Honeywell

Experion PKS
PROFIBUS Interface Implementation Guide

EPDOC-XX89-en-431A
February 2015

Release 431

Honeywell

Document	Release	Issue	Date
EPDOC-XX89-en-431A	431	0	February 2015

Disclaimer

Contents

1 About this guide

This document provides information about interfacing PROFIBUS devices with an Experion system through Control Builder configuration. This document is intended to capture all information that is relevant to the user pertaining to the integration of a communications interface for Honeywell's Experion Control System to the PROFIBUS-DP network.

Revision history

Revision	Date	Description
A	February 2015	Initial release of the document.

Assumptions about the reader

It is assumed that the users of this document are familiar with the use of the Experion system, including operations such as configuration of Control Modules and I/O modules.

2 PROFIBUS Functional Overview

This section provides a brief overview of PROFIBUS DP technology. It does not describe the details of the interface with the Experion system, which is detailed later in this document. The information in this section is not a comprehensive reference on PROFIBUS and is not a substitute for such, nor can it substitute for the available training materials and courses on PROFIBUS. We assume that the user of the system is familiar with PROFIBUS concepts, devices, and wiring.

Related topics

“Terms and Acronyms” on page 12
“PROFIBUS Communication Profiles” on page 13
“PROFIBUS Application Profiles” on page 14
“PROFIBUS DP Overview” on page 15

2.1 Terms and Acronyms

Attention
The PROFIBUS interface Module SST-PB3-CLX-HWL (SAP item 1120160021), manufactured by Molex Inc. (formerly Woodhead / SST), supersedes models SST-PBF-CLX and SST-PBF-CLX-RLL.

- **CCL** - Control Component Library
- **CB** - Experion Control Builder
- **CEE** - Experion Control Execution Environment
- **I/O** - Input/Output
- **IOM** - Input/Output Module
- **PBIM-SST** - The block template name in the Experion system that represents the SST-PB3-CLX-HWL (PROFIBUS Interface Module) - a plug-in chassis card, model number SST-PB3-CLX-HWL (SAP item 1120160021), which is manufactured by Molex Inc. (formerly Woodhead / SST).
- **PBIM**-Refers to the PBIM-SST block.
- **PROFIBUS** - Process Field Bus
- **PTO** - PROFIBUS Trade Organization
- **TBD** - To be determined
- **Molex Inc. (formerly Woodhead / SST)** - The third-party manufacturer/supplier of the PROFIBUS interface module
- **SST-PB3-CLX-HWL** - The model number of the PROFIBUS interface module, a plug-in, single-wide, chassis card, which acts as a PROFIBUS Master Class 1 device.

2.2 PROFIBUS Communication Profiles

The information below has been extracted from the PROFIBUS Technical Description describing the two communication profiles as defined by the PROFIBUS Standard Definition (EN 50170):

- **PROFIBUS DP** - DP (Decentralized Periphery) is the most frequently used PROFIBUS communication profile. It is optimized for speed, efficiency and low connection costs and is designed especially for communication between automation systems and distributed peripherals.
- **PROFIBUS FMS** - FMS (Fieldbus Messaging Specification) is a universal communication profile for complex communication tasks. FMS offers many sophisticated application functions for communication between intelligent devices.

The Experion integration addresses the use of “PROFIBUS DP Overview” on page 15, thus, no further references will be made to PROFIBUS FMS.

2.3 PROFIBUS Application Profiles

PROFIBUS application profiles (such as the PA profile) are used to define the parameters, function blocks and behavior of complex field devices such as transducers, valves and positioners. PROFIBUS DP does not employ the use of application profiles, and therefore application profiles are not included in the Experion integration.

2.4 PROFIBUS DP Overview

PROFIBUS DP is a master/slave, token passing network, which utilizes a request/response protocol. Basic data exchange operations ensure that on a periodic basis, the master sends an output message to each slave, which responds in turn with an input message. PROFIBUS DP is typically used as an I/O network. As compared to a traditional I/O network architecture that requires dedicated wiring between each I/O module and the controller device, PROFIBUS offers the advantage of a single network/bus on which all I/O peripheral devices reside.

2.4.1 Physical Media

At the physical layer, PROFIBUS DP can be hosted on two transmission media:

- RS-485 - Electrical connection employing a shielded, twisted pair
- Fiber optic

Because the physical interface to Experion employs an electrical connection, at this time, the use of fiber optic media will not be discussed in this document. It is expected however that various commercially available products can be used with the Experion system, which will allow the use of both electrical, and fiber optic media on a PROFIBUS DP network.

Bus Wiring (electrical)

PROFIBUS DP utilizes a 'daisy-chain' bus topology, with a single PROFIBUS cable wired from the master to the first slave and through each slave in the network. 'Branches' can be supported through the use of segments, isolated by repeaters, which are described briefly below.

The electrical wiring media used for PROFIBUS is a shielded twisted pair (2 conductors plus the shield). Specialized cable for PROFIBUS applications is commercially available.

The connectors used are typically a 9 pin Sub-D connector, with pins 3 and 8 used for the positive/negative data signals. Refer to wiring diagrams for the equipment in use for additional details.

The devices at the ends of each segment require active termination, the circuitry for which is generally specified on a per-device basis. Alternatively, PROFIBUS connectors with an integrated termination circuit are commercially available.

Refer to the technical documentation for the device of interest for additional details on wiring and termination.

2.4.2 PROFIBUS DP versus the OSI/ISO Communication model

The ISO 7498 standard identifies seven different layers in a communication protocol. PROFIBUS DP is profiled versus the 7-layer model in the table below.

OSI Reference Model		PROFIBUS DP Component
Layer	***Description***	
1	Physical	RS-485 / Fiber Optic
2	Data Link	Fieldbus Data Link (FDL)
3	Network	Not Used
4	Transport	Not Used
5	Session	Not Used
6	Presentation	Not Defined
7	Application	Not Defined

The presentation (or data-management) layer of the communication model is the layer in which data types and data structures and their related meaning are defined. The fact that the Presentation layer is not defined in

PROFIBUS DP allows device vendors great flexibility in the definition of the data types and data structures that are used to compose the data messages that are used to represent the device.

It is noteworthy that the recently introduced V1 extension to the DP protocol does introduce some standard data types. However, it cannot be assumed that a particular device vendor utilizes these standard data types.

2.4.3 The 'Postal Service' Analogy

An analogy that may be helpful in explaining the content of the prior section is the following: PROFIBUS DP is similar to a 'postal service' for control I/O data. For example, when the postal service delivers an envelope from one location to another, there are no restrictions on the information/data that is recorded inside of the envelope. The envelope could contain a handwritten note in the English language or a typewritten letter in German language, or an infinite number of other possibilities. The postal service is concerned with the size, shape and weight of the letter, not the meaning or format of its contents. PROFIBUS DP is similar in that it is responsible merely for the transport of data messages of known size between specific devices, and is not concerned with the meaning/interpretation of the messages, nor does it require a specific format for such.

2.4.4 Not 'Plug and Play'

Because of the described flexibility in the presentation layer of PROFIBUS DP, a great amount of variety has evolved amongst the vendor space for the representation of data in PROFIBUS DP messages. For example, if a user were to acquire a four channel, 4-20 mA, Analog Input Module, with nearly identical characteristics from several different device vendors, it is unlikely that these modules would represent the physical analog input data in the same data format. Thus, PROFIBUS DP should not be thought of as a 'plug and play' architecture, where all DP devices can seamlessly interact. It is true that they may be able to communicate, but communications itself is of no value if sender and receiver cannot interpret the contents of the messages.

2.4.5 Impact to Experion

It is because of the described flexibility in the presentation layer that the Experion control system (and/or the configuration engineer) must have intimate knowledge of the size, structure and meaning of any data structures, which it sends and receives to/from particular devices. Therefore, it should not be assumed that any/all PROFIBUS DP devices could be effectively integrated using the Experion/PROFIBUS interface. It is recommended that the device of interest is researched and profiled versus the block architecture, as described below.

2.4.6 Device Profiles

Because of the lack of definition at the presentation layer, the PROFIBUS Trade Organization (PTO) has defined a set of device profiles which provide some level of standardization at this level for certain complex devices. Because these profiles are not formally a part of the PROFIBUS protocol definition, they are not considered a part of the PROFIBUS communication model depicted above. However, for certain devices these device profiles provide some degree of standardization at the data management layer. Note that device vendors are not required to utilize these profiles. The set of available profiles include the following:

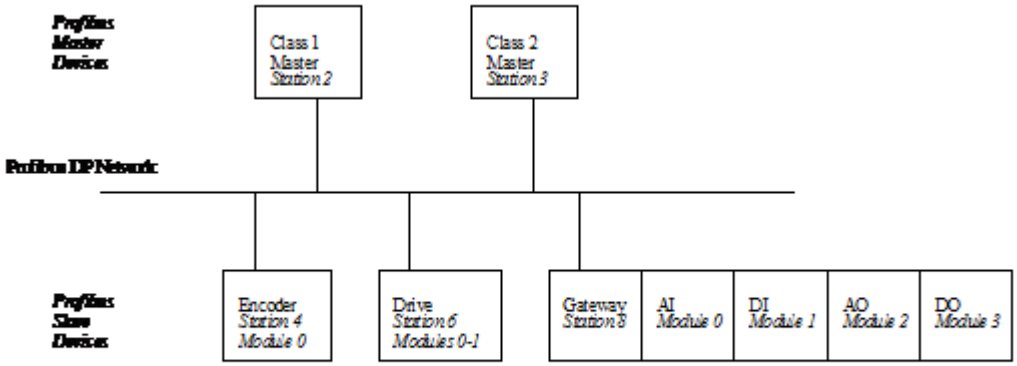
- Profile for communication between controllers
- Profile for process control devices
- Profile for NC/RC controllers (robotics)
- Profile for variable speed drives
- Profile for Encoders
- Profile for HMI systems
- Profile for safety

2.4.7 PROFIBUS DP Network Topology

Several unique types of devices can exist on a PROFIBUS network. The sections below provide a brief summary of the terminology.

Typical PROFIBUS DP Topology Diagram

The following diagram depicts the basic components in a typical PROFIBUS DP network, and a possible set of station and module number assignments. Specific definitions are cited in the sections to follow.



Notes on the PROFIBUS topology diagram:

- Individual slave devices can utilize multiple 'virtual' modules, as is common in complex devices such as motor-drives (frequency converters).
- In normal cyclic communication operations, the class 1 master communicates with each slave station, not necessarily directly with each module. In the example of the modular I/O station (#8), the Class 1 master exchanges I/O data with the station or 'gateway module', which in turn disseminates the appropriate portions of the message to/from each module. In this example, each PROFIBUS (data) module is equivalent to one physical module.

Stations

A station is any node on the network with a unique PROFIBUS physical address, which includes master devices, slave devices, communication interfaces/gateways and segment repeaters. Up to 126 unique stations, can be configured on a PROFIBUS DP network. The valid station address range spans from 0 to 125.

Segments

Although 126 stations can be supported on a single PROFIBUS DP network, the network must be further subdivided into segments, where up to 32 stations can exist on a segment. Segments are isolated by repeaters. Segments can also be used to build 'branches' (or splitting) the PROFIBUS network.

Master Stations/Devices

A DP Master Class 1 (DPM1) device serves as a distribution/collection point for the input/output messages on the network, respectively. Multiple class 1 master devices can exist on the same PROFIBUS network. A DPM1 is typically a controller or dedicated interface to a controller.

DP Master Class 2 (DPM2) devices include configuration or operating devices used for configuration, commissioning, maintenance and diagnostics.

Slave Stations/Devices

The PROFIBUS DP slave device is the peripheral I/O device, or a communication interface to it. A slave station can either be a stand-alone device or a modular station. Typical examples of stand-alone slave devices are encoders, and motor drives. A typical example of a modular station is a rack/rail/bus of I/O cards/devices that communicate on PROFIBUS via a dedicated communication interface, which serves as a station on the PROFIBUS network.

Slave Modules

Within the Experion domain, the term 'module' refers to the 'data object' which represents a particular slave device, or in some cases, a portion of a particular slave device. All slave station data is mapped into modules. For example, even stand-alone devices like encoders and motor drives do support one or more modules. For each respective station, the module(s) are identified by module number, starting at module 0. Therefore, both the station address and a module number are necessary to uniquely identify a particular module on the PROFIBUS network.

Slave Device/Module Relationship

The mapping of slave devices to slave modules is not always 1:1. It can more appropriately be described as 1:N, where N >= 1.

During normal runtime data communications, the master communicates with the slave station device, which is responsible for 'bundling' and 'unbundling' all data that is intended for a specific module. The slave station's I/O data messages consist of contiguous portions for each of its PROFIBUS modules. Because of the differences between modular and stand-alone slave devices and the fact that all PROFIBUS data must be identified by the station and module number, the term 'Module' can be used somewhat ambiguously in the context of PROFIBUS slave device configuration. A PROFIBUS module is actually a 'data object', which represents some slave device, or a portion of that device. In order to lessen potential confusion, the concepts of physical and virtual slave modules have been invented and defined below.

'Physical' Modules

With some devices, there exists a one-to-one correspondence between physical devices and PROFIBUS modules. For example, a typical rack/rail I/O system on PROFIBUS may contain several different interchangeable physical I/O modules. When this system is configured on PROFIBUS, each physical device is represented as one 'module' on PROFIBUS.

'Virtual' Modules

With other devices, such as motor drives, a single physical device may be represented as multiple PROFIBUS modules. This simply means that if the device accepts/provides many bytes of I/O data, the data may be split up into multiple parts, each of which is considered a different module. In this context, the term 'module' can be confusing. It can more appropriately be conceived of as a 'virtual module' or 'data object'.

