*explicit exchange saturation*)
   END_IF;
RESET (%M10);
END_IF;
IF(%M2 AND NOT (%MW\2.2\0.0.0.2.0)) THEN
   (*read parameters complete*)
   RESET (%M2);
END_IF;
```

WRITE_PARAM at Connection Point 2

```
IF (NOT (%MW\2.2\0.0.0.2.0)) AND RE(%M11) THEN (*no request
in progress*)
   SET (%M3); (*latching of a request for channels 0 to 7*)
   (*bit indicating explicit exchange saturation reset to 0*)
   %SW116 :=%SW116 AND 2#1111111111111011;
   WRITE_PARAM (%CH\2.2\0.0.0); (*send parameters*)
   IF ((%MW\2.2\0.0.0.1.2) AND (%SW116.2)) THEN
      (*exchange refused*)
      RESET (%M3); (*explicit exchange saturation*)
   END_IF;
RESET (%M11);
END_IF;
IF(%M3 AND NOT (%MW\2.2\0.0.0.2.0)) THEN
   (*end of parameter send*)
   RESET (%M3);
END_IF;
```

WRITE_CMD at Connection Point 2

```
IF (NOT (%MW\2.2\0.0.0.1.0)) AND RE(%M12) THEN (*no request
in progress*)
   SET (%M3); (*latching of a request for channels 0 to 7*)
   (*bit indicating explicit exchange saturation reset to 0*)
   %SW116 :=%SW116 AND 2#1111111111111011;
   WRITE_CMD (%CH\2.2\0.0.0); (*send commands*)
   IF ((%MW\2.2\0.0.0.1.1) AND (%SW116.2)) THEN
      (*exchange refused*)
      RESET (%M3); (*explicit exchange saturation*)
   END_IF;
RESET (%M12);
END_IF;
IF(%M3 AND NOT (%MW\2.2\0.0.0.1.0)) THEN
   (*end of command send*)
   RESET (%M3);
END_IF;
```

Chapter 7
Debugging a Fipio communication

Subject of this Chapter

This Chapter describes the Debugging process during set-up of Fipio communication.

What Is in This Chapter?

This chapter contains the following topics:

Topic	Page
Debug Mode	126
How to Access the Debug Screens of Remote Devices	127
Debug Screen for a Device on the Fipio Bus	128

Debug Mode

At a Glance

The Debug mode is only accessible in online mode.

This mode is used:

- to see if a device is faulty by checking whether a red square is displayed in the Fipio configuration screen,
- to adjust and debug devices connected to the Fipio bus from their application screen.

Unavailable Functions

The following functions are not available from the Fipio configuration screen in online mode:

- addition of a new device in online mode,
- duplication or movement of a device,
- modification of a device,
- modification of the communication adapter,
- deletion of a device,
- addition of an extension module,
- deletion of an extension module,
- duplication or movement of an extension module.

A message indicating the reason why the function cannot be accessed is displayed in the Control Expert status bar.

How to Access the Debug Screens of Remote Devices

At a Glance

This describes how to access the debug screen of a device connected to the Fipio bus.

Procedure

The following table shows the procedure to follow:

Step	Action
1	Connect to the manager PLC.
2	Access the Fipio *(see page 84)* bus configuration screen.
3	Double-click on the module to debug.
4	Select the **Debug** tab.

Debug Screen for a Device on the Fipio Bus

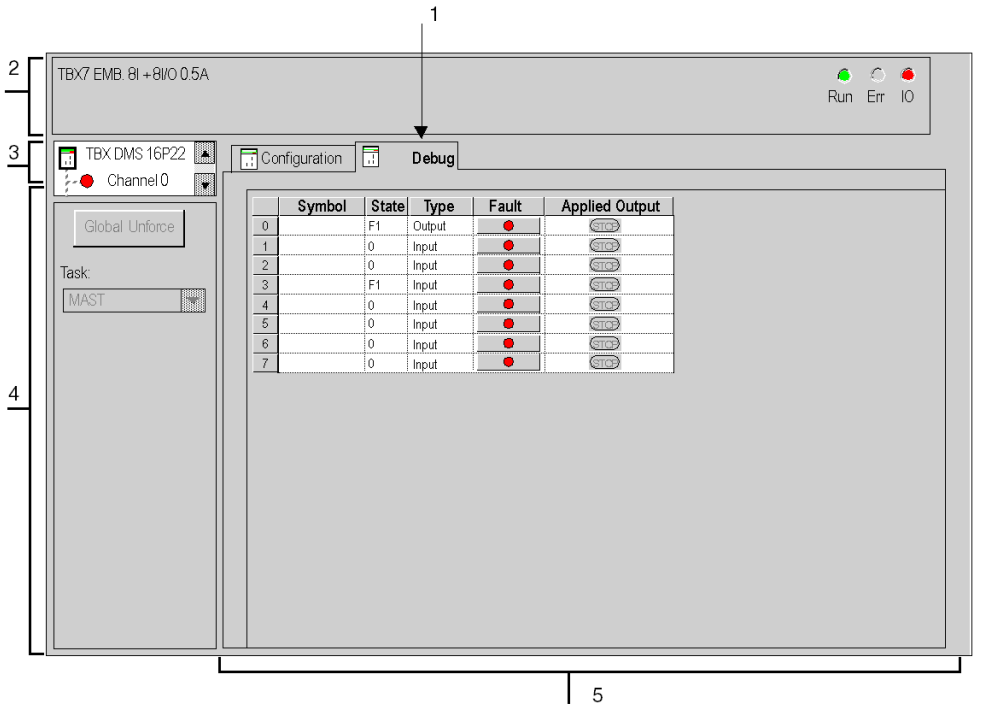
At a Glance

In debug mode the following functions can be accessed by opening the application screen of a device:

- viewing of the module configuration parameters,
- modification of the configuration,
- reading, writing and saving of adjustment parameters,
- reading, writing and forcing of inputs/outputs,
- aperiodic reading of status data,
- writing of command words.

Illustration

The figure below represents a debug screen.



Description

The following table shows the different elements of the debug screen and their functions.

Number	Element	Function
1	Tabs	The tab in the foreground indicates the mode in progress (**Debug** in this example). Each mode can be selected using the respective tab. ● **Debug** which can be accessed only in online mode, ● **Calibration** which can be accessed only in online mode, ● **Configuration**, ● **Adjust**.
2	**Module** area	Specifies the abbreviated heading of the module. In the same area there are 3 LEDs which indicate the module's operating mode: ● **RUN** indicates the operating status of the module, ● **ERR** indicates an internal fault in the module, ● **I/O** indicates a fault from outside the module or an application fault.
3	**Channel** area	Is used: ● By clicking on the reference number, to display the tabs: ❍ **Description** which gives the characteristics of the device. ❍ **I/O Objects** *(see EcoStruxure™ Control Expert, Operating Modes)* which is used to presymbolize the input/output objects. ❍ **Fault** which shows the device faults (in online mode). ● To select the required channel or channel group; to the left of the symbol there is a copy of the**CHx** channel LED.
4	**General parameters** area	Provides a reminder of the channel settings (**Task** type) and gives access to various functions (**Global Unforce**, etc.).
5	**Parameters in progress** area	This area displays the state of the inputs and outputs, and the different parameters in progress.

Specific Parameters

Refer to the documentation of each device for information on the general, configuration, adjustment and debugging parameters available.

For input/output modules, the specific parameters area shows the value and state of each of the module channels and gives access to the channel commands.

Chapter 8
Fipio communication diagnostics

Subject of this Chapter

This Chapter describes the Diagnostics process during set-up of Fipio communication.

What Is in This Chapter?

This chapter contains the following topics:

Topic	Page
Diagnostics Mode	132
How to Access the Fipio Bus Monitoring Screen	133
Diagnostics Screen: Fipio Bus Monitoring	134
How to Access the Device Monitoring Screen	136
Diagnostics Screen: Device Monitoring	137
How to Access the Device Monitoring Screen for a Single Device	139
Diagnostics Screen: Device Monitoring	140
How to Access the Communication Faults History Screen	142
Diagnostics Screen: Communication Errors History	143

Diagnostics Mode

At a Glance

In online mode it is possible to perform general diagnostics of bus errors and device faults.

These diagnostics can be performed using the screens available in Control Expert or by using the system bits and words managed by the application.

Two modes are available:

- Fipio bus monitoring,
- device monitoring with or without fault history.

Communication Counters

Each connection point configured (including connection points 0 and 63) has a **report** variable. This variable contains the communication and performance counter values.

This variable contains:

- the activity counters numbered from 16#00 to 16#20 compliant with the WORLDFIP standard,
- the fault counters numbered from 16#21 to 16#7F compliant with the WORLDFIP standard,
- the manufacturer-specific counters numbered from 16#80 to 16#FF; these counters are not compliant with the WORLDFIP standard.

The fault counters change value when communication faults occur on the Fipio bus.

Diagnostic Screens.

Four types of screen are used to view changes in the value of these various counters and so to view communication faults:

- The **Fipio Bus Monitoring** screen provides a global view of the Fipio bus. It displays the communication faults for the whole bus.
- The **Device Monitoring** screen provides a summary for each connection point. It shows all the connection points and signals any change in the value of the fault counters for each connection point.
- The **Device Monitoring at Connection Point** screen shows all counters with any change in their value for a given connection point.
- The **Communication Errors History** screen provides a fault history. It is used to keep a dated trace of changes in the fault counters.

Only one single screen of each type can be opened (one per connection point for the third type). If you try to open a screen that is already open it will be displayed instantly.

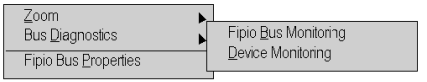
How to Access the Fipio Bus Monitoring Screen

At a Glance

This describes how to access the Fipio bus monitoring screen. It can be used in online mode only.

Procedure

The following table shows the procedure to follow:

Step	Action
1	Connect to the manager PLC.
2	From the **Configuration** directory in the project browser, select the **Fipio** bus.
3	Select the **Open** command. **Result**: the Fipio bus window is displayed
4	Right click on the background. **Result**: the contextual menu appears. 
5	Select the **Fipio Bus Monitoring** command.

Diagnostics Screen: Fipio Bus Monitoring

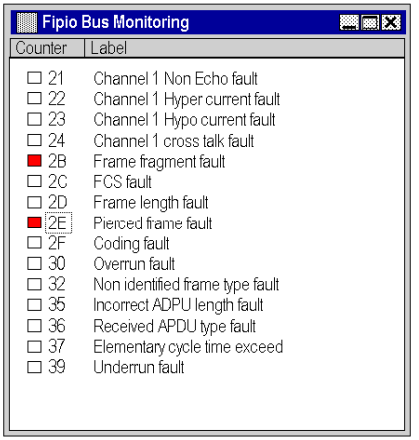
At a Glance

This screen provides the user with a global view of the bus and signals communication faults occurring on the bus.

The screen lists all the communication fault counters and signals any change in their value.

Illustration

The device screen looks like this:



Elements and functions

Each line contains:

- an icon showing the state of the fault counters (the reference state being the state they were in when the screen was opened):
  - white for normal,
  - red for a counter showing a fault,
- the counter number in hexadecimal,
- the counter label (as specified by the WORLDFIP standard).

Diagnostics

While the screen is open, the **report** variables of all connection points are read cyclically.

When the screen is opened or upon reset, the values of the counters for each **report** variable are stored to constitute a reference state. The time between two reads of the same variable is equal to the number of devices configured + 2 (0 and 63) in seconds.

As soon as a fault counter changes its value from its reference value, for any connection point, the counter icon changes to red. Reading the counter label provides details of the type of fault detected.

Example

Counter 21 changes to red. This means that at least one counter 21 of all the counters 21 has changed its value.

How to Reset the Reference State

The following procedure describes how to reset the reference state.

Step	Action
1	Right click on the background. **Result**: a shortcut menu appears. 
2	Select the **Reset** command.

Bus Monitoring by the Application

In Control Expert, the system words %SW150 and %SW151 indicate the number of frames sent and received by the channel manager.

If these two counters have stopped this indicates the Fipio channel manager function has stopped.

This is signaled to the user in the configuration screen in online mode by the connection point 0 shown in red.

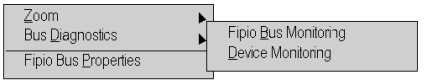
How to Access the Device Monitoring Screen

At a Glance

This describes how to access the Fipio bus device monitoring screen. It can be used in online mode only.

Procedure

The following table shows the procedure to follow:

Step	Action
1	Connect to the manager PLC.
2	From the **Configuration** directory in the project browser, select the **Fipio** bus.
3	Select the **Open** command. **Result**: the Fipio bus window is displayed
4	Right click on the background. **Result**: the contextual menu appears. 
5	Select the **Device Monitoring** command.

Diagnostics Screen: Device Monitoring

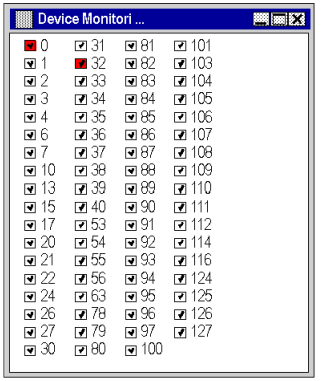
At a Glance

This screen is used to access monitoring of all configured devices.

The screen lists the configured connection points as well as points 0 and 63.

Illustration

The device screen looks like this:



Elements and Functions

The following elements are associated to each connection point:

- an icon specifying:
  - if the connection point must be monitored. If the box is checked, the device is monitored.
  - if the connection point is faulty. The icon changes to red if a fault is detected.
- the connection point number.

Diagnostics

While the screen is open, the **report** variables of all connection points are read cyclically.

When the screen is opened or upon reset, the values of the counters for each **report** variable are stored to constitute a reference state. The time between two reads of the same variable is equal to the number of devices configured + 2 (0 and 63) in seconds.

A connection point is considered to be faulty if the value of one of its fault counters has changed from the reference state. When the point is faulty, the icon is shown in red.

Diagnostics are only sent back if a communication is established between the bus manager and the faulty connection point. The faults are signaled for fault counters defined by the WORLDFIP standard.

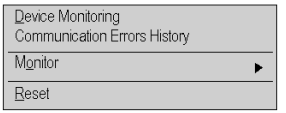
How to Deactivate Device Monitoring

The following procedure describes how to deactivate monitoring of a device on the bus.

Step	Action
1	Click on the check box of the connection point to be deactivated. **Result**: The box is unchecked In this case, the report variable of this connection point is no longer read and its reference state is reset.

How to Reset the Reference State

The following procedure describes how to reset the reference state.

Step	Action
1	Right click on the background. **Result**: a shortcut menu appears. 
2	Select the **Reset** command.

How to Access the Device Monitoring Screen for a Single Device

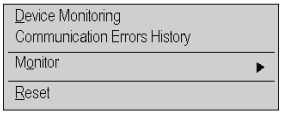
At a Glance

This describes how to access the monitoring screen for a single device connected to the Fipio bus and how to view its fault counters.

It can be used in online mode only.

Procedure

The following table shows the procedure to follow:

Step	Action
1	Access the **Device Monitoring** screen.
2	Select the connection point number you wish to access. **Result**: the number is displayed in reverse video.
3	Right click on the background. **Result**: a shortcut menu appears. 
4	Select the **Device Monitoring** command.

Diagnostics Screen: Device Monitoring

At a Glance

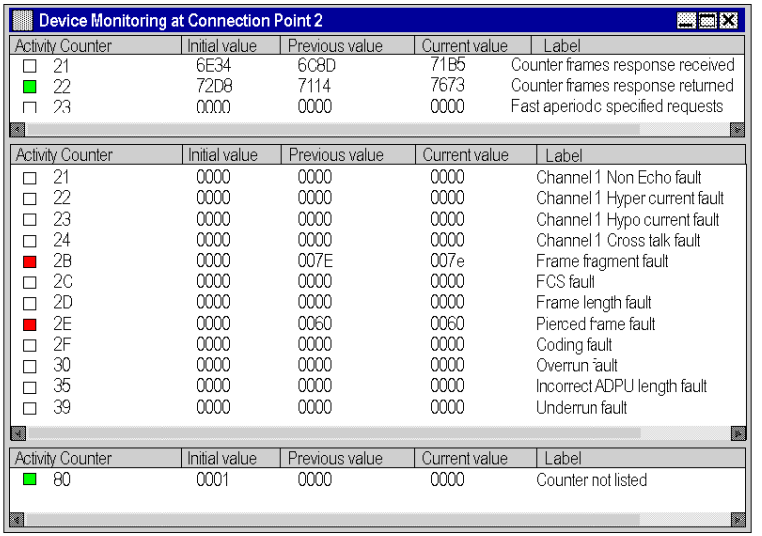
This screen is used to display the details of changes in the value of all counters for the selected connection point.

It is made up of three areas corresponding to each type of counter:

- activity,
- fault,
- manufacturer-specific.

Illustration

The device screen looks like this:



Elements and Functions

The following elements are associated to each counter:

- an icon indicating any change in the counter from its reference state:
  - white: no change,
  - green: change in the activity or manufacturer-specific counters,
  - red: change in the fault counters (the current value is different to the reference state).
- the counter number in hexadecimal,
- its reference value in hexadecimal,
- its value in hexadecimal from the last read operation,
- its current value in hexadecimal,
- its label in accordance with the WORLDFIP standard or the label **Counter not listed** for the manufacturer-specific part.

Diagnostics

While the screen is open, the **report** variables of all connection points are read cyclically.

When the screen is opened or upon reset, the values of the counters for each **report** variable are stored to constitute a reference state.

A connection point is considered to be faulty if the value of one of its fault counters has changed from the reference state. When the point is faulty, the icon is shown in red.

Diagnostics are only sent back if a communication is established between the bus manager and the faulty connection point. The faults are signaled for fault counters defined by the WORLDFIP standard.

How to Reset the Reference State

The following procedure describes how to reset the reference state.

Step	Action
1	Right click on the background. **Result**: a shortcut menu appears. 
2	Select the **Reset** command.

How to Access the Communication Faults History Screen

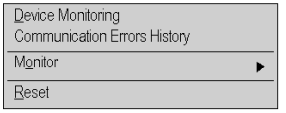
At a Glance

This describes how to access the Fipio bus communication faults history screen.

It can be used in online mode only.

Procedure

The following table shows the procedure to follow:

Step	Action
1	Access the **Device Monitoring** screen.
2	Right click on the background. **Result**: a shortcut menu appears. 
3	Select the **Communication Errors History** command.

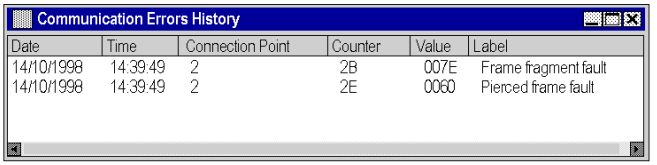
Diagnostics Screen: Communication Errors History

At a Glance

This screen lists the communication faults for the connection points configured (plus 0 and 63) and declared as monitored in the **Device Monitoring**screen.

Illustration

The history looks like this:



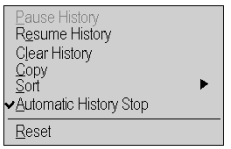
Elements and Functions

Every time a counter for a connection point changes, a line is added. It contains:

- the current date and time on the programming terminal.
- the connection point number in decimal.
- the fault counter number in hexadecimal,
- the counter value in hexadecimal,
- the counter label in accordance with the WORLDFIP standard.

How to Pause/Resume the History Function

This command is used to stop or resume updating of the screen.

Step	Action
1	Right click on the background. **Result**: a contextual menu appears. 
2	Select the **Pause History** or **Resume History** command.

How to Clear the History

This command can be accessed if the **Pause History** function has been activated. It can be used to delete all the lines on the screen.

Step	Action
1	Access the contextual menu.
2	Select the **Clear History** command.

How to Sort the History

This command can be accessed if the **Pause History** function has been activated. It can be used to sort all the lines of the history by date and time, or by connection point, or by counter.

Step	Action
1	Right click on the background.
2	Access the contextual menu.
3	Select the desired command: • **Sort → By Date and Time** • **Sort → By Connection Point** • **Sort → By Counter**

How to Copy the History

This command can be accessed if the **Pause History** function has been activated. It is used to copy the selected lines to the clipboard in text format.

Step	Action
1	Right click on the background.
2	Access the contextual menu.
3	Select the **Copy** command.

How to Stop the History Automatically

This function, when selected, allows you to stop the history when it reaches the maximum number of lines (4000 lines). A message informs you that the history has stopped. When the function has not been selected, and when the maximum number is reached, the newly added lines replace the first lines recorded (rotational mode).

Step	Action
1	Access the contextual menu.
2	Select the **Automatic History Stop**. **Result**: selecting the command checks the function.

How to Reset the Reference State

The following procedure describes how to reset the reference state.

Step	Action
1	Access the contextual menu.
2	Select the **Reset** command.

Chapter 9
Language objects associated with Fipio communication

Subject of this Section

This Chapter introduces language objects associated with Fipio communication.

What Is in This Chapter?

This chapter contains the following sections:

Section	Topic	Page
9.1	Language Objects and IODDTs for Fipio Communication	148
9.2	The IODDT Type T_GEN_MOD Applicable to All Modules	159

Section 9.1
Language Objects and IODDTs for Fipio Communication

Subject of this Section

This section presents general information about language objects and IODDTs associated with the Fipio bus.

What Is in This Section?

This section contains the following topics:

Topic	Page
Presentation of Language Objects for Fipio Communication	149
Implicit Exchange Language Objects Associated with the Application-Specific Function	151
Explicit Exchange Language Objects Associated with the Application-Specific Function	152
Management of Exchanges and Reports with Explicit Objects	154
Details of Implicit Exchange Objects for Fipio Communication	158

Presentation of Language Objects for Fipio Communication

General

Fipio communication does not have any specific associated IODDT. The IODDTs that you can use are those IODDTs defined for devices associated with Fipio communication.

All these IODDTs are described in detail in the specific manuals for Fipio devices. The `T_GEN_MOD` IODDT is presented in this manual as it can be used by all devices.

NOTE: IODDT variables can be created in two different ways:

- Using the **I/O objects** *(see EcoStruxure™ Control Expert, Operating Modes)* tab,
- Data Editor.

The following table shows the different IODDTs that are available depending on the Fipio device used:

IODDT	Device							
	Standard profiles	TBX	Waterproof IP65	Waterproof IP67	Fipio/AS-i gateway	Momentum	Lexium	Altivar
T_DIS_IN_STD		X	X	X				
T_DIS_IN_FIP_STD		X	X	X				
T_DIS_OUT_STD		X	X	X				
T_DIS_IN_GEN		X	X	X		X		
T_DIS_OUT_GEN		X	X	X		X		
T_ANA_IN_STD		X						
T_ANA_OUT_STD		X						
T_ANA_IN_GEN		X				X		
T_ANA_OUT_GEN		X						
T_DIS_IN_MOM						X		
T_DIS_OUT_MOM						X		
T_ANA_IN_MOM4						X		
T_ANA_IN_MOM8						X		
T_ANA_IN_MOM16						X		
T_ANA_OUT_MOM4						X		
T_ANA_DIS_IN_OUT_AMM						X		
T_STDP_GEN	X							
T_LEXIUM_FIPIO							X	
T_ATV58F_STD								X
T_ATV58_STD								X
T_DIS_ESF_IP67_x_STD (1)				X				

IODDT	Device							
	Standard profiles	TBX	Waterproof IP65	Waterproof IP67	Fipio/AS-i gateway	Momentum	Lexium	Altivar
T_DIS_EMF_IP67_x_STD (2)				X				
T_DIS_OUT_IP67_STD				X				
T_SAP10_STD					X			
Legend: (1) : x represents number from 0 to 7 (2) : x represents numbers 8, 10 to 13								

Language Object Types

Each IODDT contains a group of language objects which are used to control them and check their operation.

There are two types of language object:

- **Implicit exchange objects**, which are automatically exchanged on each cycle of the task associated with the module,
- **Explicit exchange objects**, which are exchanged when requested to do so by the project, using explicit exchange instructions.

Implicit exchanges concern the states of modules and slaves, etc.

Explicit exchanges are used to set up the module's parameters and for module diagnostics.

Implicit Exchange Language Objects Associated with the Application-Specific Function

At a Glance

An integrated application-specific interface or the addition of a module automatically enhances the language objects application used to program this interface or module.

These objects correspond to the input/output images and software data of the module or integrated application-specific interface.

Reminders

The module inputs (`%I` and `%IW`) are updated in the PLC memory at the start of the task, the PLC being in RUN or STOP mode.