2.4.8 Slave Configuration using GSD Files

PROFIBUS supports a uniform standard for the configuration of PROFIBUS slaves/modules that utilizes GSD files which allows 'open' configuration of PROFIBUS devices. A GSD file is essentially an electronic data description of a slave device. In accordance with a standard format, the GSD file is a text file that is defined and supplied by the device vendor. The file can be imported and interpreted by the DP Master Class 2 device such that it can supply the user with an interface to configure the slave device. Essentially, the GSD file and supporting infrastructure provides a uniform method of defining the configuration profile of a slave device such that any DP Master Class 2 configuration tool can be used to configure any slave device.

2.4.9 Details on PROFIBUS DP

The PROFIBUS network baud rate is configurable as a property of the network and all master devices. The allowable baud rates vary from a minimum of 9.6K bits/sec to a maximum of 12M bits/sec. All slaves on the network must be capable of responding at the rate at which the master is configured. The bus cycle time (the time period between successive scans of each device on the network) is a function of the network baud rate and

the number/types of devices on the network. The scan cycle time is not specified by the user; it is determined as a function of the network baud rate, the number of stations, and the data requirements for each station. However, scan cycles as fast as 1-5 ms are typical. For example, a PROFIBUS DP network of 32 stations, in which each station produces/consumes 32 bits of input/output data, can support a bus cycle time on the order of 1 ms.

The data exchange model employed between Class 1 masters and slave stations is cyclic in nature. When in data-exchange mode, on a periodic basis, the master sends an output message to each slave device that is assigned to it. The slave responds with an input data message. This request/response communication model remains in effect even if the device does not support an input or an output message; a message is still sent, but of a different format, effectively requesting the slave to send its input data. Slave devices do not initiate communication; the master initiates all communication.

The user message size that can be transported by the PROFIBUS network is 1 - 244 bytes (not including transport layer overhead such as message header). Such messages are passed between the (class 1) master and the station. In a modular station, it is the station that is responsible for gathering and distributing the internal portions of the PROFIBUS messages to each respective module. Thus, in a modular station, the sum of the module input data or output data cannot exceed 244 bytes.

For a particular configuration, only one unique input message, one unique output message, and one unique diagnostic message is defined by the device vendor for each device. The definition of these messages is typically supplied in the form of user documentation supplied with the device. The input and output messages are exchanged periodically between the master and slave as described above. The diagnostic message is exchanged on an exception basis only in the following manner: When a slave needs to send diagnostic information, it informs the master of this by setting a particular bit in the message header in an output message. On the next scan cycle, the input/output data-exchange is essentially skipped, utilizing this scan for the passage of the diagnostic message. This ensures that the cyclic nature of the network is maintained while sacrificing one scan/sample of I/O data.

The effective performance of the network is typically specified in the form of its Token Rotation Time, which can be calculated and is a function of the number of devices, the baud rate, and other factors.

3 Experion/PROFIBUS Interface Architecture

The interface of PROFIBUS DP to Experion involves the provision of a communication path from the PROFIBUS DP network to the C200 Hybrid Controller through the use of a dedicated hardware interface card (SST-PB3-CLX-HWL) that acts as the PROFIBUS master class 1 device.

Related topics

"Intended Topology" on page 22
"SST-PB3-CLX-HWL PROFIBUS Interface Module" on page 24
"Experion Block Architecture" on page 26
"Installation Overview" on page 29
"Configuration Overview" on page 30
"System Performance, Capacity and Topology Specifications" on page 33

3.1 Intended Topology

The diagrams below depict examples of the intended Experion/PROFIBUS interface in a typical C200 or C300 hardware topology:

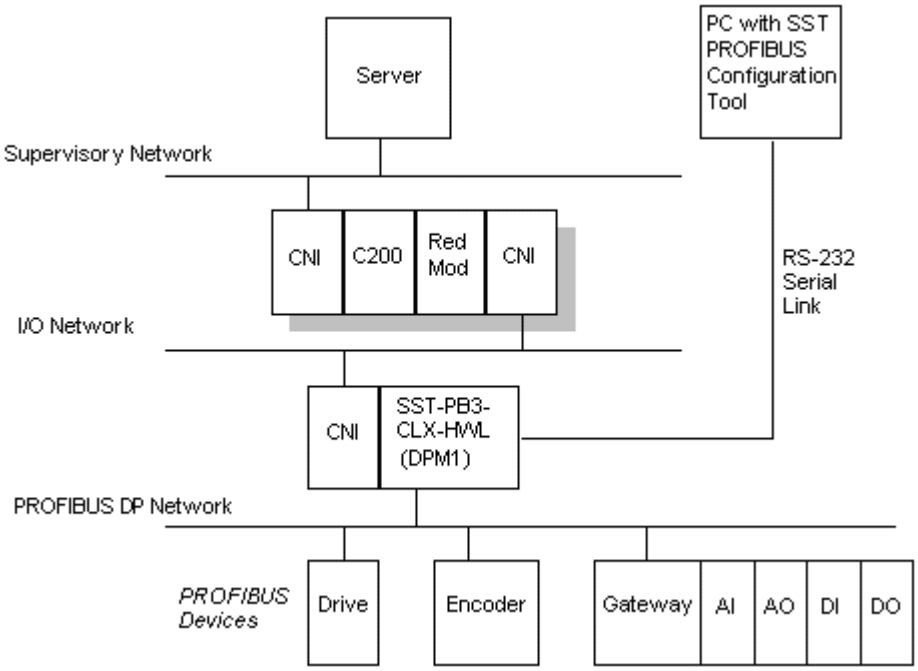


Figure 1: Typical C200 Topology

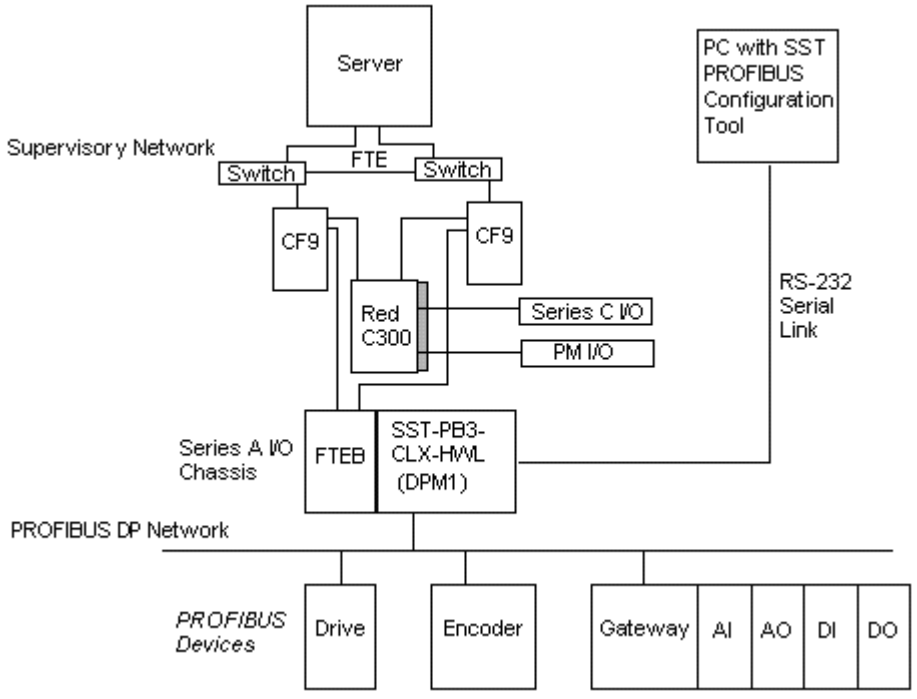


Figure 2: Typical C300 Topology

Notes on the topology diagram:

- The SST-PB3-CLX-HWL is the DPM1 (DP Master Class 1 device).
- The computer with the SST PROFIBUS Configuration Tool is used to configure the SST-PB3-CLX-HWL and the PROFIBUS network.
- Although ControlNet is depicted for the supervisory network, Ethernet supervisory networks are also possible.
- A redundant controller configuration is depicted. Non-redundant configurations are also permitted, thus allowing the SST-PB3-CLX-HWL to reside in the controller chassis.
- Note that PROFIBUS DP supports both stand-alone and modular I/O devices.
- Although only one SST-PB3-CLX-HWL is depicted, one C200 controller can host multiple modules.
- Multiple SST-PB3-CLX-HWL modules can be used either on separate PROFIBUS DP networks or on the same PROFIBUS network (a multi-master configuration).
- The SST-PB3-CLX-HWL is accompanied by a set of SW applications from Molex Inc. (formerly Woodhead Industries Inc. and SST), which are used to configure (and monitor) the PROFIBUS network through a dedicated serial configuration port. As pictured, a dedicated PC can be used for configuration operations, or alternatively, an Experion Station node can be used for this purpose. Note that the SST-PB3-CLX-HWL and PROFIBUS network cannot be configured through Experion Control Builder.

3.2 SST-PB3-CLX-HWL PROFIBUS Interface Module

The integration of PROFIBUS into Experion introduces a new hardware component into the Experion architecture, herein referred to by its model number: SST-PB3-CLX-HWL.

Attention

The PROFIBUS interface Module SST-PB3-CLX-HWL (SAP item 1120160021), manufactured by Molex Inc. (formerly Woodhead / SST), supersedes models SST-PBF-CLX and SST-PBF-CLX-RLL.

3.2.1 Characteristics

The SST-PB3-CLX-HWL possesses the following characteristics:

- Provides a communication bridge between ControlNet and PROFIBUS
- Utilizes the Rockwell 1756 form factor, which is native to Experion
- Is a PROFIBUS DP Class 1 Master and is capable of functioning in ‘multi-master' configurations.
- Supports all standard PROFIBUS baud rates up to 12 M bits/second : 9.6 KBps, 19.2 KBps, 31.25KBps, 45.45Kps, 93.75 KBps, 187.5 KBps, 500 KBps, 1.5 MBps, 3 MBps, 6 MBps, and 12 MBps.
- Is delivered with a supporting set of software utilities used for configuration, monitoring, and documentation/help.
- Supports ROM-based storage of the PROFIBUS configuration, which is preserved through loss-of-power scenarios.
- Supports an RS232 serial link for connection with a PC used for configuration.
- Supports internal realignment (padding) of module messages on byte, word, or long-word boundaries.
- Internally stores the PROFIBUS configuration into flash-ROM, and supports automatic reconfiguration of slave devices on repower.
- Input and output messages from/to the various PROFIBUS station/slave devices are ‘bundled' at the ControlNet level into 2 assemblies (data objects) which are available for transport across ControlNet from/to the C200:
  - As configured with the PROFIBUS network configuration, all input data messages (from PROFIBUS input devices) are packed into a 496 byte **input** assembly. Input data is bound from input devices to the C200.
  - As configured with the PROFIBUS network configuration, all output data messages (from PROFIBUS input devices) are packed into a 492 byte **output** assembly. Output data is bound from the C200 to the output device.

3.2.2 Functional Exclusions

The SST-PB3-CLX-HWL possesses the following noteworthy exclusions in capability:

- Does not currently support the acyclic services associated with the PROFIBUS DP-V1 extension.

3.2.3 SST-PB3-CLX-HWL / PROFIBUS Network Configuration

The PROFIBUS DP network is configured using the Molex Inc. (formerly Woodhead / SST) PROFIBUS Configuration Software Tool that accompanies the SST-PB3-CLX-HWL card. Supporting documentation is provided as well. Therefore, this document will provide only high-level information and will not attempt to duplicate the information and instructions provided by Molex Inc. (formerly Woodhead / SST). Refer to the supporting Molex Inc. (formerlyWoodhead / SST) documentation for the details of configuring the PROFIBUS network or any of the following operations:

- SW/HW installation, including serial port

- Configuration SW usage
- Downloading the configuration to the SST-PB3-CLX-HWL
- Connecting/disconnecting, deriving network status information when connected
- Upgrading Firmware

The SST PROFIBUS Configuration Tool manages the configuration of the PROFIBUS DP network, including the following:

- The import and interpretation of the electronic configuration files (GSD files) that are provided by the device vendor with each slave device.
- The assignment of station and module numbers and the corresponding offsets into the input/output assemblies. The tool ensures that there is no ‘overlapping' of messages to/from individual devices.
- The assignment of the PROFIBUS network baud rate, and other timing parameters.

3.3 Experion Block Architecture

The set of function blocks that comprise the Experion PROFIBUS library are packaged as a Control Component Library (CCL) that is bundled with Experion software releases.

3.3.1 PBUSIF Library

All PROFIBUS blocks are housed in the PBUSIF (PROFIBUS Interface) *library* of the Experion database, as visible on the Library tab of the Experion Control Builder (CB).

3.3.2 Licensing Requirements

Use of the SST-PB3-CLX-HWL module with the PBUSIF library requires the purchase of one of a set of available licenses.

3.3.3 PBIM Block Represents the SST-PB3-CLX-HWL

The SST-PB3-CLX-HWL is represented within the Experion system as an I/O module function block, specifically, the PROFIBUS Interface Module Function Block, or PBIM block. The block template name for the PBIM block in the Experion database is *PBIM-SST*. The following conventions will be adhered to in the remainder of this document:

- 'SST-PB3-CLX-HWL' refers to the physical module/card
- 'PBIM' refers to the PBIM-SST block template in the Experion architecture which represents the actual SST-PB3-CLX-HWL module/card

3.3.4 Function Block Set

The PBUSIF library introduces the following blocks:

- PBIM block, an IOM block representing the SST-PB3-CLX-HWL hardware/module.
- 'Generic' blocks, specifically a generic Device/Module Block and associated generic input and output channel blocks. These blocks can be configured to create an interface to most PROFIBUS DP devices
- Encoder Module/Channel blocks, which conform to the PTO Encoder Profile
- Motor Drive (frequency converter) module/channel blocks, which conform to the PTO PROFIDRIVE profile.
- Siemens Simatic I/O Module blocks: AI, AO, DI, and DO, which are used with the standard I/O channel blocks.
- Siemens Simocode 3UF5x Motor Protection and Control Device module/channel blocks.
- Bizerba Weighing Terminal ST Device module/channel blocks.