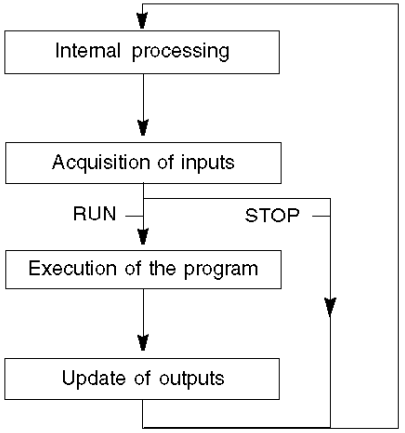
The outputs (`%Q` and `%QW`) are updated at the end of the task, only when the PLC is in RUN mode.

NOTE: When the task occurs in STOP mode, either of the following are possible, depending on the configuration selected:

- outputs are set to fallback position (fallback mode)
- outputs are maintained at their last value (maintain mode)

Figure

The following diagram shows the operating cycle of a PLC task (cyclical execution).



Explicit Exchange Language Objects Associated with the Application-Specific Function

At a Glance

Explicit exchanges are exchanges performed at the user program's request, and using instructions:

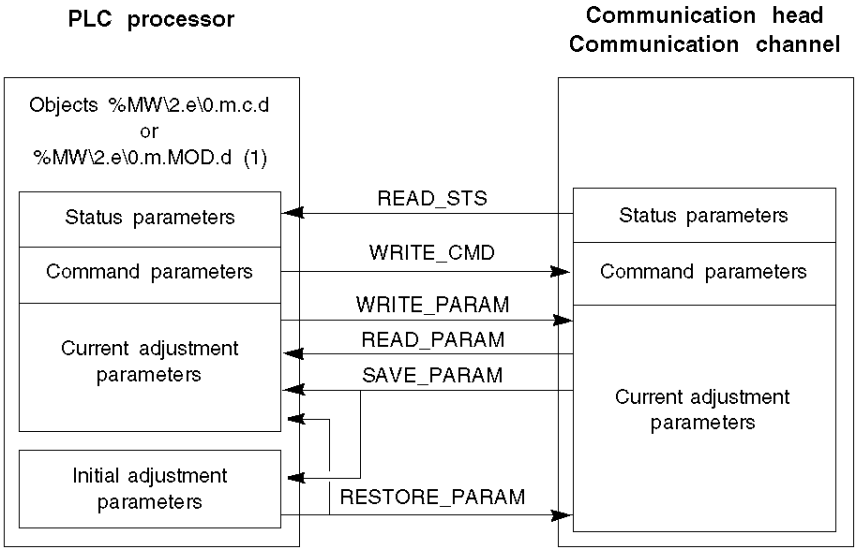
- READ_STS *(see EcoStruxure™ Control Expert, I/O Management, Block Library)* (read status words)
- WRITE_CMD *(see EcoStruxure™ Control Expert, I/O Management, Block Library)* (write command words)
- WRITE_PARAM *(see EcoStruxure™ Control Expert, I/O Management, Block Library)* (write adjustment parameters)
- READ_PARAM *(see EcoStruxure™ Control Expert, I/O Management, Block Library)* (read adjustment parameters)
- SAVE_PARAM *(see EcoStruxure™ Control Expert, I/O Management, Block Library)* (save adjustment parameters)
- RESTORE_PARAM *(see EcoStruxure™ Control Expert, I/O Management, Block Library)* (restore adjustment parameters)

These exchanges apply to a set of %MW objects of the same type (status, commands or parameters) that belong to a channel.

NOTE: These objects provide information about the module (e.g., type of channel fault, etc.), and are used to control the modules and to define their operating modes (saving and restoring currently applied adjustment parameters).

General Principle For Using Explicit Instructions

The diagram below shows the different types of explicit exchanges that can be made between the processor and module.



(1) Only with the instructions READ_STS and WRITE_CMD.

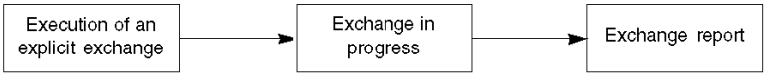
Managing exchanges

During an explicit exchange, it is necessary to check its performance in order that data is only taken into account when the exchange has been correctly executed.

To do this, two types of information are available:

- information concerning the exchange in progress *(see page 156)*
- the exchange report *(see page 157)*

The following diagram describes the management principle for an exchange:



NOTE: In order to avoid several simultaneous explicit exchanges for the same channel, it is necessary to test the value of the word EXCH_STS (`%MWr.m.c.0`) of the IODDT associated to the channel before to call any EF using this channel.

Management of Exchanges and Reports with Explicit Objects

At a Glance

When data is exchanged between the PCL memory and the module, the module may require several task cycles to acknowledge this information. All IODDTs use two words to manage exchanges:

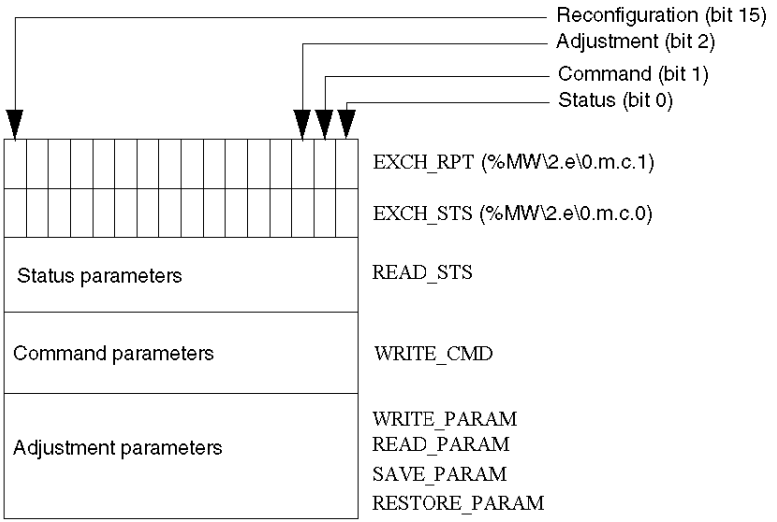
- `EXCH_STS` (%MW\2.e\0.m.c.0): exchange in progress
- `EXCH_RPT` (%MW\2.e\0.m.c.1): report

NOTE: Depending on the localization of the module, the management of the explicit exchanges `(%MW0.0.MOD.0.0 for example)will not be detected by the application:`

- For in-rack modules, explicit exchanges are done immediately on the local PLC Bus and are finished before the end of the execution task. So, the `READ_STS`, for example, is always finished when the `%MW0.0.MOD.0.0` bit is checked by the application.
- For remote bus (Fipio for example), explicit exchanges are not synchronous with the execution task. So, the detection is possible by the application.

Illustration

The illustration below shows the different significant bits for managing exchanges:



Description of the Significant Bits

Each bit of the words EXCH_STS (%MW\2.e\0.m.c.0) and EXCH_RPT (%MW\2.e\0.m.c.1) is associated with a type of parameter:

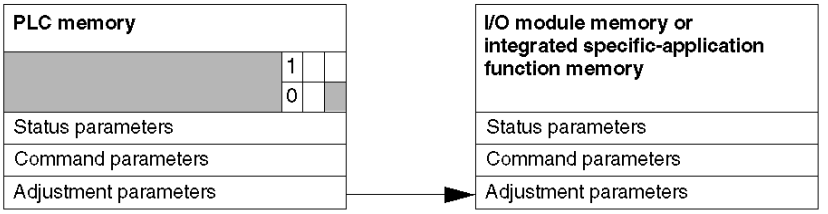
- Rank 0 bits are associated with the status parameters:
  - The STS_IN_PROGR bit (%MW\2.e\0.m.c.0.0) indicates whether a read request for the status words is in progress.
  - The STS_ERR bit (%MW\2.e\0.m.c.1.0) specifies whether a read request for the status words is refused by the module channel.
- Rank 1 bits are associated with the command parameters:
  - The CMD_IN_PROGR bit (%MW\2.e\0.m.c.0.1) indicates whether command parameters are being sent to the module channel.
  - The CMD_ERR bit (%MW\2.e\0.m.c.1.1) specifies whether the command parameters are refused by the module channel.
- Rank 2 bits are associated with the adjustment parameters:
  - The ADJ_IN_PROGR bit (%MW\2.e\0.m.c.0.2) indicates whether the adjustment parameters are being exchanged with the module channel (via WRITE_PARAM, READ_PARAM, SAVE_PARAM, RESTORE_PARAM).
  - The ADJ_ERR bit (%MW\2.e\0.m.c.1.2) specifies whether the adjustment parameters are refused by the module.
    If the exchange is correctly executed, the bit is set to 0.
- Rank 15 bits indicate a reconfiguration on channel c of the module from the console (modification of the configuration parameters + cold start-up of the channel).

NOTE: m stands for the position of the module; **c** stands for the channel number in the module.

NOTE: Exchange and report words also exist at module level EXCH_STS (%MW\2.e\0.m.MOD) and EXCH_RPT (%MW\2.e\0.m.MOD.1) in the IODDT type T_GEN_MOD.

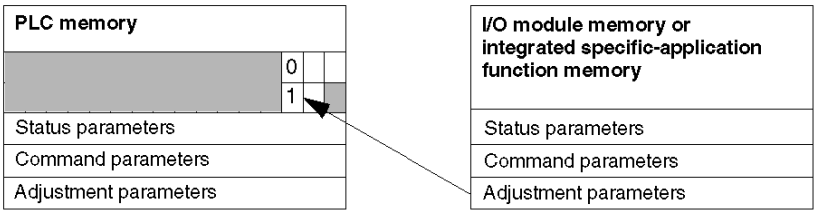
Example

Phase 1: Sending data by using the WRITE_PARAM instruction.



When the instruction is scanned by the PLC processor, the **Exchange in progress** bit is set to 1 in %MW\2.e\0.m.c.

Phase 2: Analysis of the data by the I/O module and report



When the data is exchanged between the PLC memory and the module, processing by the module is managed by the `ADJ_ERR` bit (%MW\2.e\0.m.c.1.2): Report (0 = correct exchange, 1 = faulty exchange).

NOTE: There is no adjustment parameter at module level.

Execution Indicators for an Explicit Exchange: EXCH_STS

The table below shows the control bits of the explicit exchanges: `EXCH_STS` (%MW\2.e\0.m.c.0).

Standard symbol	Type	Access	Meaning	Address
`STS_IN_PROGR`	BOOL	R	Reading of channel status words in progress	%MW\2.e\0m.c.0.0
`CMD_IN_PROGR`	BOOL	R	Command parameters exchange in progress	%MW\2.e\0m.c.0.1
`ADJ_IN_PROGR`	BOOL	R	Adjust parameters exchange in progress	%MW\2.e\0m.c.0.2
`RECONF_IN_PROGR`	BOOL	R	Reconfiguration of the module in progress	%MW\2.e\0.m.c.0.15

NOTE: If the module is not present or is disconnected, explicit exchange objects (Read_Sts for example) are not sent to the module (STS_IN_PROG (%MWr.m.c.0.0) = 0), but the words are refreshed.

Explicit Exchange Report: EXCH_RPT

The table below shows the report bits: `EXCH_RPT` (%MW\2.e\0.m.c.1).

Standard symbol	Type	Access	Meaning	Address
`STS_ERR`	BOOL	R	Error in reading status words of the channel (1 = failure)	%MW\2.e\0.m.c.1.0
`CMD_ERR`	BOOL	R	Error when exchanging command parameters (1 = failure)	%MW\2.e\0.m.c.1.1
`ADJ_ERR`	BOOL	R	Error when exchanging adjustment parameters (1 = failure)	%MW\2.e\0.m.c.1.2
`RECONF_ERR`	BOOL	R	Fault when reconfiguring the channel (1 = failure)	%MW\2.e\0.m.c.1.15

Details of Implicit Exchange Objects for Fipio Communication

At a Glance

The following table presents all the implicit exchange objects for Fipio communication that can be displayed or modified by the application program.

Bit Objects

The table below shows the different bit objects:

Symbol	Type	Access	Meaning	Address
At user's initiative	EBOOL	R	Image of physical inputs of module (acquisition of inputs).	%I\2.e\0.m.c.d
At user's initiative	EBOOL	R	Image of physical outputs of the module (output commands).	%Q\2.e\0.m.c.d

Word Objects

The table below shows the different word objects:

Symbol	Type	Access	Meaning	Address
At user's initiative	INT	R	Image of physical inputs of module (acquisition of inputs).	%IW\2.e\0.m.c.d
At user's initiative	INT	R	Image of physical outputs of the module (output commands).	%QW\2.e\0.m.c.d

Section 9.2
The IODDT Type T_GEN_MOD Applicable to All Modules

Details of the Language Objects for the IODDT of Type T_GEN_MOD

At a Glance

All Premium PLC modules have an associated IODDT of type `T_GEN_MOD`.