These blocks are described in detail in later sections of the document. Additional device-specific blocks may be introduced in the future, dependent upon market demand.

Generic Blocks versus Device Specific Blocks

Device-specific blocks are programmed with intimate knowledge of the data format of the I/O message structure for the respective device. Because of this capability, device-specific blocks are relatively straightforward to configure. Although the set of device specific blocks included in the PROFIBUS library is limited, additional device specific blocks can be added at a future point.

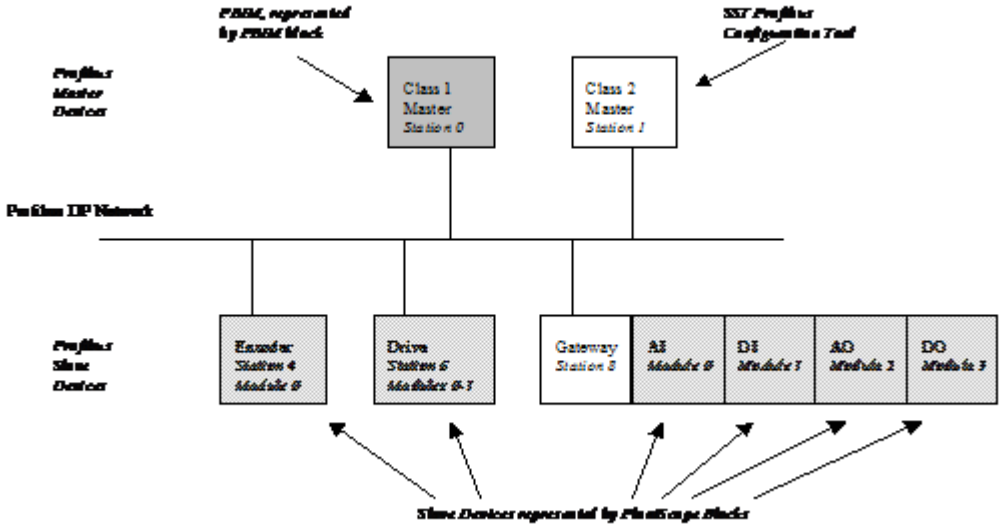
Generic blocks offer the advantage of allowing a flexible configuration model that allows them to provide a simple interface to most devices. This flexibility comes at the cost of requiring the configuration engineer to

research and understand the details of the I/O message structure for the device of interest. This information must be configured into the generic blocks, therefore requiring greater effort to configure than device-specific blocks.

3.3.5 Block Architecture in Relation to PROFIBUS Topology

I/O Module Blocks

The diagram below is an extension of the previous *Typical PROFIBUS DP Topology Diagram* (which depicts a typical PROFIBUS DP topology), but specifically identifies the mapping of Experion blocks over the PROFIBUS topology. Shaded boxes are represented by Experion device/module blocks.



Slave device/module blocks are independent (tagged) blocks that are configured to reference a particular PBIM block (similar to I/O channel assignment to I/O modules.)

Note that the 'Drive' is represented as a single device/module block, even though it supports multiple (virtual) modules.

Both the PBIM block and the device module block are I/O module function blocks, and both are necessary to configure a PROFIBUS slave device/module. Additionally, channel blocks are required as described below.

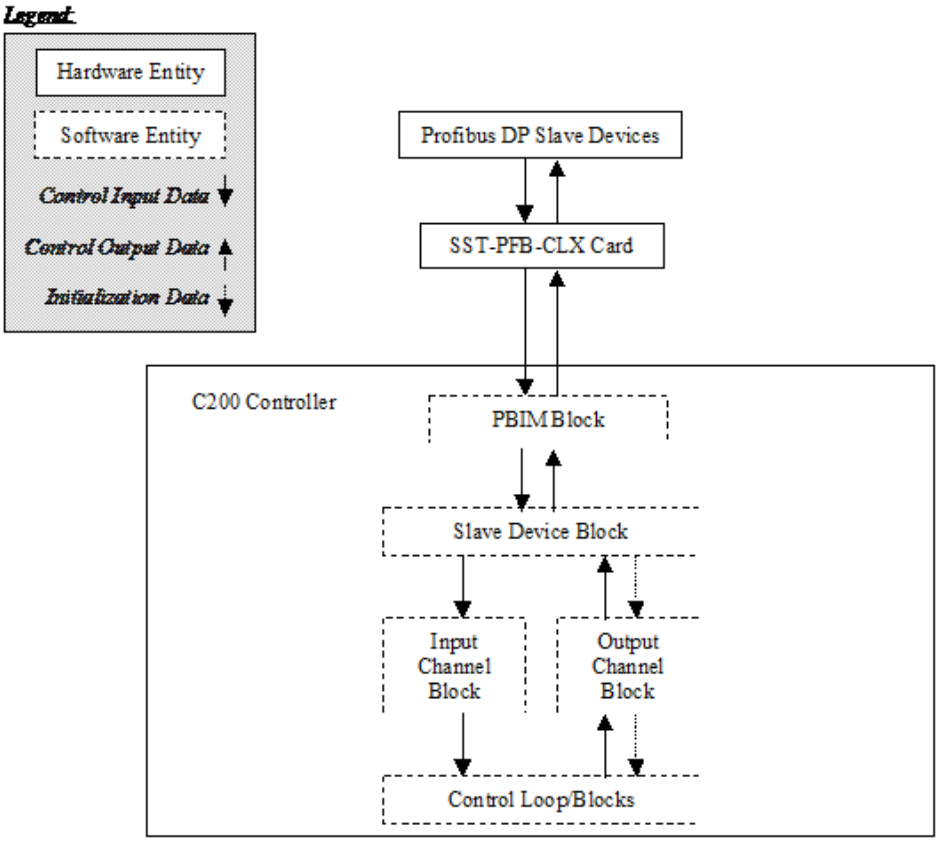
I/O channel blocks

Like existing Experion I/O module blocks, control data is generally not made available through the module block. Instead, I/O channel blocks are 'associated' with the I/O module, and are the blocks through which control data flows in and out of the controller.

In some cases, the existing standard I/O channel blocks are used with PROFIBUS I/O module blocks. In other cases, specific PROFIBUS I/O channel blocks have been developed. Note that some of the channel blocks that have been developed as part of the PROFIBUS library are capable of providing multiple input/output parameters per block.

3.3.6 Data Flow Model

The diagram below depicts the data flow through the various components and blocks in the architecture.



For output modules that support initialization behavior, note that the use of the output channel blocks supports the automatic configuration of the initialization connection.

Three levels of PROFIBUS blocks

Note that I/O data must flow through three levels of blocks:

- PBIM Block
- Slave Device/Module Block
- I/O channel blocks

The presence of three levels of blocks is different than conventional Experion I/O which employs a two-level block hierarchy, specifically module and channel blocks.

Initialization data

In the diagram above, initialization data is pictured as flowing from the slave device/module block to the output channel block, and to the control blocks. In fact, not all PROFIBUS device/module blocks support this initialization data. Furthermore, it must be noted that initialization data does not originate from the device itself; rather, it originates from the module block. Therefore, 'bumpless' initialization cannot be guaranteed given that the output value supplied by the module block could differ from the actual field value.

3.4 Installation Overview

This section is intended to list the high level activities necessary to install the various components in the architecture. This section is not a comprehensive reference on the installation of the various components.

In general, the following installation activities are required prior to the commissioning and use of the Experion/ PROFIBUS interface.

- Install/upgrade Experion Software
- Install Molex Inc. (formerly Woodhead / SST) Profibus Configuration SW on a computer that will be used for configuration of the SST-PB3-CLX-HWL
- Install PROFIBUS hardware (including) on the PROFIBUS network, including SST-PB3-CLX-HWL, PROFIBUS slave station devices, wiring, and termination.

Please refer to the supporting documentation for each of these activities for more detailed instructions.

Attention

Firmware Upgrade Required

The SST-PB3-CLX-HWL card is (currently) shipped from Molex Inc. (formerly Woodhead Industries Inc. / SST).

After acquiring and installing the SST-PB3-CLX-HWL and accompanying Molex Inc. (formerly Woodhead / SST) PROFIBUS Configuration Software, and the latest firmware version, the firmware must be loaded to the SST-PB3-CLX-HWL. Unlike many other Experion components, this operation cannot be accomplished using NTools, nor is it accomplished through the use of the PROFIBUS Configuration Software. Instead, a serial emulation/communication program (such as HyperTerminal) is used to download the new firmware. See the section entitled 'Upgrading the Scanner Firmware' in the SST-PB3-CLX-HWL User's Guide, which is distributed with the SST-PB3-CLX-HWL.

3.5 Configuration Overview

Related topics

"Configuration Tools" on page 30
"Residency of Configuration Data and Supporting Operations" on page 30
"Configure SST-PB3-CLX-HWL first, Experion second" on page 31
"Using the Molex Inc. (formerly Woodhead / SST) PROFIBUS Configuration Software Tool" on page 31
"Other Considerations" on page 32

3.5.1 Configuration Tools

Each of the affected hardware components in the Experion/PROFIBUS architecture require the use of specific configuration tools as listed in the table below:

Component	Configuration Tool
SST-PB3-CLX-HWL	SST PROFIBUS Configuration Tool
Profibus Network	SST PROFIBUS Configuration Tool
Slave Module/Station/Device (using GSD file)	SST PROFIBUS Configuration Tool
All Experion Blocks	Experion Control Builder

3.5.2 Residency of Configuration Data and Supporting Operations

Because multiple configuration tools are involved in the use of PROFIBUS with Experion, it is important to distinguish what configuration data (and related operations) are required for each of the configuration tools. Special attention must be paid to any operations that are applicable to multiple entries (shade boxes below) such that when a change is made, the corresponding change is made in each affected entity using the appropriate configuration tool.

Configuration Operation	Applicable to:			
	PFB Config. Tool	Experion Control Builder		
		PBIM Block	Device Module Blocks	Device Channel Blocks
Configuration of PROFIBUS network parameters, including network baud rate, highest station address, token rotation time, scan cycle times, other network timing parameters	X			
Station address for the SST-PB3-CLX-HWL	X			
Physical location of the SST-PB3-CLX-HWL on ControlNet		X		
Slave station/module configuration (using GSD files)	X			
Station address for slave devices	X	X	X	
Module number(s) for slave devices	X	X	X	

Configuration Operation	Applicable to:			
	PFB Config. Tool	Experion Control Builder		
		PBIM Block	Device Module Blocks	Device Channel Blocks
Module I/O message offsets into PBIM I/O assemblies	X	X		
Module I/O message sizes/formats	X		X	
Channel I/O message sizes/offsets				X
Data format/interpretation				X
Selection of particular control I/O parameters from a slave device for connection with other blocks				X

Note that the unique combination of station number and module number is used to identify a device throughout the architecture. Therefore, if a station/module number is changed, both the PBIM block and the Slave/device module block must be reconfigured to reflect this change.

3.5.3 Configure SST-PB3-CLX-HWL first, Experion second

The table in the preceding section indicates that several types of configuration operations require a change in both the SST-PB3-CLX-HWL configuration and the Experion Control Builder. It is generally recommended that such changes are made in the SST-PB3-CLX-HWL configuration first, and then in the supporting Experion blocks.

3.5.4 Using the Molex Inc. (formerly Woodhead / SST) PROFIBUS Configuration Software Tool

Instructions on the use of the Molex Inc (formerly Woohead / SST) PROFIBUS Configuration are provided with the SST-PB3-CLX-HWL. Several specific points are noteworthy when using the SST-PB3-CLX-HWL with the Experion system

The first step in creating a new configuration is creating an SST-PB3-CLX-HWL on the network (using a drag-and-drop technique in the configuration tool.) After creating the SST-PB3-CLX-HWL, and before creating/configuring any slave devices, it is recommended that the SST-PB3-CLX-HWL and the PROFIBUS network parameters be properly configured.

Network Baud Rate

The PROFIBUS baud rate is not configured as a property of the master, but as a property of the network itself. However, the selection cannot be made until a SST-PB3-CLX-HWL master has been created. Locate the SST-PB3-CLX-HWL of interest, and select the parent PROFIBUS object. Locate the PROFIBUS baud rate it and set it as desired. Note that all slave devices must be capable of operating at the rate selected. If a device is configured which cannot support the configured baud rate, a warning will be issued, and the baud rate changed to the closest supported value.

Highest Station

The highest configured station address is also a configurable property of the PROFIBUS network (not the SST-PB3-CLX-HWL). In order to minimize PROFIBUS cycle times, it is recommended that the highest station address is set equal to the highest station address that is physically present on the network. It is also generally recommended that where possible, unnecessary gaps in the configured station address range are avoided.

Input/Output Data Type

On the 'CLX Options' tab of the SST-PB3-CLX-HWL, the user can specify the 'Input Data Type' and 'Output Data Type', with choices of *Byte, Word*, and *Dword*, with the default value set to *Byte*. This is actually a 'data

alignment option for each module's data in the I/O messages. For example, choice of Word for the Input Data Type will ensure that within the 496 byte master input message assembly, each module's individual message starts at a word boundary, or even byte boundary. Note that changing these data alignment settings after slaves have been configured can change the entire configuration, specifically the slave data offsets. *Recommendation*: It is recommended that the default value of *byte* be always used for the data type of both input and output data. This ensures optimal packing of data and that 'padding' bytes have not been added between modules, which in some cases will affect the block that is interpreting the data.

Request Periodic Interval Rate

On the 'CLX Options' tab of the SST-PB3-CLX-HWL, the user should ensure that the check box is selected which indicates 'Request Periodic Interval Rate'. The default value of 5000 usec is not appropriate for use with the Experion system. This value should be set to the same value as the Update Rate (PUBRATE parameter) of the PBIM block. As a general rule, this periodic interval should be set to half of the base execution period of the CEE. Thus, the default value for the PUBRATE parameter of the PBIM block is 25,000 usec (half of a 50 ms cycle).

Watchdog time

After setting the 'Request Periodic Update Interval Rate' of the SST-PB3-CLX-HWL, certain parameters on the Parameters tab will be changed automatically, including the 'Watchdog' time. In order to ensure proper behavior in the event of a switchover/failover of a redundant Experion C200 controller, it is necessary that the watchdog time is set to a minimum value of 15 (x 10 ms). If the automatic setting is less than 15, select the Watchdog configuration manually, and enter a value of 15.

3.5.5 Other Considerations

In the event that a change needs to be made in the configuration of an SST-PB3-CLX-HWL (such as the addition or change in configuration of a station or module) that is currently communicating with an associated PBIM block in a C200 controller, prior to the load of the new configuration to the SST-PB3-CLX-HWL, the PBIM block must first be 1) put in the Clear Mode and 2) Inactivated. Note that these operations must be performed in the order specified, and are not 'bumpless' operations, and therefore should not be initiated in an on-process environment.

3.6 System Performance, Capacity and Topology Specifications

The information in this section is intended to provide a set of specifications that bound the system topology given the introduction of the PROFIBUS interface.

3.6.1 Experion/Profibus Interface Specifications

1	The SST-PB3-CLX-HWL module is not a redundancy compliant device and therefore cannot be introduced into a controller rack of a C200 redundant controller configuration. However, in non-redundant applications, the module can be deployed in either the C200 controller rack or a downlink I/O chassis.
2	The communication update interval between the SST-PB3-CLX-HWL and the PBIM block is configurable. The minimum interval is 5 ms, if in the local controller chassis; and 12.5 ms, if in a remote chassis. The maximum update interval is 50 ms.
3	The maximum number of SST-PB3-CLX-HWL modules that can be supported are: 10 by a C200 hosting a 50 ms CEE, 2 by a C200 hosting a 5 ms CEE, 8 by a C300, or 4 by a Fault Tolerant Ethernet Bridge module (FTEB).
4	The maximum number of SST-PB3-CLX-HWL modules that can be supported per I/O subnetwork or downlink CNI are: 4 at a 25ms rate or 2 at a 12.5 ms rate.
5	The SST-PB3-CLX-HWL imposes the following constraints, which restricts the number of PROFIBUS slave stations/devices supported: • The sum of all input message sizes from all input devices cannot exceed 496 bytes. • The sum of all output message sizes from all output devices cannot exceed 492 bytes.
6	The PBIM block supports communication to any/all PROFIBUS stations within the valid station address range of 0-125.
7	The PBIM block supports a maximum of 100 unique modules, identified by a unique station/module number combination.
8	Regarding the existing system limit of 64 IO Units per CPM, 24 IO Units per downlink CNI, and 16 IO Units per FTEB: • The SST-PB3-CLX-HWL module is the equivalent of 2 IO Units for C200 CPM. • The SST-PB3-CLX-HWL module is the equivalent of 6 IO Units for downlink CNI. • The SST-PB3-CLX-HWL module is the equivalent of 8 IO Units for C300 CPM. • The SST-PB3-CLX-HWL module is the equivalent of 4 IO Units for FTEB.
9	All PROFIBUS DP baud rates are supported, up to 12 Mbps.

4 PBIM block

The PBIM function block (template name PBIM-SST) is an I/O module function block that represents the SST-PB3-CLX-HWL card. Because the SST-PB3-CLX-HWL card itself can host multiple I/O modules on PROFIBUS, the PBIM block supports multiple PROFIBUS I/O module blocks, which reference the PBIM block for I/O data. The PBIM block itself does not interpret any I/O data; it merely serves I/O data to each of its associated module blocks.

Related topics

“PBIM block configuration” on page 36
“PBIM block operation” on page 46

4.1 PBIM block configuration

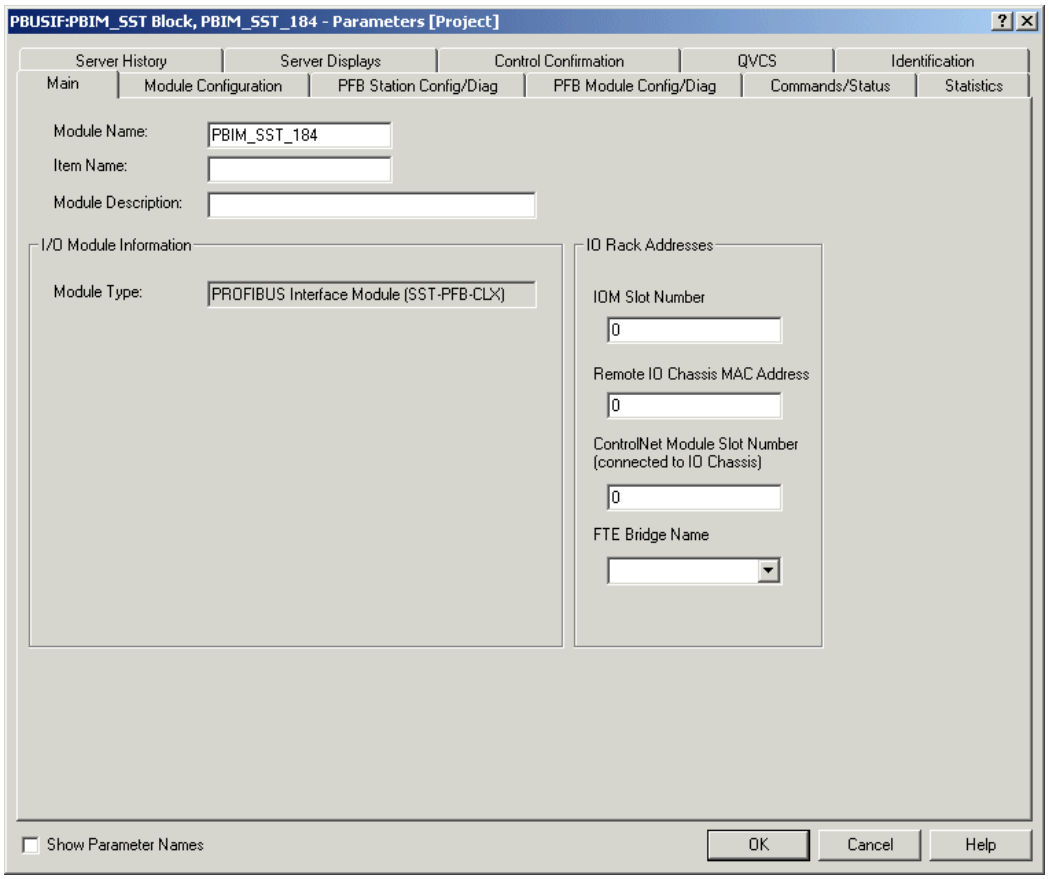
As accessed using the Experion Control Builder, configuration of the PBIM block is accomplished by entering the appropriate configuration data on the following tabs:

4.1.1 Main Tab

The main tab of the PBIM is similar to that of other I/O module function blocks. Be sure to enter the appropriate entries to specify the physical location of the SST-PB3-CLX-HWL card on the I/O network using the following parameters in the box labeled 'I/O Rack Address':

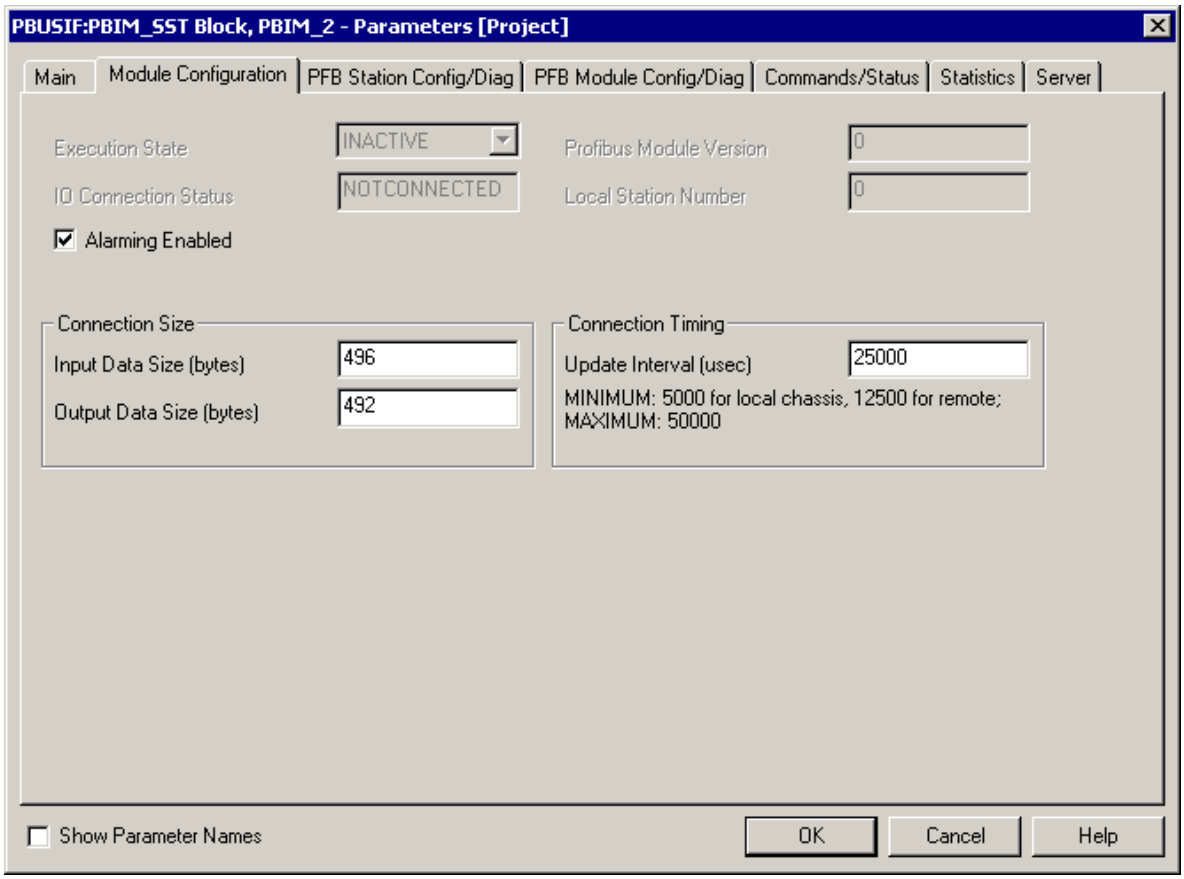
- IOMSLOT - The physical slot number of the module in the chassis
- ULCNBMAC - MAC Address of the uplink CNI - in the chassis which contains the module (0 if co-located with a non-redundant CPM)
- DLCNBSLOT - Slot number of the downlink CNI - in the controller chassis (0 if co-located with a non-redundant CPM)
- FTEBBLOCK - The FTEB module/block to which the PROFIBUS Interface Module is attached/assigned. (Only appears if module is used in an FTE network.)

The following is an example of the Main tab of the PBIM block:



4.1.2 Module Configuration Tab

The following is an example of the Module Configuration tab of the PBIM block as seen from the Project tab of CB:



4.1.3 Input/Output Data connection Sizes

Although the PBIM card is capable of providing up to 496 bytes of input and 492 bytes of output data to/from the C200 controller, in practice, dependent upon the PROFIBUS configuration, the entire input and output data may not be used. As a performance optimization, the amount of data that is transported across the Input and Output data connections can be reduced from the maximum values. The INCONNSIZE and OUTCONNSIZE parameters are described below.

Parameter Name:	**INCONNSIZE**
Specific to Block:	PBIM
Description:	Input Connection Size in bytes
Data Type:>	INT32
Range:	1 - 496
Default:	496
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	OUTCONNSIZE

Remarks:	If less than the maximum amount of input data is being utilized based on the PROFIBUS configuration, reducing this parameter from the default value will decrease the communications loading on the I/O subnetwork. Requires a minimum value of one (1), since a value of zero is not allowed for Control Builder Configuration.

Parameter Name:	**OUTCONNSIZE**
Specific to Block:	PBIM
Description:	Output Connection Size in bytes
Data Type:	INT32
Range:	0 - 492
Default:	492
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	INCONNSIZE
Remarks:	If less than the maximum amount of output data is being utilized based on the PROFIBUS configuration, reducing this parameter from the default value will decrease the communications loading on the I/O subnetwork.

4.1.4 Data connection update rate

The rate at which data is exchanged between the SST-PB3-CLX-HWL card and the C200 controller (PBIM block) can be configured through the PUBRATE parameter. The units are in microseconds and represent the time period of the data transfer. Although I/O data publication is periodic in nature, the input and output data exchange between the PBIM and the C200 is not synchronized with the C200 block execution cycle. Therefore, it is generally recommended that the data exchange rates are at least double the fastest execution period that is being used in control of the I/O data for the respective PBIM.

When the SST-PB3-CLX-HWL is located in a remote I/O chassis, the allowable ranges for the data transfer period vary from 12,500 - 50,000 usec, or 12.5 to 50 ms. If the SST-PB3-CLX-HWL is located in the local controller chassis (non-redundant configuration), the lower limit is 5000 usec, or 5 msec. If an entry is made which is beyond these limits, the load will proceed and the value will be clamped to the limit value and a warning returned on the load operation.