Observations

- In general, the meaning of the bits is given for bit status 1. In specific cases an explanation is given for each status of the bit.
- Not all bits are used.

List of Objects

The table below presents the objects of the IODDT:

Standard symbol	Type	Access	Meaning	Address
`MOD_ERROR`	BOOL	R	Module error bit.	%I\2.e\0.m.MOD.ERR
`EXCH_STS`	INT	R	Module exchange control word.	%MW\2.e\0.m.MOD.0
`STS_IN_PROGR`	BOOL	R	Reading of module status words in progress.	%MW\2.e\0.m.MOD.0.0
`EXCH_RPT`	INT	R	Exchange report word.	%MW\2.e\0.m.MOD.1
`STS_ERR`	BOOL	R	Error when reading status words of the module.	%MW\2.e\0.m.MOD.1.0
`MOD_FLT`	INT	R	Module internal error word.	%MW\2.e\0.m.MOD.2
`MOD_FAIL`	BOOL	R	Internal fault, module unserviceable.	%MW\2.e\0.m.MOD.2.0
`CH_FLT`	BOOL	R	Faulty channel(s).	%MW\2.e\0.m.MOD.2.0
`BLK`	BOOL	R	Terminal block fault.	%MW\2.e\0.m.MOD.2.2
`CONF_FLT`	BOOL	R	Hardware or software configuration error.	%MW\2.e\0.m.MOD.2.5
`NO_MOD`	BOOL	R	Module missing or inoperative.	%MW\2.e\0.m.MOD.2.6
EXT_MOD_FLT	BOOL	R	Internal error word of the module (Fipio extension only).	%MW\2.e\0.m.MOD.2.7
MOD_FAIL_EXT	BOOL	R	Internal fault, module unserviceable (Fipio extension only).	%MW\2.e\0.m.MOD.2.8
CH_FLT_EXT	BOOL	R	Faulty channel(s) (Fipio extension only).	%MW\2.e\0.m.MOD.2.9
BLK_EXT	BOOL	R	Terminal block fault (Fipio extension only).	%MW\2.e\0.m.MOD.2.10

Standard symbol	Type	Access	Meaning	Address
CONF_FLT_EXT	BOOL	R	Hardware or software configuration fault (Fipio extension only).	%MW\2.e\0.m.MOD.2.13
NO_MOD_EXT	BOOL	R	Module missing or inoperative (Fipio extension only).	%MW\2.e\0.m.MOD.2.14

Chapter 10
Fipio communication standard profiles

Subject of this Chapter

This chapter introduces devices compatible with the standard communication profiles for use with the Fipio bus and its services.

What Is in This Chapter?

This chapter contains the following sections:

Section	Topic	Page
10.1	Introduction to standard Fipio communication profiles	162
10.2	Configuration of the Fipio bus standard profiles	167
10.3	Debugging the Fipio bus standard profiles	173
10.4	Language objects associated with the Fipio bus standard profiles	178

Section 10.1
Introduction to standard Fipio communication profiles

Aim of this Section

This Section provides a summary description of the standard profiles of the Fipio bus and its associated services.

What Is in This Section?

This section contains the following topics:

Topic	Page
Standard profiles: general	163
Designation of a Standard Profile	164
List of Catalog Standard Profiles Proposed by Control Expert	165

Standard profiles: general

Introduction

A device conforming to the standard Fipio communication profiles can be linked to a Fipio bus, to exchange the input/output variables.

Data exchanges with devices are used to:

- exchange inputs/outputs,
- write configuration or adjustment parameters,
- diagnose a connected device.

There are three types of standard profile:

- FRDP: Fipio Reduced Device Profile
- FSDP: Fipio Simple Device Profile
- FEDP: Fipio Extended Device Profile

Volume of data exchanged

The device profiles, which conform to a standard profile are a function of the number of inputs and outputs to be exchanged.

The table below describes the words managed by the different standard profiles:

Profile	FRD	FSD	FED
Process data input acquisition output control	 2 words 2 words	 8 words 8 words	 32 words 32 words
Configuration (optional)	-	16 words	30 words
Adjust (optional)	-	32 words	30 words
Commands specific command	 -	 -	 8 words
Diagnostics input validity specific status	 1 byte -	 1 byte -	 1 byte 8 words

Designation of a Standard Profile

At a Glance

A device conforming to the standard Fipio communication profiles is defined by assigning one of the standard profiles from the **Standard Profiles** family to it at the Fipio connection point.

The family consists of a list of standard profile references.

Composition of a Reference

Each reference is made up of several significant codes:

- the standard profile used:
  - FRD
  - FSD
  - FED
- the connected device structure:
  - C: Compact
  - M: Modular
- the number and type of objects exchanged with the manager: corresponds to the input/output exchange volume in 16 bit words
  - 2 words
  - 8 words
  - 32 words
- P: corresponds to the presence of configuration or adjustment parameters (optional).

Example of a compact standard profile reference: FED C32 P.

List of Catalog Standard Profiles Proposed by Control Expert

At a Glance

When configuring the Fipio bus, you can connect standard profiles according to the type of device.

Compact Devices

The following table lists the compact standard profiles:

Reference	Designation	Language interface
FRD C2	EQP RED CMPCT 2M	32 bits %I, 32 bits %Q
FRD C2 P	EQP STD CMPCT 2M P	32 %I bits, 32 %Q bits
	STD DEV CMPCT 2W P	16 configuration words, 32 adjustment words
FSD C8	EQP STD CMPCT 8M	8 %IW words, 8 %QW words
	STD DEV CMPCT 8W	
FSD C8 P	EQP STD CMPCT 8M P	8 %IW words, 8 %QW words
	STD DEV CMPCT 8W P	16 configuration words, 32 adjustment words
FED C32	EQP STD CMPCT 32M	32 %IW words, 32 %QW words
	STD DEV CMPCT 32W	
FED C32 P	EQP STD CMPCT 32M P	32 %IW words, 32 %QW words
	STD DEV CMPCT 328W P	30 configuration words, 30 adjustment words

Modular Devices

The following table lists the modular standard profiles:

Reference	Designation	Language interface
FSD M8	EQP STD MOD 8M	8 %IW words, 8 %QW words
	STD DEV MOD 8W	
FSD M8 P	EQP STD MOD 8M P	8 %IW words, 8 %QW words
	STD DEV MOD 8W P	16 configuration words, 32 adjustment words
FED M32	EQP STD MOD 32M	32 %IW words, 32 %QW words
	EXTENDED DEV MOD 32W	
FED M32 P	EQP STD MOD 32M P	32 %IW words, 32 %QW words
	EXTENDED DEV MOD 328W P	30 configuration words, 30 adjustment words

Momentum Devices

With the Momentum product family, an analog input/output module is not available in the catalog. In this case, you need to select one of the OTHER_FxD_[P] standard profile references.

The selection criteria are as follows:

Reference	Language interface
OTHER_FRD	32 %I bits, 32 %Q bits
OTHER_FRDP	
OTHER_FSD	8 %IW words, 8 %QW words
OTHER_FSDP	
OTHER_FED	32 %IW words, 32 %QW words
OTHER_FEDP	

Communicator

The following communicators are available:

Family	Reference	Language interface
STD-P	TSX FPP10	TSX Fipio communication module
Momentum	170 FNT 110 01	TSX Fipio communication module for Momentums

Section 10.2
Configuration of the Fipio bus standard profiles

Aim of this Section

This Section introduces the configuration principles for the Fipio bus standard profiles.

What Is in This Section?

This section contains the following topics:

Topic	Page
Configuration Screen for a Standard Profile	168
How to Modify the Parameters of a Standard Profile	170
Adjustment Mode	172

Configuration Screen for a Standard Profile

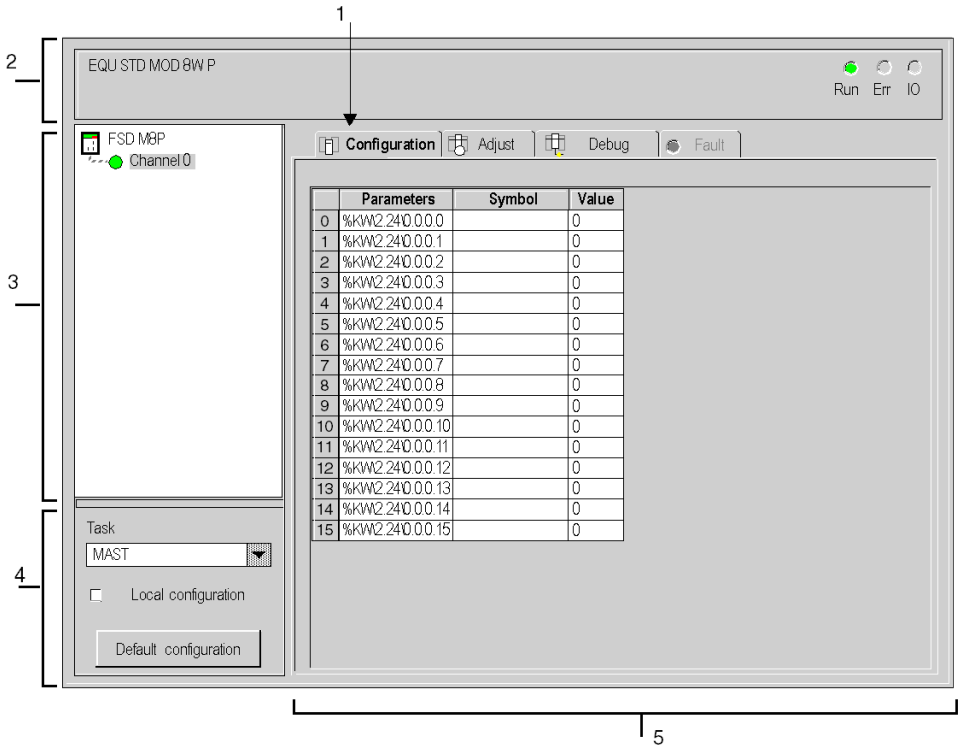
At a Glance

The configuration screen for a standard profile reuses the elements of the devices on the Fipio bus *(see page 105)*.

However, certain parameters are specific to this screen. The remainder of this document only covers the part specific to standard profiles.

Illustration

The figure below represents a configuration screen.



Description

The following table presents the various elements of the configuration screen and their functions.

Number	Element	Function
1	Tabs	The tab in the foreground indicates the mode in progress (**Configuration** in this example). Every mode can be selected using the respective tab. The available modes are: ● **Configuration**, ● **Adjust**, ● **Debug** which can be accessed only in online mode, ● **Fault** (channel level) accessible only in online mode.
2	**Module** area	Gives a reminder of the device's shortened name.
3	**Channel** area	Is used: ● By clicking on the device reference number, to display the tabs: ❍ **Description** which gives the characteristics of the device. ❍ **I/O Objects** *(see EcoStruxure™ Control Expert, Operating Modes)* which is used to presymbolize the input/output objects, ❍ **Fault** which shows the device faults (in online mode). ● **Channel:** single channel for a standard profile. ● **Symbol:** name of the channel defined by the user (using the variable editor).
4	**General parameters** area	This area is used to define the task (**MAST** or **FAST**) in which the channel implicit exchange objects will be exchanged. The **Local configuration** checkbox is only available for the profiles FSD C8 P and FSD M8 P. If the box is checked, the configuration and adjustment parameters are not sent to the device connected to the bus. In this case, the device starts up or initializes itself using its own parameters. If the box is checked in the configuration screen, the channel parameters cannot be accessed from the **configuration** area. The **Default configuration** button is used to return to the initial configuration.
5	**Configuration** area	In cases where the standard profile of %KW configuration words or the %MW adjustment words exist, this area displays them. A **Symbol** and a **Value** are associated to each word. For each **value** the contextual menu can be used to select the display base for the value of the selected word. Three types are available: ● Decimal, ● Hexadecimal, ● Binary.

How to Modify the Parameters of a Standard Profile

At a Glance

The following procedures are applicable for configuration or adjustment parameters.

How to Modify the Display Base

The following procedure describes how to select the display base. In the example, the chosen base is binary.

Step	Action
1	In the **Configuration** area, select the **Value** cell of the word to be modified.
2	Select **Binary** from the contextual menu (right-click). **Result**: the cell displays `2#1011 0000 0000 0000`.

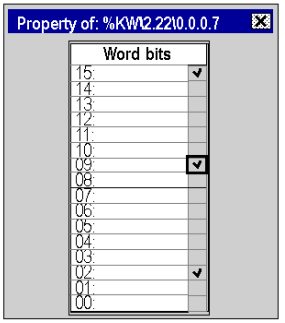
How to Enter a Value Using the Keyboard

The following procedure describes how to enter a value in a cell.

Step	Action
1	In the **Configuration** area, select the **Value** cell of the word to be modified.
2	Enter the value using the keyboard. **Note:** keyboard entry depends on the display base.

How to Enter a Value in Binary Format

The following procedure describes how to enter a value in binary format.

Step	Action
1	Double click on the **Value** cell in the **Configuration** area of the word to be modified. **Result**: the following window appears. 
2	Select the bits to be set to 1.
3	Close the **Properties ...** window by clicking on the cross in the top right-hand corner of the window. **Result**: Closing the window confirms the entered value and it is displayed in the cell format.

Adjustment Mode

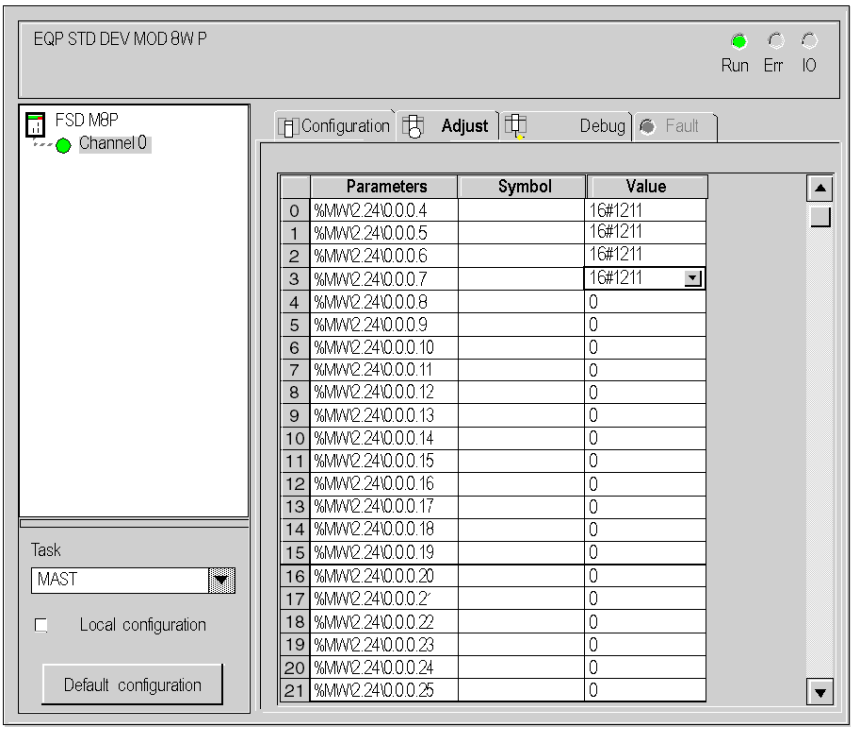
At a Glance

Adjustment mode can be accessed from the **Adjust** tab. The principle is the same as for configuration mode; the adjustment words are %MW words.

There are 32 possible words for FSD and FRD profiles and 30 words for FED profiles.

Illustration

The adjust screen for devices looks like this:



	Parameters	Symbol	Value
0	%MW\2.24\0.0.0.4		16#1211
1	%MW\2.24\0.0.0.5		16#1211
2	%MW\2.24\0.0.0.6		16#1211
3	%MW\2.24\0.0.0.7		16#1211
4	%MW\2.24\0.0.0.8		0
5	%MW\2.24\0.0.0.9		0
6	%MW\2.24\0.0.0.10		0
7	%MW\2.24\0.0.0.11		0
8	%MW\2.24\0.0.0.12		0
9	%MW\2.24\0.0.0.13		0
10	%MW\2.24\0.0.0.14		0
11	%MW\2.24\0.0.0.15		0
12	%MW\2.24\0.0.0.16		0
13	%MW\2.24\0.0.0.17		0
14	%MW\2.24\0.0.0.18		0
15	%MW\2.24\0.0.0.19		0
16	%MW\2.24\0.0.0.20		0
17	%MW\2.24\0.0.0.21		0
18	%MW\2.24\0.0.0.22		0
19	%MW\2.24\0.0.0.23		0
20	%MW\2.24\0.0.0.24		0
21	%MW\2.24\0.0.0.25		0

Section 10.3
Debugging the Fipio bus standard profiles

Aim of this Section

This Section introduces the debugging principles Fipio bus standard profiles.

What Is in This Section?

This section contains the following topics:

Topic	Page
Debug Screen for a Standard Profile	174
How to Modify the Debug Parameters for a Standard Profile	176

Debug Screen for a Standard Profile

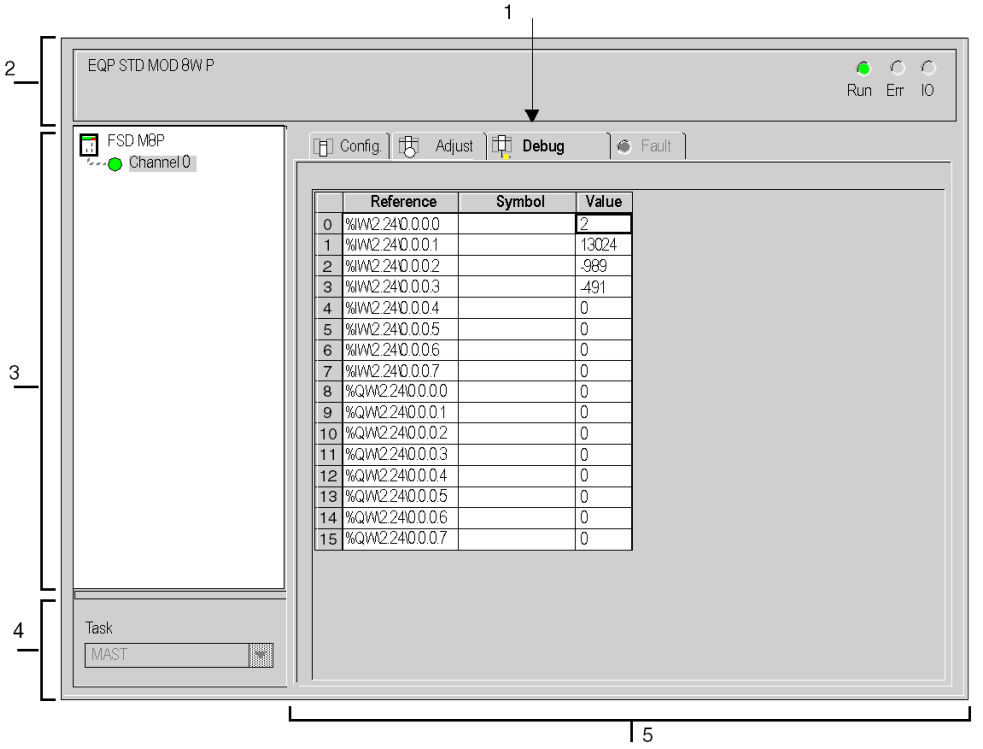
At a Glance

The debug screen for a standard profile reuses the elements of the other devices *(see page 128)* available on the Fipio bus.

However, certain elements are specific to this screen. The remainder of this document only covers the part specific to standard profiles.

Illustration

The figure below represents a debug screen.



Description

The following table shows the different elements of the debug screen and their functions.

Number	Element	Function
1	Tabs	The tab in the foreground indicates the mode in progress (**Debug** in this example). Each mode can be selected using the respective tab. The available modes are: ● **Debug** which can be accessed only in online mode, ● **Fault** (channel level) accessible only in online mode, ● **Configuration**, ● **Adjust**.
2	**Module** area	Gives a reminder of the device's shortened name. In the same area there are 3 LEDs which indicate the module's operating mode: ● **RUN** indicates the operating status of the module, ● **ERR** indicates an internal fault in the module, ● **I/O** indicates a fault from outside the module or an application fault.
3	**Channel** area	Is used: ● By clicking on the reference number, to display the tabs: ❍ **Description** which gives the characteristics of the device. ❍ **I/O Objects** *(see EcoStruxure™ Control Expert, Operating Modes)* which is used to presymbolize the input/output objects, ❍ **Fault** which shows the device faults (in online mode). ● **Channel:** single channel for a standard profile. ● **Symbol:** name of the channel defined by the user (using the variable editor).
4	**General parameters** area	This area is used to display the type of task (**MAST** or **FAST**) in which the channel implicit exchange objects are exchanged.
5	**Parameters in progress** area	In cases where for the standard profile the input bits, the **%IW** input words or the **%QW** output words exist, this area displays them. A **Reference**, a **Symbol** and a **Value** are associated to each word. For each **value** the contextual menu can be used to select the display base for the value of the selected word. Three types are available: ● Decimal, ● Hexadecimal, ● Binary. For FRD ••• standard profile, the **Error** column provides direct access to the diagnostics for each channel when these are faulty (indicated by the indicator lamp built into the diagnostics access button, which turns red).

How to Modify the Debug Parameters for a Standard Profile

At a Glance

The following procedures are applicable for debug parameters.

How to Modify the Display Base

The following procedure describes how to select the display base. In the example, the chosen base is binary.

Step	Action
1	In the **Parameters in progress** area, select the **Value** cell of the word to be modified.
2	Select **Binary** from the contextual menu (right-click). **Result**: the cell displays `2#1011 0000 0000 0000`.

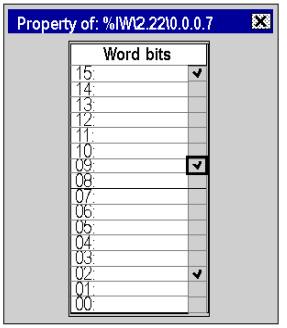
How to Enter a Value Using the Keyboard

The following procedure describes how to enter a value in a cell using the keyboard.

Step	Action
1	In the **Parameters in progress** area, select the **Value** cell of the word to be modified.
2	Enter the value using the keyboard. **Note:** keyboard entry depends on the display base.

How to Enter a Value in Binary Format

The following procedure describes how to enter a value in binary format. It is only valid for outputs.

Step	Action
1	Double click on the **Value** cell in the **Parameters in progress** area of the word to be modified. **Result**: the following window appears. 
2	Select the bits to be set to 1.
3	Close the **Properties ...** window by clicking on the cross in the top right-hand corner of the window. **Result**: Closing the window confirms the entered value and it is displayed in the cell format.

Section 10.4
Language objects associated with the Fipio bus standard profiles

Aim of this Section

This section introduces the language objects associated with the Fipio bus standard profiles.

What Is in This Section?

This section contains the following topics:

Topic	Page
Details of the Language Objects for the IODDT of Type T_STDP_GEN	179
Objects Accessible by Standard Profiles	181
Implicit Exchange Language Objects	183
Explicit Exchange Language Objects	184
Language Objects Associated with Configuration	186

Details of the Language Objects for the IODDT of Type T_STDP_GEN

At a Glance

The tables below present the objects for the T_STDP_GEN-type IODDT that are applicable to all Fipio standard profiles.

Error Bit %I\2.e\0.m.c.ERR

The table below presents the error bit %I\2.e\0.m.c.ERR..

Standard Symbol	Type	Access	Meaning	Address
CH_ERROR	BOOL	R	Channel c error bit	%I\2.e\0.m.c.ERR

Execution Indicators of an Explicit Exchange: EXCH_STS

The table below presents the meanings of the exchange control bits of the channel EXCH_STS (%MW\2.e\0.m.c.0).

Standard Symbol	Type	Access	Meaning	Address
STS_IN_PROGR	BOOL	R	Reading of channel status words in progress	%MW\2.e\0.m.c.0.0
CMD_IN_PROGR	BOOL	R	Current command parameters exchange in progress	%MW\2.e\0.m.c.0.1
ADJ_IN_PROGR	BOOL	R	Adjustment parameters exchange in progress	%MW\2.e\0.m.c.0.2
RECONF_IN_PROGR	BOOL	R	Reconfiguration in progress	%MW\2.e\0.m.c.0.15

Explicit Exchange Report: EXCH_RPT

The table below presents the meaning of the exchange report bits EXCH_RPT (%MW\2.e\0.m.c.1).