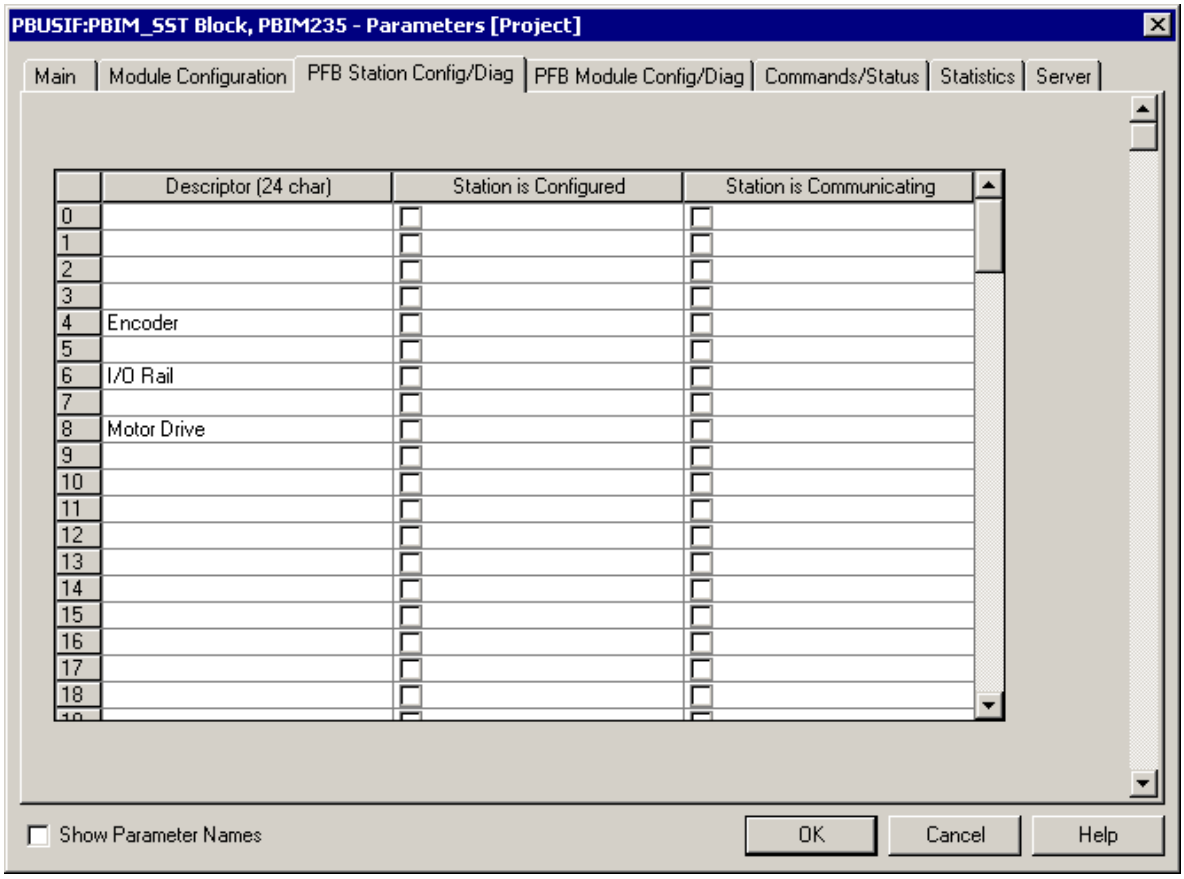
The setting for the 'Request Periodic Interval Rate' of the SST-PB3-CLX-HWL should set to the same value as the PUBRATE parameter of the PBIM block. Refer to the *Request Periodic Interval Rate* Section for more information.

Parameter Name:	**PUBRATE**
Specific to Block:	PBIM
Description:	I/O Update Rate - in microseconds (usec)
Data Type:	INT32
Range:	Remote chassis: 12,500 - 50,000 usec (12.5 - 50 ms) Local chassis: 5,000 - 50,000 usec (5 - 50 ms)
Default:	25,000 usec (25 ms)
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE

Related Parameters:	
Remarks:	Specifies the period at which input/output data is exchanged between the SST-PB3-CLX-HWL and the CEE. It is generally recommended that the data exchange rates is at least double the fastest execution period which is being used to host control using input data from the respective PBIM. For a 50ms execution period, the default value of 25,000 usec (25 msec) is appropriate. Increasing the value will increase control latency while lessening the communications loading on the I/O subnetwork. Raising the value will decrease control latency while increasing the communications loading on the I/O subnetwork. The setting for the 'Request Periodic Interval Rate' of the SST-PB3-CLX-HWL should set to the same value as the PUBRATE parameter.

4.1.5 PFB Station Config/Diag Tab

This tab contains a table that is used for both configuration and diagnostic information for PROFIBUS station devices. The only configuration parameter is a 24-character text descriptor. The table is indexed by station number, which can vary from 0 to 125. The data entered in this table should be in agreement with the actual configuration of the PROFIBUS network, using the SST PROFIBUS Configuration Software. A sample of the tab is pictured below:



The parameters on this form are listed below:

4.1.6 Configuration parameters related to PFB Station Config/Diag tab

Parameter Name:	**STATIONDESC[0..125]**
Specific to Block:	PBIM

Description:	Station Descriptor	
Data Type:	24 Character Text String	
Range:	N/A	
Default:		
Config Load:	Yes	
Access Lock:	Application Developer	
Residence:	SR	
Related Parameters:		
Remarks:		

4.1.7 Station Diagnostic Parameters

The two columns of discrete parameters (PFBSTNCFG and PFBSTNACT) provide information on each station that is generated by the SST-PB3-CLX-HWL, specifically if each station is configured and online/ communicating. See the parameter definitions below for additional details.

Parameter Name:	**PFBSTNCFG[0..125]**	
Specific to Block:	PBIM	
Description:	Configured Station List	
Data Type:	BOOLEAN	
Range:	Off (0)	Slave station not configured.
	On (1)	Slave station configured.
Default:	Off	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	N/A	
Remarks:	The PBIM maintains a 'configured slave station' list across the entire Profibus address space. When asserted, indicates that the PBIM was configured with a slave station to reside at the associated Profibus address (i.e. the Boolean array index).	

Parameter Name:	**PFBSTNACT[0..125]**	
Specific to Block:	PBIM	
Description:	Active Station List	
Data Type:	BOOLEAN	
Range:	Off (0)	Slave station is absent and/or returning errors.
	On (1)	Slave station is present and returning no errors.
Default:	Off	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	N/A	

Remarks:	The PBI Module maintains an 'active slave station' list across the entire Profibus address space. When asserted, the slave station that resides at the associated Profibus address (i.e. the Boolean array index) is present on the Profibus network, is communicating with the master device, and is not returning any errors.

4.1.8 PFB Module Config/Diag Tab

This tab contains a table that captures the details of the PROFIBUS configuration, specifically the identity (station and module number) of each module and the physical location of the input and output data message for each module within the PBIM's input and output data structures.

Acquiring the PROFIBUS Configuration Data

The information entered in this tab/table should be in agreement with the actual configuration of the PROFIBUS network, as configured by the Molex Inc. (formerly Woodhead / SST) PROFIBUS Configuration Software. This information is available in a summarized format by invoking the following command sequence in the Molex Inc. (formerly /Woodhead / SST) PROFIBUS Configuration SW Tool:

File >> Print Preview

For each slave station, the four necessary parameters are listed:

- Station number
- Module number
- Input data offset
- Output data offset

Configuration Mismatch between PBIM and Device/Module Blocks

After load of the PBIM block and its associated device/module blocks, the device/module block attempts to 'register' for the use of the correct entry in this table. Registration will not succeed if any of the following illogical conditions apply:

- The device block requires input data, but the PBIM table has a null value for input offset.
- The device block requires output data, but the PBIM table has a null value for output offset.
- The device block requires no input data, but the PBIM table has a non-null value for input offset.
- The device block requires no output data, but the PBIM table has a non-null value for output offset.

No Overlap Configuration Checks in PBIM Block

When the actual PROFIBUS configuration is entered using the Molex Inc. (formerly Woodhead / SST) PROFIBUS Configuration Tool, the offset values are assigned in a manner that ensures that there is no ‘overlapping' of individual I/O messages within the I/O assembly data structures. Because the PROFIBUS configuration tool enforces this functionality, and because this table in the PBIM block represents a copy of that configuration, no consistency or overlap checking is performed by the PBIM block.

One Entry Per Device/Module Block

As a general rule, one entry should be made in the table for each required device/module block. This rule allows the possibility for the configuration in the Molex Inc. (formerly Woodhead / SST) Configuration SW and the PBIM block to be different under the following condition:

If a particular slave device is configured on PROFIBUS such that it has multiple (physical or virtual) modules, and *if* that device and its modules will be represented by a *single* generic or custom device/module block (rather than a separate block for each module), only a single entry needs to be made in the module configuration table. Having entries for each is not harmful, but is also not necessary, given that no device/module block will register

for those entries. Here is an example of this situation for a motor control device that employs the use of multiple virtual modules:

- *PROFIBUS Configuration*: One station, *two* modules: Station 8, with Module 0 providing 8 bytes of I/O data and Module 1 providing 12 bytes of I/O data.
- *Experion PBIM block and Module Device Block Configuration*: One station, *one* module: Station 8 and Module 0

Consistency of Configuration

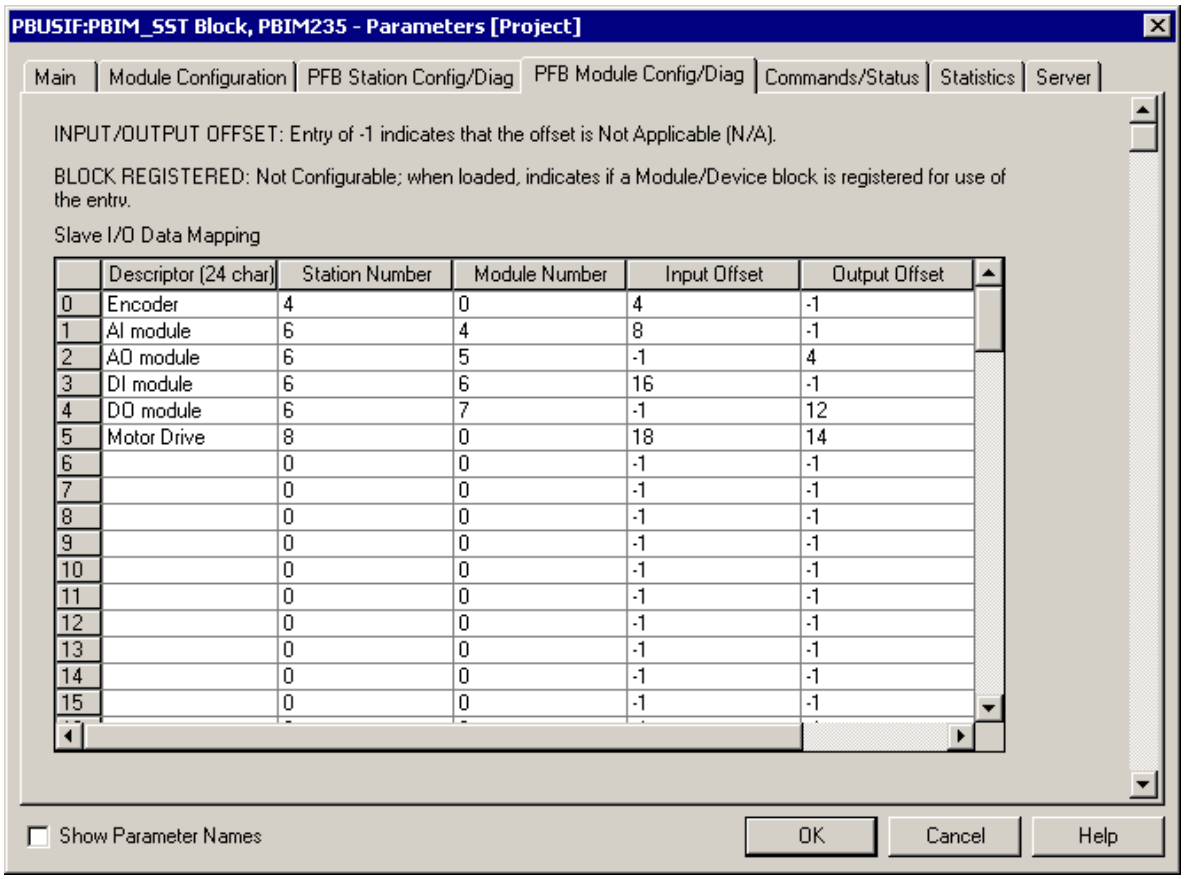
Given the exceptions listed above, in general, the information configured in this table should be in agreement with the actual configuration of the PROFIBUS network, as configured by the Molex Inc. (formerly Woodhead / SST) PROFIBUS Configuration Software. Note however, that there is no enforcement of this postulate. For example, if a particular device is physically configured on PROFIBUS (using the Molex Inc. (formerly Woodhead / SST) Configuration Tool) as Station X / Module Y, with input data at offset 42 and output data at offset 30, it would be possible to configure the PBIM block, and an associated I/O module block with a different station and module number, provided that the correct values were entered for the offsets. Although illogical, such a configuration would allow successful interpretation of the I/O data and control of the device.

WARNING

It is possible for the user to make configuration errors in the configuration of the PBIM module configuration table and/or the associated I/O Module and channel blocks, which reference the PBIM block that could cause unpredictable behavior of output devices. For example, it is possible to configure two blocks that are writing the same byte(s) in the PBIM output assembly, without the generation of a configuration error. Therefore, it is recommended that all address, offset, and data size parameters be checked for accuracy prior to block activation.

4.1.9 Sample Module Configuration Table

The table is of the following format, with some sample entries listed:



	Descriptor (24 char)	Station Number	Module Number	Input Offset	Output Offset
0	Encoder	4	0	4	-1
1	AI module	6	4	8	-1
2	AO module	6	5	-1	4
3	DI module	6	6	16	-1
4	DO module	6	7	-1	12
5	Motor Drive	8	0	18	14
6		0	0	-1	-1
7		0	0	-1	-1
8		0	0	-1	-1
9		0	0	-1	-1
10		0	0	-1	-1
11		0	0	-1	-1
12		0	0	-1	-1
13		0	0	-1	-1
14		0	0	-1	-1
15		0	0	-1	-1

Notes on the example configuration table:

- The table allows up to 100 entries, representing 100 unique module devices. Unlike the station configuration tab, where the table index represented the station number, the module configuration tab index has no relation to the module number (which is entered on the third column).
- Each entry in the table is referenced by a single I/O device/module block.
- The valid range for input address offset is 4 to 499 (bytes).
- The valid range for output address offset is 4 to 495 (bytes).
- A seemingly illogical entry of -1 for input or output offset indicates that the module does not have input or output data, respectively. Such is represented as 'N/A' in the printed report generated by the Molex Inc. (formerly Woodhead / SST) PROFIBUS configuration SW.
- The sizes of the I/O data messages for each device are not specified in this table, only the offsets. Note that the Molex Inc. (formerly Woodhead / SST) PROFIBUS Configuration Software ensures that there are no 'overlapping' messages for slave devices.
- The order of the entries in the PBIM configuration form is variable.
- Gaps (skipped entries) in the table are permitted.
- Duplicate entries with the same station/module number entries should be avoided. However, if multiple identical station/module number entries are entered, no error will be indicated to the user. In the event of such a configuration, the associated IOM block would always register for use of the first entry that matches its station/module configuration.
- The 'Module Block Registered' column is not configurable. (This column is at the right side of the table, and cannot be seen in the example above - requires horizontal scrolling.) However, after load of the PBIM block and associated device/module blocks, this flag indicates if a device/module block has registered for use of the respective entry. This can be helpful in debugging configuration errors.

4.1.10 Module Configuration Parameters

The parameters that appear in *the Slave I/O Data Mapping* table of the *PFB Module Config/Diag* tab are described below:

Parameter Name:	**MODDESC[0..99]**
Specific to Block:	PBIM
Description:	Module Descriptor
Data Type:	24 Character Text String
Range:	N/A
Default:	
Config Load:	Yes
Access Lock:	Application Developer
Residence:	SR
Related Parameters:	
Remarks:	

Parameter Name:	**STATION[0..99]**
Specific to Block:	PBIM
Description:	Station Number
Data Type:	INT32
Range:	0..125
Default:	0
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	MODULE
Remarks:	The Profibus Station Number and Module number are the two parameters necessary to uniquely identify a slave device/module.

Parameter Name:	**MODULE[0..99]**
Specific to Block:	PBIM
Description:	Module Number
Data Type:	INT32
Range:	0..100
Default:	0
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	STATION
Remarks:	The Profibus Station Number and Module number are the two parameters necessary to uniquely identify a slave device/module. The Module Number is unique only within the domain of the associated Profibus Station.

Parameter Name:	**INOFFSET[0..99]**
Specific to Block:	PBIM
Description:	Module Input Data Offset
Data Type:	INT32
Range:	-1, 4-499
Default:	-1
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	OUTOFFSET
Remarks:	Specifies the location of the input data message for the module of interest as a byte offset within the PBIM's input data assembly. An entry of -1 indicates that the module does not supply input data to the CEE.

Parameter Name:	**OUTOFFSET[0..99]**	
Specific to Block:	PBIM	
Description:	Module Output Data Offset	
Data Type:	INT32	
Range:	-1, 4-495	
Default:	-1	
Config Load:	Yes	
Access Lock:	Application Developer	
Residence:	CEE	
Related Parameters:	INOFFSET	
Remarks:	Specifies the location of the output data message for the module of interest as a byte offset within the PBIM's output data assembly. An entry of -1 indicates that the module does not supply output data to the CEE.	

Parameter Name:	**REGISTERED[0..99]**	
Specific to Block:	PBIM	
Description:	Registered	
Data Type:	BOOLEAN	
Range:	Off (0)	No device/module block has registered with the PBIM block for use of this entry and associated I/O data.
	On (1)	A device/module block has registered with the PBIM block for use of this entry and associated I/O data.
Default:	Off	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	STATION, MODULE, INOFFSET, OUTOFFSET	
Remarks:	Indicates whether or not a Profibus Device FB has successfully registered against a given entry within the PBIM Assembly Map Table.	

4.2 PBIM block operation

The following section describes the operation of the PBIM block.

Attention
During the controller load/reload or checkpoint restore operation, the error message 'ASAPATH: Read Only Parameter' appears. You can ignore this message and proceed with the operation.

Related topics

“Connection Status” on page 46
“Execution State Behavior” on page 46
“Clear Mode / Run Mode” on page 47
“PBIM blocks operational parameters” on page 47
“PBIM blocks diagnostic alarms” on page 47
“Command Parameters” on page 47
“Configuration Diagnostic Parameters” on page 48
“Status Parameters” on page 49
“Statistical Parameters” on page 54
“Debug Parameters” on page 61

4.2.1 Connection Status

After load of the PBIM block, the Connection Status (IOCONNSTATUS parameter) is used to verify that a communications connection has been successfully established with the SST-PB3-CLX-HWL. Use of this parameter is helpful in debugging configuration errors. The I/O connection status should assume the CONNECTED state if the card is present at the specified location. If the physical location of the PBIM was improperly configured or if the PBIM card is physically not present, the connection status will be IOMNOTFOUND.

4.2.2 Execution State Behavior

The EXECSTATE parameter (Execution State) is common to all I/O module blocks, and is a property of the PBIM block itself, and does not reflect the behavior of the SST-PB3-CLX-HWL card. The specific behavior is listed below:

- ACTIVE: I/O data is regularly transferred between the PBIM block and the SST-PB3-CLX-HWL card.
- INACTIVE: I/O data is not transferred between the PBIM block and the SST-PB3-CLX-HWL card. Fail-Safe data is returned to client input blocks.

Because PBIM inactivation causes a break in the communication between I/O device/module blocks and the PROFIBUS network, those blocks will generate alarms when the PBIM block is inactivated.

In order to change the configuration of the PROFIBUS network, such as adding, removing or reconfiguring one or more slave devices, the PBIM block must be in the INACTIVE state. Because I/O communication is halted when the PBIM is inactive, making any changes to the PROFIBUS configuration is not recommended when on-process.

When the module is active, the vaule of the Local Station number is read from the active module. If you change the module status from ACTIVE to INACTIVE, the Local Station number is displayed as '255' on the faceplate of the module's detail display. This indicates that the SST-PFB-CLX card is INACTIVE state.

4.2.3 Clear Mode / Run Mode

The PBIM block also contains a separate set of command and status parameters that are used for toggling between 'run mode' and 'clear mode'. Although accessed through the PBIM block, this mode is a property of the SST-PB3-CLX-HWL card rather than the block itself. Clear/run mode affects only the output data (data from the CEE to all output devices), and does not affect input data (from input devices to the CEE).

- Run Mode- In this state, the output data that has been provided by the PBIM block to the SST-PB3-CLX-HWL is actively communicated to the slave devices via PROFIBUS.
- Clear Mode - In this state, the output data which has been provided by the PBIM block, is not sent out on PROFIBUS by the SST-PB3-CLX-HWL; instead all output messages to all output slave devices are set to an array of zero's.

By default, the PROFIBUS Run Mode is enabled when the PBIM block is loaded.

4.2.4 PBIM blocks operational parameters

In general, the PBIM block supports all of the parameters typically associated with I/O Module Function Blocks in the Experion system.

4.2.5 PBIM blocks diagnostic alarms

When active, all I/O Module blocks report diagnostic alarms in the event that communication has been disrupted between the block and the end device. Specifically, the following diagnostic alarms are presented for the PBIM block, provided that the PBIM block is currently in the active state:

- A system diagnostic alarm indicating a 'Communication Error' is generated when the controller to SST-PB3-CLX-HWL communication connection is broken. This is a common notification for all I/O Module Function Blocks in the Experion system. Several causes for generation of this alarm are {1} power is disrupted to the remote chassis in which the SST-PB3-CLX-HWL resides, {2} the SST-PB3-CLX-HWL is removed under power, or {3} the controller to SST-PB3-CLX-HWL ControlNet communication path (assuming remote I/O) is broken. Note that there is a one-to-one correlation between this diagnostic alarm and the IOCONNSTATUS parameter (i.e. alarm generated when IOCONNSTATUS is not set to CONNECTED).
- A system diagnostic alarm indicating a 'PROFIBUS Comm Error' is generated when the SST-PB3-CLX-HWL PROFIBUS communication connection is broken. For example, this alarm is generated when the PROFIBUS cable is removed either from the SST-PB3-CLX-HWL or from the last Station such that the SST-PB3-CLX-HWL no longer has view to any Stations on PROFIBUS.

4.2.6 Command Parameters

Parameter Name:	**PFBCLRMODE**	
Specific to Block:	PBIM	
Description:	Enable Clear Mode	
Data Type:	BOOLEAN	
Range:	Off (0)	Transition the PROFIBUS network out of Clear mode and into Run mode, where output data is transferred from the CEE, through the SST-PB3-CLX-HWL and to the slave device.
	On (1)	Transition the PROFIBUS network out of Run mode and into clear mode, where the output assembly is set to an array of zero's. Transition the PBI IOM online with the Profibus network (if not already the case) and into the Run mode.

Default:	Off
Config Load:	No
Access Lock:	Operator
Residence:	CEE
Related Parameters:	PFBMASCFG02, PFBMASCFGRAW
Remarks:	When asserted, the PBIM block's internal state machine issues the appropriate commands to automatically {1} transition the PBIM online with the Profibus network and {2} transition the PBIM into the Run mode.

Parameter Name:	**PFBCLRERR**	
Specific to Block:	PBIM	
Description:	Clear Counters	
Data Type:	BOOLEAN	
Range:	Off (0)	
	On (1)	
Default:	Off	
Config Load:	No	
Access Lock:	Operator	
Residence:	CEE	
Related Parameters:	N/A	
Remarks:	When asserted, the PBIM block's internal state machine issues the appropriate command to request the SST-PB3-CLX-HWL module to clear its counters to 0. This attribute automatically transitions to the negated state to indicate that the counters have been cleared. Note that this parameter is only acted upon when the PBIM is online with the Profibus network.	

4.2.7 Configuration Diagnostic Parameters

Parameter Name:	**PFBMODVER**	
Specific to Block:	PBIM	
Description:	Profibus Module Version	
Data Type:	INT32	
Range:	N/A	
Default:	0	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	N/A	
Remarks:	(e.g. 0x0102 = 1.02)	

Parameter Name:	**PFBSTATION**
Specific to Block:	PBIM
Description:	Local Station Number
Data Type:	INT32

Range:	N/A
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	N/A
Remarks:	Profibus station number assigned to the PBIM.

4.2.8 Status Parameters

Parameter Name:	**PFBMODSTSRAW**
Specific to Block:	PBIM
Description:	Raw Interface Module Status
Data Type:	INT32
Range:	N/A
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	PFBMODSTSENM
Remarks:	The PBIM initializes the module status register. Numerical values are interpreted as follows:
0x0000	The PBIM has no errors.
0x0002	The configured baud rate is not valid.
0x0003	The configured station address is not valid.
0x0004	The configured Highest Station Address is not valid.
0x0080	The PBIM has encountered a fatal internal error.
0x0081	The PBIM has run out of internal resources (i.e. ran out of application blocks).

0x0083	The PBIM has run out of internal resources (i.e. ran out of local RAM).
0x0084	The PBIM has run out of internal resources (i.e. ran out of shared RAM).

Parameter Name:	**PFBMODSTSENM**	
Specific to Block:	PBIM	
Description:	Interface Module Status	
Data Type:	ENUM	
Range:	Unknown (0)	Parameter not yet initialized by the PBI IOM.
	NoError (1)	The PBIM has no errors.
Bad_Baud (2)	The configured baud rate is not valid.	
Bad_Stn_Addr (3)	The configured station address is not valid.	
Bad_Hi_Addr (4)	The configured Highest Station Address is not valid.	
InternalErr (5)	The PBIM has encountered a fatal internal error.	
ResourceErr (6)	The PBIM has run out of internal resources (refer to PFBMODSTSRAW).	
Undefined (7)	None of the above (refer to PFBMODSTSRAW).	
Default:	Unknown	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	PFBMODSTSRAW	
Remarks:	The PBIM initializes the module status register.	

Parameter Name:	**PFBMASSTSRAW**	
Specific to Block:	PBIM	
Description:	Raw DP Master Global Status	
Data Type:	INT32	
Range:	N/A	
Default:	0	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	

Related Parameters:	PFBMASSTSENM	
Remarks:	Global Status for all master blocks. The PBIM initializes the value of this parameter and numerical values are interpreted as follows:	
	0	At least on DP slave is not configured or not present on the bus.
	1	All the configured DP slave devices are on-line and operating.

Parameter Name:	**PFBMASSTSENM**	
Specific to Block:	PBIM	
Description:	DP Master Global Status	
Data Type:	ENUM	
Range:	Unknown (0)	Parameter not yet initialized by the PBI IOM.
	All_Not_Ok (1)	At least on DP slave is not configured or not present on the bus.
	All_Ok (2)	All the configured DP slave devices are on-line and operating.
	Undefined (3)	None of the above (refer to PFBMASSTSRAW).
Default:	Unknown	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	PFBMASSTSRAW	
Remarks:	Global Status for all master blocks; this enumeration is derived from PFBMASSTSRAW.	

Parameter Name:	**PFBMASCFGRAW**	
Specific to Block:	PBIM	
Description:	Raw DP Master Configuration	
Data Type:	INT32	
Range:	N/A	
Default:	0	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	N/A	
Remarks:	DP Master options and configuration. The PBI IOM initializes the value of this parameter and numerical values are interpreted as follows:	
	0x01	Sync Scan mode is enabled.
	0x02	The DP Bus state is Run (Operate).
	0x04	The DP slave data addresses have been assigned to controller memory.
	0x08	DP Master mode is enabled.
	0x80	The DP slave data to controller memory addresses are valid.

Parameter Name:	**PFBMASCFG01**
Specific to Block:	PBIM
Description:	Sync Scan mode enabled
Data Type:	BOOLEAN

Range:	Off (0)	Sync Scan mode disabled.
	On (1)	Sync Scan mode enabled.
Default:	Off	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	PFBMASCFGRAW	
Remarks:	This Boolean parameter derives from bit 0 of PFBMASCFGRAW.	