Standard Symbol	Type	Access	Meaning	Address
STS_ERR	BOOL	R	Reading error for channel status words	%MW\2.e\0.m.c.1.0
CMD_ERR	BOOL	R	Error during command parameter exchange	%MW\2.e\0.m.c.1.1
ADJ_ERR	BOOL	R	Error during adjustment parameters exchange	%MW\2.e\0.m.c.1.2
RECONF_ERR	BOOL	R	Error when configuring the channel	%MW\2.e\0..m.c.1.15

Standard Channel Faults, CH_FLT

The following table presents the meanings of the bits of the CH_FLT status word (%MW\2.e\0.m.c.2). The reading is performed by a READ_STS (IODDT_VAR1).

Standard Symbol	Type	Access	Meaning	Address
INTERNAL_FLT	BOOL	R	Internal error or channel self-testing	%MW\2.e\0.m.c.2.4
CONF_FLT	BOOL	R	Hardware or software configuration error	%MW\2.e\0.m.c.2.5
COM_FLT	BOOL	R	Bus communication fault	%MW\2.e\0.m.c.2.6
APPLI_FLT	BOOL	R	Application fault (adjustment or configuration error)	%MW\2.e\0.m.c.2.7

Objects Accessible by Standard Profiles

At a Glance

The following tables list the language objects associated with each profile in the Control Expert catalog.

FRD Profile

The following table relates to FRD profiles

	FRD C2	FRD C2 P
Acquisition of inputs	%I\2.e\0.0.0 to %I\2.e\0.0.31	%I\2.e\0.0.0 to %I\2.e\0.0.31
Output command	%Q\2.e\0.0.0 to %Q\2.e\0.0.31	%Q\2.e\0.0.0 to %Q\2.e\0.0.31
Configuration words	-	%KW\2.e\0.0.c.0 to %KW\2.e\0.0.c.15
Status: exchange management	%MW\2.e\0.0.c.0	%MW\2.e\0.0.c.0
Exchange report	%MW\2.e\0.0.c.1	%MW\2.e\0.0.c.1
Status: standard to channel	%MW\2.e\0.0.c.2	%MW\2.e\0.0.c.2
Input validity	%MW\2.e\0.0.c.3	%MW\2.e\0.0.c.3
Status: specific to channel	-	-
Command words	-	-
Adjustment words	-	%MW\2.e\0.0.c.4 to %MW\2.e\0.0.c.35
Standard profiles only have channel 0 (**c =0**).		

FSD Profile

The following table relates to FSD profiles

	FSD C/M8	FSD C/M8 P
Acquisition of inputs	%IW\2.e\0.0.0 to %IW\2.e\0.0.7	%IW\2.e\0.0.0 to %IW\2.e\0.0.7
Output command	%QW\2.e\0.0.0 to %QW\2.e\0.0.7	%QW\2.e\0.0.0 to %QW\2.e\0.0.7
Configuration words	-	%KW\2.e\0.0.c.0 to %KW\2.e\0.0.c.15
Status: exchange management	%MW\2.e\0.0.c.0	%MW\2.e\0.0.c.0

	FSD C/M8	FSD C/M8 P
Exchange report	%MW\2.e\0.0.c.1	%MW\2.e\0.0.c.1
Status: standard to channel	%MW\2.e\0.0.c.2	%MW\2.e\0.0.c.2
Input validity	%MW\2.e\0.0.c.3	%MW\2.e\0.0.c.3
Status: specific to channel	-	-
Command words	-	-
Adjustment words	-	%MW\2.e\0.0.c.4 to %MW\2.e\0.0.c.35
Standard profiles only have channel 0 (**c =0**).		

FED profile

The following table relates to FED profiles

	FED C/M32	FED C/M32 P
Acquisition of inputs	%IW\2.e\0.0.0 to %IW\2.e\0.0.31	%IW\2.e\0.0.0 to %IW\2.e\0.0.31
Output command	%QW\2.e\0.0.0 to %QW\2.e\0.0.31	%QW\2.e\0.0.0 to %QW\2.e\0.0.31
Configuration words	-	%KW\2.e\0.0.c.0 to %KW\2.e\0.0.c.29
Status: exchange management	%MW\2.e\0.0.c.0	%MW\2.e\0.0.c.0
Exchange report	%MW\2.e\0.0.c.1	%MW\2.e\0.0.c.1
Status: standard to channel	%MW\2.e\0.0.c.2	%MW\2.e\0.0.c.2
Input validity	%MW\2.e\0.0.c.3	%MW\2.e\0.0.c.3
Status: specific to channel	%MW\2.e\0.0.c.4 to %MW\2.e\0.0.c.11	%MW\2.e\0.0.c.4 to %MW\2.e\0.0.c.11
Command words	%MW\2.e\0.0.c.12 to %MW\2.e\0.0.c.19	%MW\2.e\0.0.c.12 to %MW\2.e\0.0.c.19
Adjustment words	-	%MW\2.e\0.0.c.20 to %MW\2.e\0.0.c.49
Standard profiles only have channel 0 (**c =0**).		

Implicit Exchange Language Objects

At a Glance

This page describes all the implicit exchange language objects associated with standard profiles for Fipio communication that can be displayed or modified by the application program.

Bit Objects

The table below shows the different bit objects for implicit exchange.

Object (1)	Function	Meaning
%I\2.e\0.0.MOD.ERR	Module error bit	This bit set to 1 indicates a module error (at least one of the channels has an error, for instance)
%I\2.e\0.m.c.ERR	Channel error bit	This bit set to 1 indicates that a channel is faulty.
%I\2.e\0.0.0 to %I\2.e\0.0.31	Bit exchange	Acquisition of input bits for FRD profiles
%Q\2.e\0.0.0 to %Q\2.e\0.0.31	Bit exchange	Acquisition of output bits for FRD profiles

Word Objects

The following table shows the various implicit exchange word objects.

Object (1)	Function	Meaning
%IW\2.e\0.0.0.0 to %IW\2.e\0.0.0.7	FSD word exchange	Acquisition of input words for FSD profiles
%QW\2.e\0.0.0.0 to %QW\2.e\0.0.0.7	FSD word exchange	Acquisition of output words for FSD profiles
%IW\2.e\0.0.0.0 to %IW\2.e\0.0.0.31	FED word exchange	Acquisition of input words for FED profiles
%QW\2.e\0.0.0.0 to %QW\2.e\0.0.0.31	FED word exchange	Acquisition of output words for FED profiles

Explicit Exchange Language Objects

At a Glance

This page describes all the explicit exchange language objects associated with standard profiles for Fipio communication that can be displayed or modified by the application program.

Word Objects Common to All Profiles

The table below shows the different explicit exchange word objects.

Object	Function	Meaning
%MW\2.e\0.0.MOD.2	Module status	Least significant byte: corresponds to the base. ● bit 0 = 1: internal fault ● bit 1 = 1: functional fault ● bit 2 = 1: terminal block fault ● bit 3 = 1: base module in self-test phase ● bit 4 = 1: reserved (= 0) ● bit 5 = 1: configuration error ● bit 6 = 1: module missing ● bit 7 = 1: extension module fault Most significant byte: corresponds to the extension. ● bit 8 = 1: internal fault ● bit 9 = 1: functional fault ● bit 10 = 1: terminal block fault ● bit 11 = 1: extension module in self-test phase ● bit 12 = 1: reserved (= 0) ● bit 13 = 1: configuration error ● bit 14 = 1: module missing ● bit 15 = 1: reserved (= 0)
%MW\2.e\0.0.c.2	Base module channel status	Least significant byte: corresponding to the base ● bit 0 = 1: external fault 4 ● bit 1 = 1: external fault 3 ● bit 2 = 1: external fault 2 ● bit 3 = 1: external fault 1 ● bit 4 = 1: internal fault ● bit 5 = 1: hardware configuration fault ● bit 6 = 1: communication fault ● bit 7 = 1: application fault
%MW\2.e\0.0.c.3	Input validity	For FRD, FSD and FED profiles, the least significant byte is specific to the connected device. Refer to the device documentation.

NOTE: For each connected device, please refer to the corresponding documentation for details of the application-specific status words.

Internal Words for FRD and FSD Profiles

The following table describes the internal words specific to FRD and FSD profiles:

Object	Function	Meaning
%MW\2.e\0.0.c.4 to %MW\2.e\0.0.c.35	Adjustment parameters	32 specific adjustment words. Parameters accessible in read and write using the `WRITE_PARAM`, `READ_PARAM`, `SAVE_PARAM`, and `RESTORE_PARAM` functions.

Internal Words for FED Profiles

The following table describes the internal words specific to FED profiles:

Object	Function	Meaning
%MW\2.e\0.0.c.4 to %MW\2.e\0.0.c.11	Specific status	8 specific status words. Parameters accessible in read using the `READ_STATUS` function.
%MW\2.e\0.0.c.12 to %MW\2.e\0.0.c.19	Specific channel level command	32 specific command words. Parameters accessible in write using the `WRITE_CMD` function.
%MW\2.e\0.0.c.20 to %MW\2.e\0.0.c.49	Adjustment parameters	32 specific adjustment words. Parameters accessible in read and write using the `WRITE_PARAM`, `READ_PARAM`, `SAVE_PARAM`, and `RESTORE_PARAM` functions.

Language Objects Associated with Configuration

At a Glance

This page describes the configuration words associated with devices used for Fipio communication.

Internal Constants

The following table describes the internal constants:

Object	Function	Meaning
%KW\2.e\0.0.0.0 to %KW\2.e\0.0.0.15	Channel level configuration words	Contain the parameters of channels defined using the configuration editor for FRD and FSD profiles.
%KW\2.e\0.0.0.0 to %KW\2.e\0.0.0.29	Channel level configuration words	Contain the parameters of channels defined using the configuration editor for FED profiles.

Chapter 11
Fipio Agent

Subject of this Chapter

This Chapter introduces the Agent devices on the Fipio bus and its services.

What Is in This Chapter?

This chapter contains the following sections:

Section	Topic	Page
11.1	Introduction to Fipio Agents	188
11.2	Configuration of Fipio Agents	194
11.3	Debugging Fipio Agents	199
11.4	Language objects associated with the Fipio Agents	201

Section 11.1
Introduction to Fipio Agents

Aim of this Section

This section provides a summary description of the agents on the Fipio bus and their associated services.

What Is in This Section?

This section contains the following topics:

Topic	Page
Fipio Agent: General	189
Periodic Data Exchange	190
Special Cases	191
Communication from a Fipio Agent	192

Fipio Agent: General

At a Glance

Micro or Premium PLCs, equipped with a TSX FPP 10 PCMCIA card on their built-in communication channel act as agents on the Fipio bus.

The Fipio Agent function can be used for two types of data exchange:

- periodic exchanges of information with the manager (of shared variable type),
- aperiodic message exchanges with the manager or with other agents.

NOTE: This function is available with version V1.9 or later of the TSX FPP 10 card.

NOTE: The periodic cycle time of Fipio agents must be at least 5ms but the value depends on the configuration.

Periodic Data Exchange

At a Glance

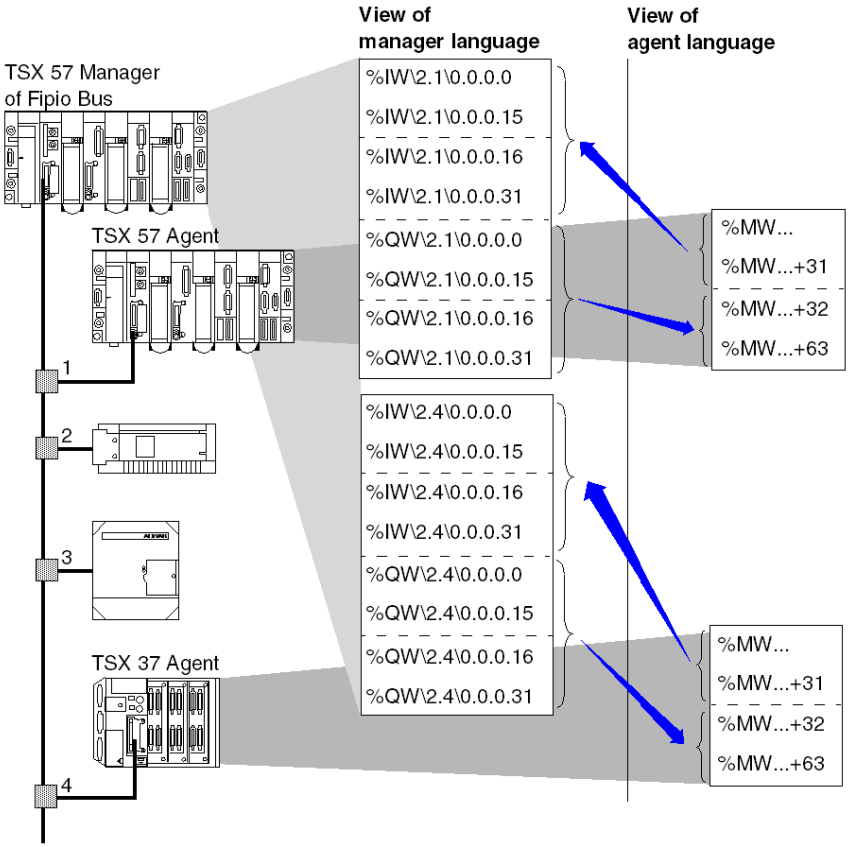
Each Fipio Agent PLC uses 64 consecutive %MW words to exchange periodic data.