Parameter Name:	**PFBMASCFG02**	
Specific to Block:	PBIM	
Description:	DP Bus in Run State	
Data Type:	BOOLEAN	
Range:	Off (0)	DP Bus in Clear Mode.
	On (1)	DP Bus in Run Mode.
Default:	Off	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	PFBMASCFGRAW	
Remarks:	This Boolean parameter derives from bit 1 of PFBMASCFGRAW.	

Parameter Name:	**PFBMASCFG04**	
Specific to Block:	PBIM	
Description:	Slave offsets assigned	
Data Type:	BOOLEAN	
Range:	Off (0)	Slave offsets not assigned.
	On (1)	Slave offsets assigned.
Default:	Off	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	PFBMASCFGRAW	
Remarks:	This Boolean parameter derives from bit 3 of PFBMASCFGRAW.	

Parameter Name:	**PFBMASCFG08**	
Specific to Block:	PBIM	
Description:	DP Master mode enabled	
Data Type:	BOOLEAN	
Range:	Off (0)	DP Master mode disabled.
	On (1)	DP Master mode enabled.
Default:	Off	

Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	PFBMASCFGRAW
Remarks:	This Boolean parameter derives from bit 4 of PFBMASCFGRAW.

Parameter Name:	**PFBMASCFG80**	
Specific to Block:	PBIM	
Description:	Slave offsets valid	
Data Type:	BOOLEAN	
Range:	Off (0)	Slave offsets are not valid.
	On (1)	Slave offsets are valid.
Default:	Off	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	PFBMASCFGRAW	
Remarks:	This Boolean parameter derives from bit 5 of PFBMASCFGRAW.	

Parameter Name:	**PFBDIAGCONF**	
Specific to Block:	PBIM	
Description:	Total Confirmations	
Data Type:	INT32	
Range:	N/A	
Default:	0	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	N/A	
Remarks:	Counts total confirmations (good replies to messages that this station has generated). This is the total for DP master, FDL messages, and FMS messages. Note that this counter rolls over to zero when maximum value reached.	

Parameter Name:	**PFBDIAGIND**	
Specific to Block:	PBIM	
Description:	Total Indications	
Data Type:	INT32	
Range:	N/A	
Default:	0	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	N/A	

Remarks:	Counts total indications (unsolicited messages to this station). This is the total for DP master, FDL messages and FMS messages. Note that this counter rolls over to zero when maximum value reached.

Parameter Name:	**PFBERRNOTOK**	
Specific to Block:	PBIM	
Description:	Total Not Ok Conf/Ind	
Data Type:	INT32	
Range:	N/A	
Default:	0	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	N/A	
Remarks:	Counts the Total Not OK confirmations and indications (total bad replies and bad unsolicited messages (indications)). This is the total for DP master, FDL messages and FMS messages. Note that this counter holds its maximum value.	

4.2.9 Statistical Parameters

Parameter Name:	**PFBFMSSTS**	
Specific to Block:	PBIM	
Description:	FMS Global Status	
Data Type:	INT32	
Range:	N/A	
Default:	0	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	N/A	
Remarks:	Global Status for all FMS blocks.	

Parameter Name:	**PFBERRLAN**	
Specific to Block:	PBIM	
Description:	LAN Errors	
Data Type:	INT32	
Range:	N/A	
Default:	0	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	N/A	
Remarks:	LAN encountered errors and went into off-line state. This counter holds its maximum value.	

Parameter Name:	**PFBTOKHLD**
Specific to Block:	PBIM
Description:	Inst. Token Hold Time
Data Type:	INT32
Range:	N/A
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	N/A
Remarks:	Instantaneous token hold time in Tbits. This time is the time available to send messages when the PBI IOM gets the token.

Parameter Name:	**PFBMINTOKHLD**
Specific to Block:	PBIM
Description:	Min Token Hold Time
Data Type:	INT32
Range:	N/A
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	N/A
Remarks:	Minimum Actual token hold time in Tbits. This time is the minimum value of PFBTOKHLD. If this number is 0, it may be necessary to increase the target token rotation time.

Parameter Name:	**PFBMASUPDATE**
Specific to Block:	PBIM
Description:	I/O Update Cycles Completed
Data Type:	INT32
Range:	N/A
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	N/A
Remarks:	Number of master I/O update cycles completed (i.e. the number of complete I/O scans completed by the master). Counter rolls over to zero when maximum value reached.

Parameter Name:	**PFBERRRECNFG**
Specific to Block:	PBIM
Description:	Master offline for Reconfiguration

Data Type:	INT32
Range:	N/A
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	N/A
Remarks:	Number of times a DP slave went offline and had to be reconfigured; the master was actively updating a slave and got a faulty message. The card increments this counter after it has retried the message the specified number of times. This counter holds its maximum value.

Parameter Name:	**PFBERRMASERR**
Specific to Block:	PBIM
Description:	Master->Slave Communication Errors
Data Type:	INT32
Range:	N/A
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	N/A
Remarks:	Number of DP master to DP slave communication errors. It increments any time a message failed because of retries exceeded, etc. This counter holds its maximum value.

Parameter Name:	**PFBINSSCAN**
Specific to Block:	PBIM
Description:	Inst. Master Scan Time
Data Type:	INT32
Range:	N/A
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	N/A
Remarks:	Instantaneous master scan time in microseconds. This is the time to scan all the slaves assigned to this master.

Parameter Name:	**PFBMAXSCAN**
Specific to Block:	PBIM
Description:	Max Master Scan Time
Data Type:	INT32
Range:	N/A
Default:	0

Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	N/A
Remarks:	Maximum master scan time in microseconds. More specifically, contains the maximum value that PFBINSSCAN has reached since it was last cleared.

Parameter Name:	**PFBERRFIFO**
Specific to Block:	PBIM
Description:	FIFO Overflow Errors
Data Type:	INT32
Range:	N/A
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	N/A
Remarks:	Counts FIFO overflow errors. When this (1 byte) counter reaches 255, it holds this maximum value until cleared.

Parameter Name:	**PFBERRREQLEN**
Specific to Block:	PBIM
Description:	Invalid Req. Length Errors
Data Type:	INT32
Range:	N/A
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	N/A
Remarks:	Counts invalid request length errors. When this (1 byte) counter reaches 255, it holds this maximum value until cleared.

Parameter Name:	**PFBERRDBLTOK**
Specific to Block:	PBIM
Description:	Double Token Errors
Data Type:	INT32
Range:	N/A
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE

Related Parameters:	N/A
Remarks:	Counts double token errors. These errors may occur when more than one node thinks it has the token or they may occur due to wiring errors, duplicate nodes, etc. When this (1 byte) counter reaches 255, it holds this maximum value until cleared.

Parameter Name:	**PFBERRRXOVR**
Specific to Block:	PBIM
Description:	Receive Overrun Errors
Data Type:	INT32
Range:	N/A
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	N/A
Remarks:	Counts receive overrun errors. When this (1 byte) counter reaches 255, it holds this maximum value until cleared.

Parameter Name:	**PFBERRSYNI**
Specific to Block:	PBIM
Description:	Syni Errors
Data Type:	INT32
Range:	N/A
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	N/A
Remarks:	Indicates general network errors. These errors occur when there are problems on the network (e.g. due to bad hardware or faulty wiring) but the problems are not severe enough to cause a network timeout error. When this (1 byte) counter reaches 255, it holds this maximum value until cleared.

Parameter Name:	**PFBERRRESP**
Specific to Block:	PBIM
Description:	Response Errors
Data Type:	INT32
Range:	N/A
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	N/A

Remarks:	Counts response errors (that is, when a message failed or there was no response from the destination. This error may be due to bad hardware or faulty wiring. When this (1 byte) counter reaches 255, it holds this maximum value until cleared.

Parameter Name:	**PFBERRHSA**
Specific to Block:	PBIM
Description:	Station Higher Than HSA
Data Type:	INT32
Range:	N/A
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	N/A
Remarks:	Increments when a station with a station number higher than the high station address set on the PBI IOM was heard. When this (1 byte) counter reaches 255, it holds this maximum value until cleared.

Parameter Name:	**PFBERRNETTMO**
Specific to Block:	PBIM
Description:	Network Timeout Errors
Data Type:	INT32
Range:	N/A
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	N/A
Remarks:	Counts network timeout errors. These errors occur when the network is dead. When this (1 byte) counter reaches 255, it holds this maximum value until cleared.

Parameter Name:	**PFBERRPASTOK**
Specific to Block:	PBIM
Description:	Unable to Pass Token
Data Type:	INT32
Range:	N/A
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	N/A

Remarks:	Increments when the PBI IOM is unable to pass the token (on Profibus). This is usually caused by bad wiring (usually shorted) or other hardware problems. When this (1 byte) counter reaches 255, it holds this maximum value until cleared.

Parameter Name:	**PFBERRSTN**
Specific to Block:	PBIM
Description:	Duplicate Station Detected
Data Type:	INT32
Range:	N/A
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	N/A
Remarks:	Increments when a duplicate station is detected. When this (1 byte) counter reaches 255, it holds this maximum value until cleared.

Parameter Name:	**PFBERRLASBAD**
Specific to Block:	PBIM
Description:	Active Station List Invalid
Data Type:	INT32
Range:	N/A
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	N/A
Remarks:	Increments when the active station list is invalid because of multiple network errors. Bad wiring or hardware causes this error. When this (1 byte) counter reaches 255, it holds this maximum value until cleared.

Parameter Name:	**PFBERREVTOVR**
Specific to Block:	PBIM
Description:	Event Queue Overflow
Data Type:	INT32
Range:	N/A
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	N/A
Remarks:	Increments when a new event occurred before the last one was processed.

Parameter Name:	**PFBERRINTFLT**	
Specific to Block:	PBIM	
Description:	Internal Error Code	
Data Type:	INT32	
Range:	N/A	
Default:	0	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	PFBERRARG	
Remarks:	Internal Error Code (configuration error or runtime fault).	

Parameter Name:	**PFBERRARG**	
Specific to Block:	PBIM	
Description:	Internal Error Argument	
Data Type:	INT32	
Range:	N/A	
Default:	0	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	PFBERRINTFLT	
Remarks:	Additional information about the internal error code.	

4.2.10 Debug Parameters

Parameter Name:	**PFBINSTATUS**	
Specific to Block:	PBIM	
Description:	[In] Status Register	
Data Type:	INT32	
Range:	0xE0	PBI IOM is not online with Profibus bus. Issue the PFB_START_BUS command to transition online.
	0xE1	PBI IOM is online with Profibus bus.
	0xE2	PBI IOM is in 'serial configuration' mode. A terminal is being used to download Profibus configuration files.
Default:	0	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	N/A	
Remarks:		

Parameter Name:	**PFBINCMDECHO**

Specific to Block:	PBIM	
Description:	[In] Cmd Reply Register	
Data Type:	INT32	
Range:	0x00	No command.
0x01	PFB_BUS_RUN_MODE has been processed.	
0x02	PFB_BUS_CLR_MODE has been processed.	
0x03	PFB_CLR_ERR_CNT has been processed.	
0x04	PFB_START_BUS has been processed.	
Default:	0	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	PFBOUTCMDARG	
Remarks:		

Parameter Name:	**PFBOUTCMDCTL**	
Specific to Block:	PBIM	
Description:	[Out] Cmd Control Register	
Data Type:	INT32	
Range:	0x00	No command in CMD Argument register.
0x01	Issue the command that is in the CMD Argument register.	
Default:	0	
Config Load:	No	
Access Lock:	Application Developer	
Residence:	CEE	
Related Parameters:	PFBOUTCMDARG	

Remarks:	

Parameter Name:	**PFBOUTCMDARG**	
Specific to Block:	PBIM	
Description:	[Out] Cmd Argument Register	
Data Type:	INT32	
Range:	0x00	No command.
0x01	PFB_BUS_RUN_MODE, Put Profibus into Run mode.	
0x02	PFB_BUS_CLR_MODE, Put Profibus into Clear mode.	
0x03	PFB_CLR_ERR_CNT, Clear counters in Status assembly.	
0x04	PFB_START_BUS, Put PBIIOM online with Profibus bus.	
Default:	0	
Config Load:	No	
Access Lock:	Application Developer	
Residence:	CEE	
Related Parameters:	N/A	
Remarks:		

Parameter Name:	**OUTCTRLREG**	
Specific to Block:	PBIM	
Description:	Output Control Register	
Data Type:	INT32	
Range:	N/A	
Default:	0	
Config Load:	No	
Access Lock:	Application Developer	
Residence:	CEE	

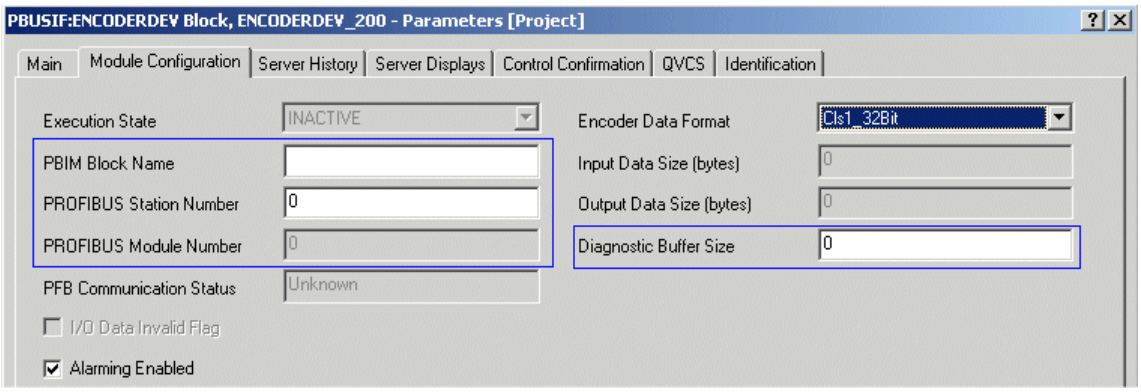
Related Parameters:	N/A
Remarks:	

5 Commonality of PROFIBUS I/O Device/Module Blocks

In subsequent sections, both generic and custom PROFIBUS I/O device/module blocks are described. Certain attributes and parameters are common to all PROFIBUS I/O Module blocks, and are captured in this section in order to avoid duplication of information. The device/module block represents the module or device of interest and is standalone/independent block type. For example, the following are typical devices that would be represented by a device/module block:

- Modular I/O, such as an Analog Input card/module that is physically located in slot number 6 of an I/O chassis on PROFIBUS. This module would typically be one of several modules in a rack/rail connected with PROFIBUS through a PROFIBUS station or gateway device. The single physical device/module is represented by a single PROFIBUS module, and a single Experion block.
- A complex device such as a motor drive (frequency converter). The drive is directly connected to PROFIBUS, and may contain multiple (virtual) modules. Therefore, in this case there is a single physical device, represented by multiple PROFIBUS data modules, and a single Experion block.