- The first 32 words are reserved for sending the information to the manager.
- The next 32 words are reserved for receiving information.

NOTE: The %MW status words and the %MW command words are not significant.

View of Exchanges

The following diagram shows the exchanges between a Fipio Agent and a Premium manager.



Special Cases

At a Glance

A Premium PLC declared as a Fipio Agent can communicate with a TSX 47-107 or APRIL 5000 manager.

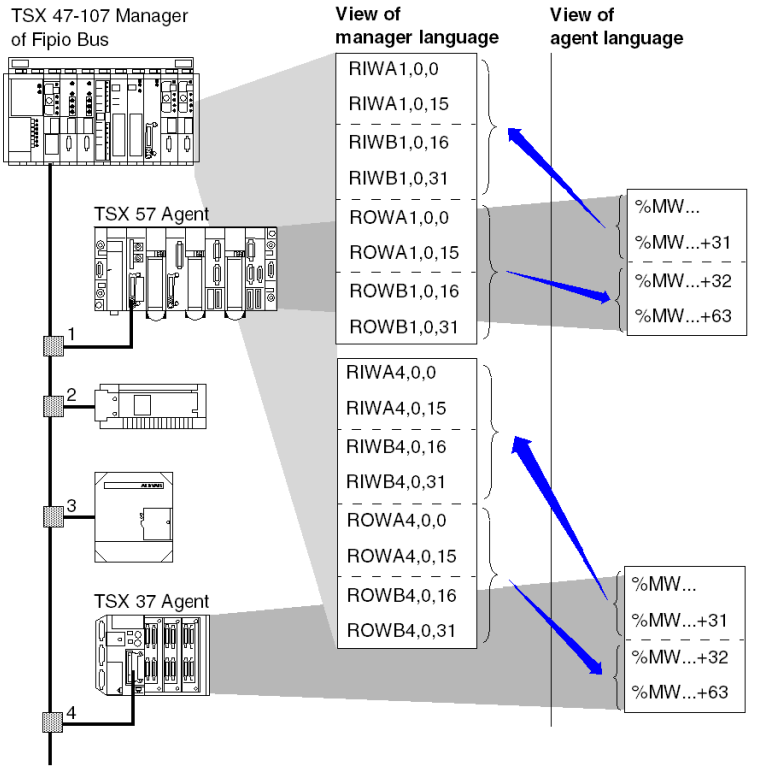
When used in this way, the Fipio Agent is declared on the bus as a standard profile at the connection point.

The standard profile to be declared is as follows:

- Device family: STD_P.
- Base: FED M32.
- Communicator: TSX FPP 10.

Exchanges with a TSX 47-107

The following diagram shows the exchanges between a Fipio Agent and a TSX 47-107 manager.



Communication from a Fipio Agent

At a Glance

A PLC declared as a Fipio Agent can communicate:

- with the bus manager server,
- with the server of another Fipio Agent.

Address Settings

The syntax of an address is as follows:

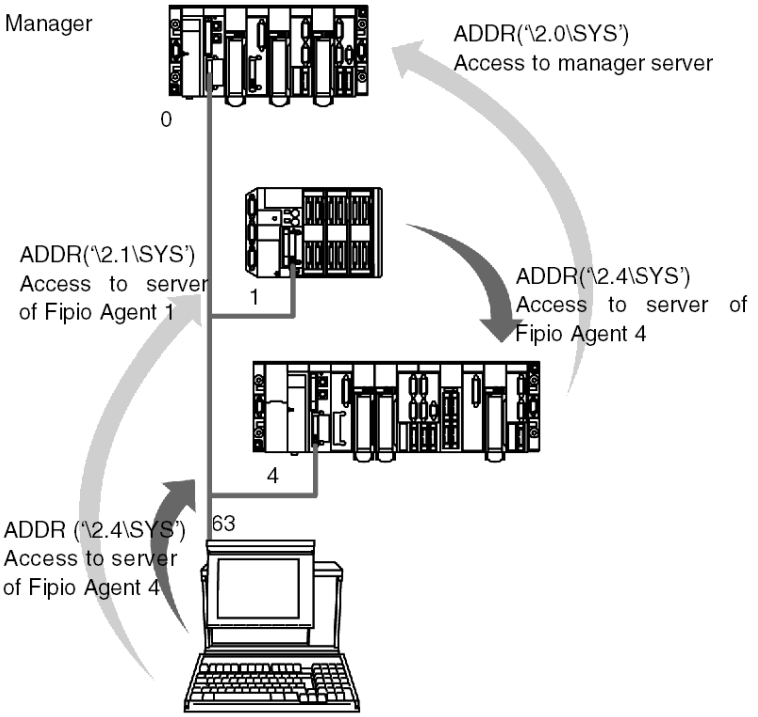
```
ADDR('\2.4\SYS')
```

The following table describes the address parameters:

Parameter	Description
2.4	• 2 : corresponds to the Fipio bus number • 4 : corresponds to the destination connection point
SYS	Corresponds to the server address.

Example

Example of How Different PLCs Access the System



The Fipio Agent PLC at address 1 reads 10 words from the Fipio Agent at address 4.

```
READ_VAR(ADDR('\2.4\SYS'), '%MW',0,10, %MW100:4,%MW10:10)
```

Section 11.2
Configuration of Fipio Agents

Aim of this Section

This Section introduces the configuration principles for Fipio Agents.

What Is in This Section?

This section contains the following topics:

Topic	Page
How to Fipio Agent Parameters Using PCMCIA Cards	195
Configuration Screen of a Fipio Agent	197

How to Fipio Agent Parameters Using PCMCIA Cards

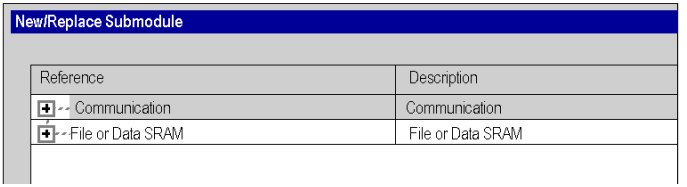
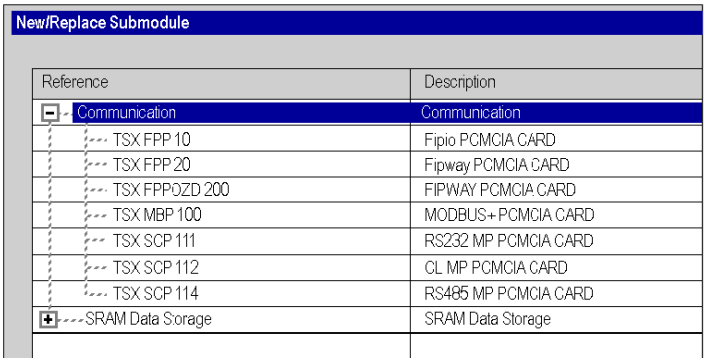
At a Glance

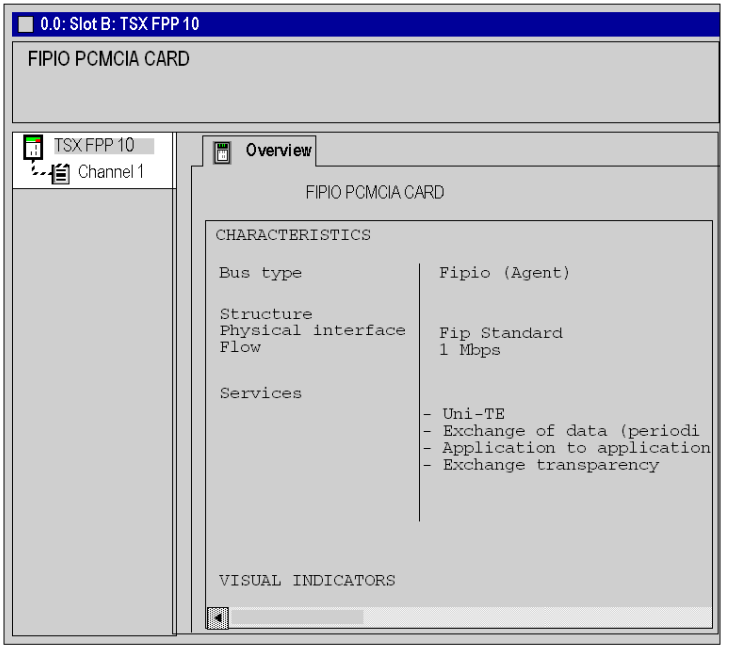
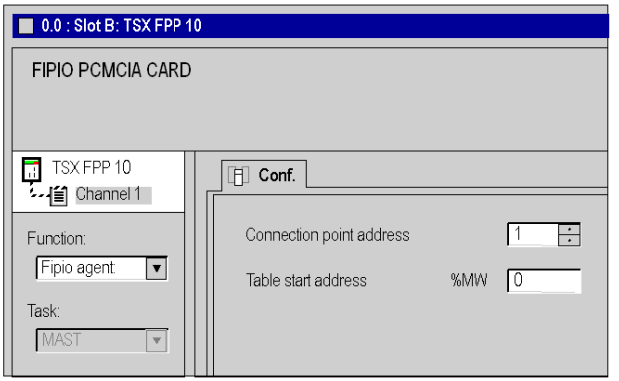
The creation of a project implementing Fipio Agents requires you to configure the communication channel of the PLC declared as a Fipio Agent.

This part describes how to access the configuration parameters of the Fipio Agent link using the TSX FPP 10 PCMCIA card for Micro/Premium PLCs.

How to Declare the Fipio PCMCIA Card

The following table shows the procedure for physically declaring the Fipio PCMCIA card on the PLC:

Step	Action
1	Open the hardware configuration editor.
2	Double-click the PCMCIA communication card slot (bottom slot). **Result:** The card type selection window appears 
3	Expand the **Communication** line by clicking on the **+** sign. **Result:** 
4	Select the TSX FPP 10 Fipio card and then confirm with OK. **Result:** the hardware configuration editor is displayed with the configured card.

Step	Action
5	Double click on the processor's PCMCIA communication card. **Result:** 
6	Select the channel and choose the Fipio AGENT function. **Result:** 

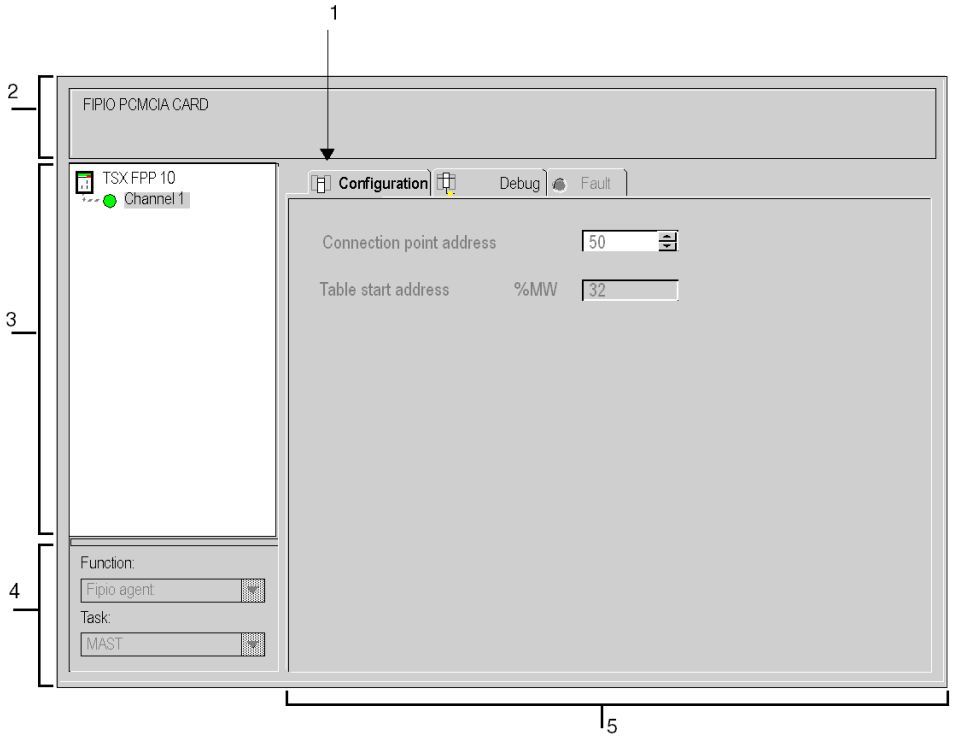
Configuration Screen of a Fipio Agent

At a Glance

This screen, separated into two areas, is used to declare the communication channel and to configure the necessary parameters for a Fipio Agent link.

Illustration

The following figure shows an example of a Fipio Agent configuration screen.



Description

The following table shows the various elements of the configuration screen and their functions.

Number	Element	Function
1	Tabs	The tab in the foreground indicates the mode in progress (**Configuration** in this example). Every mode can be selected using the respective tab. The available modes are: ● **Configuration**, ● **Debug** which can be accessed only in online mode. ● **Fault** (channel level) accessible only in online mode.
2	**Module** area	Gives a reminder of the device's shortened name.
3	**Channel** area	Is used: ● By clicking on the device reference number, to display the tabs: ❍ **Description** which gives the characteristics of the device. ❍ **I/O Objects** *(see EcoStruxure™ Control Expert, Operating Modes)* which is used to presymbolize the input/output objects. ❍ **Fault** which shows the device faults (in online mode). ● By clicking on the **Channel**, to select the channel to be debugged. To the left of the symbol there is a copy of the **CHx** channel LED.
4	**General parameters** area	This area displays the task (**MAST** or **FAST**) in which the channel's implicit exchange objects will be exchanged. The **function** menu is used to select the **Fipio Agent** function. **None** is selected by default.
5	**Configuration** area	Used to select and enter the following parameters: ● the address of the connection point on the Fipio bus, ● the address of the start of the table.

Table Start Address

This parameter indicates the table address to be reserved for periodic data exchanges between the bus manager and the Fipio Agent.

The table contains 64 words:

- 32 words for the inputs (%IW\2.e\0.0.1.d),
- 32 words for the outputs (%QW\2.e\0.0.1.d).

Section 11.3
Debugging Fipio Agents

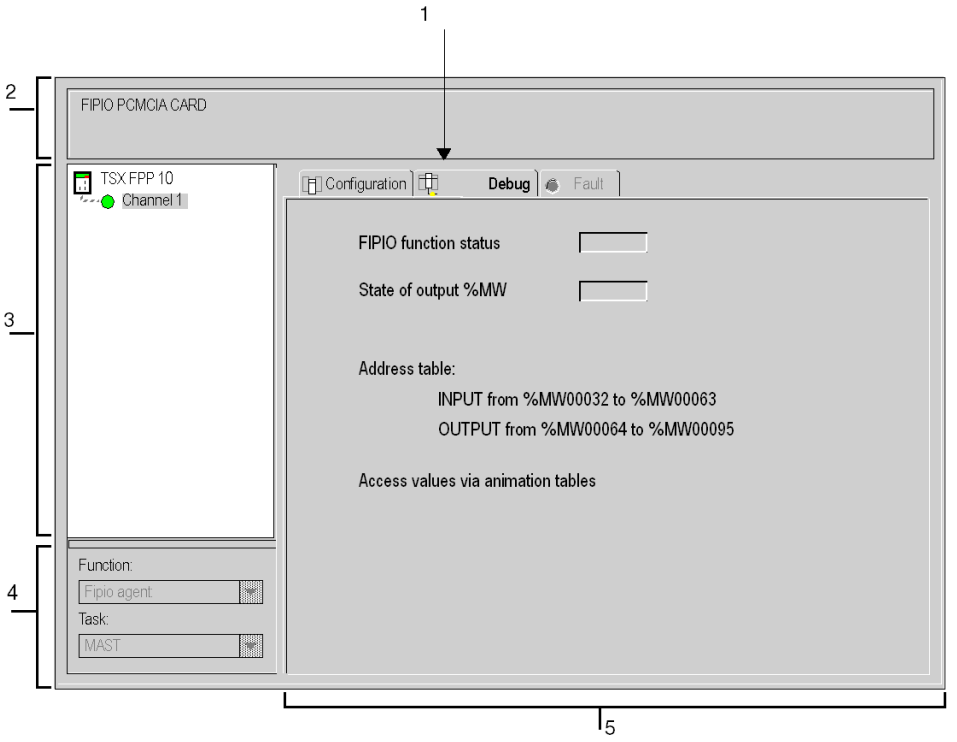
Debug Screen of a Fipio Agent

At a Glance

This screen, separated into two areas, allows you to debug a Fipio Agent.

Illustration

The figure below represents a debug screen.



Description

The following table shows the different elements of the debug screen and their functions.

Number	Element	Function
1	Tabs	The tab in the foreground indicates the mode in progress (**Debug** in this example). Each mode can be selected using the respective tab. The available modes are: ● **Debug** which can be accessed only in online mode. ● **Fault** (channel level) accessible only in online mode, ● **Configuration**.
2	**Module** area	Gives a reminder of the device's shortened name.
3	**Channel** area	Is used: ● By clicking on the device reference number, to display the tabs: ❍ **Description** which gives the characteristics of the device. ❍ **I/O Objects** *(see EcoStruxure™ Control Expert, Operating Modes)* which is used to presymbolize the input/output objects. ❍ **Fault** which shows the device faults (in online mode). ● By clicking on the **Channel**, to select the channel to be debugged. To the left of the symbol there is a copy of the **CHx** channel LED.
4	**General parameters** area	This area displays the task (**MAST** or **FAST**) in which the channel's implicit exchange objects will be exchanged.
5	**Debug** area	Gives access to the debug parameters of a Fipio Agent. No data can be accessed from this screen.

Section 11.4
Language objects associated with the Fipio Agents

Aim of this Section

This Section introduces the language objects associated with Fipio Agents.

What Is in This Section?

This section contains the following topics:

Topic	Page
Implicit Exchange Language Objects	202
Explicit Exchange Language Objects	203
Explicit Exchange Management and Report	204
Language Objects Associated with Configuration	205

Implicit Exchange Language Objects

At a Glance

This page describes all the implicit exchange language objects associated with agent for Fipio communication that can be displayed or modified by the agent application program.

Bit Objects

The table below shows the different bit objects for implicit exchange.

Object (1)	Function	Meaning
%I\2.e\0.0.MOD.ERR	Module error bit	This bit set to 1 indicates a module error (at least one of the channels has an error, for instance)
%I\2.e\0.0.1.ERR	Channel error bit	This bit set to 1 indicates a line fault.

Word Objects

The table below shows the various implicit exchange word objects.

Object (1)	Function	Meaning
%IW\2.e\0.0.1.0	Agent function Status	Byte 0: ● = 16#01: IDLE Function ● = 16#02: RUN Function ● = 16#03: STOP Function
	Strategy to apply to outputs	Byte 1: ● = 16#01: security ● = 16#02: valid %MW value ● = 16#03: fallback
Key		
(1)	Address 2.e\0.m.c.d ● 2: corresponds to the bus number ● e: corresponds to the connection point number ● 0: corresponds to the (virtual) rack number ● 0: corresponds to the module position number ● 1: corresponds to the channel number ● d: corresponds to the channel data	

Explicit Exchange Language Objects

At a Glance

This page describes all the explicit exchange language objects associated with agent for Fipio communication that can be displayed or modified by the agent application program.

Word Objects

The table below shows the different explicit exchange word objects.

Object (1)	Function	Meaning
%MW\2.e\0.m.MOD.2	Module status	• bit 0 = 1: defective module • bit 1 = 1: functional error (error between the processor and the module, adjustment or configuration error, etc.) • bit 2 = 1: terminal block fault (not connected) • bit 3 = 1: self-tests running • bit 4 = 1: reserved • bit 5 = 1: error in hardware or software configuration (the module present is not that declared in the configuration, the sub-modules are not compatible) • bit 6 = 1: module missing • bit 7 = 1: error in one of the sub-modules
%MW\2.e\0.m.c.2	Channel standard status	Byte 0: • bit 0 = 1: the inputs words are not valid if the manager is in RUN • bit 1 = 1: the output words are not valid if the agent is in RUN • bit 4 = 1: internal fault on the TSX FPP 10 card or card not yet started by the manager • bit 5 = 1: configuration fault (the configured module is different from the one connected) • bit 6 = 1: Fipio communication fault • bit 7 = 1: application fault (%MW table overflow) Byte 1: reserved
Key		
(1)	Address 2.e\0.m.c.d • 2 : corresponds to the bus number • e: corresponds to the connection point number • 0 : corresponds to the (virtual) rack number • m: corresponds to the module position number • c: corresponds to the channel number • d: corresponds to the channel data	

Explicit Exchange Management and Report

At a Glance

This page describes all language objects which manage explicit exchanges.

Word objects

The table below shows the different word objects for the management of explicit exchanges.

Object (1)	Function	Meaning
%MW\2.e\0.m.MOD.0	Module exchanges in progress	• bit 0 = 1: status reading in progress • bit 1 = 1: sending of command parameters to the communication module • bit 2 = 1: sending of adjustment parameters to the communication module
%MW\2.e\0.m.MOD.1	Module report	• bit 1 = 0: command parameters received and accepted by the module • bit 2 = 0: adjustment parameters received and accepted by the module
%MW\2.e\0.m.c.0	Channel exchanges in progress	• bit 0 = 1: status reading in progress • bit 1 = 1: sending of command parameters to the communication channel • bit 2 = 1: sending of adjustment parameters to the communication channel
%MW\2.e\0.m.c.1	Channel report	• bit 1 = 0: command parameters received and accepted by the communication channel • bit 2 = 0: adjustment parameters received and accepted by the communication channel
Key		
(1)	Address 2.e\0.m.c.d • 2 : corresponds to the bus number • e: corresponds to the connection point number • 0 : corresponds to the (virtual) rack number • m: corresponds to the module position number • c: corresponds to the channel number • d: corresponds to the channel data	

Language Objects Associated with Configuration

At a Glance

This page describes all configuration language objects associated with Fipio communication agents which can be displayed by the agent application program.

Internal Constants

The following table describes the internal constants:

Object	Function	Meaning
%KW\2.e\0.m.c.0	Type	Byte 0 = 16#09: for the Fipio Agent function
%KW\2.e\0.m.c.1	Reserved	-
%KW\2.e\0.m.c.2	Connection point address	Byte 0 = 16#01 to 16#7F: corresponds to addresses 1 to 127
%KW\2.e\0.m.c.3	Table start address	Byte 0: Least significant byte of address Byte 1: Most significant byte of address

Index

0-9

170XTS05000, *26*

A

adjusting, *172*

C

channel data structure for all modules
    T_GEN_MOD, *159*
channel data structure for Fipio devices
    T_STDP_GEN, *179*
configuring, *81*
configuring devices, *103*
configuring Fipio Agent, *194*
configuring standard profiles, *167*
Connecting
    TSXFPP10, *36*

D

Debug screen
    Access, *127*
debugging devices, *125*
debugging Fipio Agent, *199*
debugging standard profiles, *173*
Description
    TSX FPP 20, *46*
diagnostics, *131*

F

Fipio Agent, *187*
Fipio management, *117*

P

parameter settings, *147*
    Fipio Agent, *201*
    standard profiles, *178*
programming, *117*

S

standard profiles, *161*

T

T_GEN_MOD, *159*, *159*
T_STDP_GEN, *179*
task parameters, *110*
tasks, *53*
TBXBLP01, *23*
TBXBLP10, *24*
topologie
    IP65, *24*
    IP67, *26*
topologies
    IP20, *23*
TSXEFACC7, *26*
TSXEFACC99, *26*
TSXEFCF01, *26*
TSXEFCF02, *26*
TSXEFCF03, *26*
TSXEFCM01, *26*
TSXEFCM03, *26*
TSXFPACC10, *26*
TSXFPACC12, *23*
TSXFPACC14, *23*
TSXFPACC2, *23*
TSXFPACC3, *23*
TSXFPACC4, *23*
TSXFPACC6, *23*
TSXFPACC7, *23*
TSXFPACC8, *23*
TSXFPC10, *45*

TSXFPCA/CR xxx, *23*
TSXFPCAx00, *26*
TSXFPCCx00, *26*
TSXFPCCxxx, *23*
TSXFPCE030, *23*
TSXFPCFxxx, *24*
TSXFPCG0x0, *26*
TSXFPP10, *32*
consumption, *44*

X

XZ-LG101, *26*
XZ-LG102, *26*