All of the parameters described in this section appear on the Module Configuration tab, such as the following:



Related topics

"PROFIBUS I/O Device/Module Blocks configuration" on page 66
"PROFIBUS I/O Device/Module Blocks operation" on page 68

5.1 PROFIBUS I/O Device/Module Blocks configuration

All PROFIBUS IOM blocks have the following four configuration parameters in common: SRCBLOCK, STATION, MODULE and DIAGBUFSIZE.

If any of the first three parameters is invalid, the load may succeed without error. The user will need to debug the problem based on the value of the DEVSTATUS parameter after load.

If the DIAGBUFSIZE parameter is zero, Diagnostic Message retrieval is disabled for this IOM. Diagnostic Channel blocks may still be assigned to the IOM, but during load a 'Configuration mismatch warning' will be generated.

Parameter Name:	**SRCBLOCK**
Specific to Block:	All PROFIBUS I/O Module Blocks
Description:	PBIM Block Name
Data Type:	**BLOCKID**
Range:	N/A
Default:	0
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	STATION, MODULE
Remarks:	Each PROFIBUS I/O Device/module block must be configured to identify the PBIM block that will serve its I/O data.

Parameter Name:	**STATION**
Specific to Block:	All PROFIBUS I/O Module Blocks
Description:	PROFIBUS Station Number
Data Type:	INT32
Range:	0 - 125
Default:	0
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	SRCBLOCK, MODULE
Remarks:	The entry should match the actual PROFIBUS configuration.

Parameter Name:	**MODULE**
Specific to Block:	All PROFIBUS I/O Module Blocks
Description:	PROFIBUS Module Number
Data Type:	INT32
Range:	0 - 100
Default:	0
Config Load:	Yes
Access Lock:	Application Developer

Residence:	CEE
Related Parameters:	SRCBLOCK, STATION
Remarks:	The entry should match the actual PROFIBUS configuration. Note that this parameter is not configurable for certain device-specific IOM blocks, and is fixed at zero.

WARNING

It is possible for the user to make configuration errors in the configuration of the PBIM module configuration table and/or the associated I/O Module and channel blocks which reference the PBIM block that could cause unpredictable behavior of output devices. For example it is possible to configure two blocks that are writing the same byte(s) in the PBIM output assembly, without the generation of a configuration error. Therefore, it is recommended that all address, offset, and data size parameters be checked for accuracy prior to block activation.

Parameter Name:	**DIAGBUFSIZE**
Specific to Block:	All PROFIBUS I/O Module Blocks
Description:	Configure the size of the Profibus Diagnostic Messages buffer in auxiliary memory.
Data Type:	16-Bit Unsigned Integer
Range:	0 to 244 (Profibus allows a maximum of 244 message bytes)
Default:	0
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	SRCBLOCK, STATION
Remarks:	A zero parameter value disables Diagnostic Message retrieval for this device IOM.

5.2 PROFIBUS I/O Device/Module Blocks operation

This section describes the operational characteristics that are common to all PROFIBUS I/O Device/module blocks.

Related topics

“Activation” on page 68
“PROFIBUS I/O Device/Module Blocks operational parameters ” on page 68
“PROFIBUS I/O Device/Module blocks diagnostic alarms” on page 69

5.2.1 Activation

Like all Experion I/O module blocks, the Execution State (EXECSTATE) parameter must be ACTIVE in order for I/O data to be processed by the PROFIBUS device/module blocks. It is recommended that the associated PBIM block be activated prior to the activation of the device/module block to avoid warnings when activating the I/O Device/Module blocks.

5.2.2 PROFIBUS I/O Device/Module Blocks operational parameters

All PROFIBUS IOM blocks are equipped the following status parameter that is useful in debugging configuration and communication problems:

Parameter Name:	**DEVSTATUS**		
Specific to Block:	All PROFIBUS I/O Module Blocks		
Description:	PROFIBUS Communication Status : a composite status		
Data Type:	Enumeration of DEVICESTATUS		
Range:	0	Unknown	Default value
	1	NoPbimRef	Invalid or missing PBIM reference. Resolution: Reconfigure the PBIM block reference and reload.
	2	NotRegistered	Block not registered with PBIM block. Resolution: Ensure a matching configuration between PBIM table and IOM block, including Station number, Module Number, and Data Offsets that correspond to the data expected by the block.
	3	PbimInactive	PBIM block is INACTIVE
	4	PbimCommErr	PBIM communication errors, probably between the PBIM block and the SST-PB3-CLX-HWL. Resolution: Investigate the PBIM block's communication status.
	5	PfbOffline	PBIM block is not online with PROFIBUS
	6	PfbNotInRun	PROFIBUS Run mode is not enabled (on PBIM block), which disables the communication of output values from the CEE.
	7	DevInactive	EXECSTATE is INACTIVE
	8	DevNotConfig	PROFIBUS network is not configured with a device with the specified station/module numbers.
	9	DevOffline	The slave device is absent or returning errors on the PROFIBUS network.
	10	DevOnline	OK state; Device is configured and communicating.
Default:	0 - Unknown		

Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	PBIM_SST block parameters: EXECSTATE, PFBMASCFG02, PFBSTNCFG[..], PFBSTNACT[..]
Remarks:	

Parameter Name:	**DATAINVALID**	
Specific to Block:	All PROFIBUS I/O Module Blocks	
Description:	I/O Data Invalid Flag	
Data Type:	Boolean	
Range:	Off (0)	Valid I/O data is available
	On (1)	Valid I/O data is not available
Default:	Off (0)	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	DEVSTATUS	
Remarks:	DATAINVALID is Off only if DEVSTATUS = DevOnline This parameter is intended for use in custom logic or alarm processing.	

5.2.3 PROFIBUS I/O Device/Module blocks diagnostic alarms

When active, all I/O Device/module blocks report diagnostic alarms in the event that communication has been disrupted between the block and the end device. Specifically, PROFIBUS IOM blocks may generate the following system diagnostic alarms:

System Diagnostic Alarm	Description
No PBIM Reference	Generated if the SRCBLOCK parameter is missing or contains an invalid PBIM reference.
PBIM Registration Error	Generated if the block is not registered with PBIM block. This indicates a configuration mismatch between this block and the PBIM block with regard to the Station number, Module Number, and Data Offsets that correspond to the data expected by the block.
PBIM Inactive	Generated if the PBIM block is INACTIVE.
PBIM Comm Error	Generated if there are communication difficulties between the controller and the SST-PB3-CLX-HWL.
PROFIBUS Offline	Generated if the PBIM block is not online with PROFIBUS.
PROFIBUS Not In Run	Generated if the PROFIBUS Run mode is not enabled (on PBIM block).
Device Not Configured	Generated if the PROFIBUS network is not configured with a device with the specified station/module numbers.
Device Offline	Generated if the slave device is absent or returning errors on the PROFIBUS network.

Another important scenario is a hardware fault while the station remains powered and connected to PROFIBUS. Modular (rack/rail/bus) PROFIBUS I/O from various vendors has revealed inconsistent different behavior when one or more I/O modules is removed from the station, specifically the following:

- The PFBSTNACT[x] parameter of the PBIM block that represents this station goes to the OFF state, thus causing the DEVSTATUS parameter of the device/module block to go to the DEVOFFLINE state, thus causing the generation of an alarm.
- The PFBSTNACT[x] parameter of the PBIM block that represents this station remains unchanged in the ON state, thus causing no change in the DEVSTATUS parameter of the device/module block which is in the DEVONLINE state, thus no alarm is generated.

WARNING

Because of the inconsistent behavior described above, and the fact that PROFIBUS device specific diagnostic information is not available in the initial release of PROFIBUS, it is recommended that the user conduct fault analysis testing to ensure appropriate alarm behavior when integrating Experion with 3rd party PROFIBUS I/O.

6 Commonality of PROFIBUS I/O Channel Blocks

In subsequent sections, both generic and custom PROFIBUS I/O channel blocks are described. Certain attributes and parameters are common to all PROFIBUS I/O Module blocks, and are captured in this section in order to avoid duplication of information.

Related topics

“PROFIBUS I/O Channel Blocks parameters” on page 72

6.1 PROFIBUS I/O Channel Blocks parameters

Related topics

"PROFIBUS I/O Channel Blocks configuration parameters" on page 72
"PROFIBUS I/O Channel Blocks operational parameters" on page 72

6.1.1 PROFIBUS I/O Channel Blocks configuration parameters

Parameter Name:	**BADCONFIGFL**	
Specific to Block:	All PROFIBUS Channel Blocks	
Description:	Configuration conflict between channel block and device/module block detected at block activation.	
Data Type:	Boolean	
Range:	Off(0)	No configuration conflict
	On(1)	Configuration conflict/mismatch detected.
Default:	Off(0)	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	DATAOFFSET, DATASIZE INCONNSIZE,OUTCONNSIZE (of PBI_DEVICE block)	
Remarks:	• When asserted, this parameter indicates an invalid combination of configuration parameters. Specifically, DATAOFFSET + DATASIZE must be less than or equal to the INCONNSIZE or OUTCONNSIZE of the associated device/module block. • The configuration is validated against the device/module block when the CM is activated. Thus, this parameter will not be asserted until the CM is activated. • This parameter is not asserted if the module block is not loaded or is inactive.	

6.1.2 PROFIBUS I/O Channel Blocks operational parameters

Input Channel Block Parameters

Parameter Name:	**BADINPUTSTS**	
Specific to Block:	All PROFIBUS Input Channel Blocks	
Description:	Communications error or failure between block and input device	
Data Type:	Boolean	
Range:	Off(0)	No communications errors.
	On(1)	Communication error, resulting in no data flow between block and end device.
Default:	Off(0)	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	

Related Parameters:	BADCONFIGFL, DEVSTATUS (of PBI_DEVICE block)
Remarks:	When asserted, this parameter indicates that data is not actively flowing from the output device to the output channel block. This can occur due to any of the following conditions: If any of the containing CM, parent device/module block or PBIM block are INACTIVE If the device/module block is not in the DEVONLINE state.

Output Channel Block Parameters

Parameter Name:	**BADOUTPUTSTS**	
Specific to Block:	All PROFIBUS Output Channel Blocks	
Description:	Communications error or failure between block and input device	
Data Type:	Boolean	
Range:	Off(0)	No communications errors.
	On(1)	Communication error, resulting in no data flow between block and end device.
Default:	Off(0)	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	BADCONFIGFL, DEVSTATUS (of PBI_DEVICE block)	
Remarks:	When asserted, this parameter indicates that data is not actively flowing from the output device to the output channel block. This can occur due to any of the following conditions: If any of the containing CM, parent device/module block or PBIM block are INACTIVE If the device/module block is not in the DEVONLINE state.	

7 Generic PROFIBUS I/O Blocks

Related topics

“Generic PROFIBUS I/O Blocks overview” on page 76
“PBI_DEVICE block” on page 79
“Generic Input Channel Block” on page 82
“Generic Output Channel Block” on page 93

7.1 Generic PROFIBUS I/O Blocks overview

The 'Generic' PROFIBUS blocks described in this section consist of a module block, an input channel block and an output channel block, that are capable of being configured to provide a simple interface to *most* PROFIBUS DP devices. The use of the term 'generic' stems from the fact that the blocks have *not* been designed for the sole purpose of interfacing to a specific device.

7.1.1 Generic PROFIBUS I/O Blocks Template Names

The block template names for the generic blocks are the following:

- PBI_DEVICE - Generic device/module block
- PBI_INCHAN - Generic input channel block
- PBI_OUTCHAN - Generic output channel block

7.1.2 Data Formats

Because PROFIBUS DP does not enforce the use of a standard for structured data, such as floating point values, integer values, Boolean/discrete values, enumeration ordinals, etc., there exists a great variation in how data messages are formatted and interpreted amongst the vendor community. Thus, data interpretation is a complex problem for the controller. In fact, it would be virtually impossible to create a single function block that could anticipate every possible means of data representation.

The data formats for the messages for a particular device are generally specified in the technical documentation that accompanies the device. The GSD file generally does not contain all of the information necessary to interpret or assemble the data messages for a particular device.

7.1.3 Numeric/Real Data Types

Typical Numeric Data Representation on PROFIBUS DP

Numerical data (such as real or integer numbers) is most commonly transported across PROFIBUS DP in integer format. Real numbers are converted to/from integer values through a simple linear conversion process.

For example, the following equations show how a 4-20 mA value could be represented using a 16 bit unsigned integer.

- Bottom of raw integer value range = -20,000 = 4 mA = 0%
- Top of raw integer value range = 20,000 = 20 mA = 100%

Each equation represents a unique point on a linear equation, and (according to basic algebra) any two points define a line, from which a linear conversion equation can be derived.

Numeric data handling capability in the generic channel blocks

The generic blocks are configured with the necessary linear scaling parameters necessary to convert I/O data from/to integer format.

Five integer formats are supported for both input and output channel blocks. The table below lists the data types, sizes and the minimum and maximum mathematical ranges for the data types. This information is a property of the device of interest and must be known by the configuration engineer in order to properly configure the generic input and output channel blocks.

Data Type	Description	Size (bytes)	Minimum Value	Maximum Value
BYTE_SIGNED	Signed 8 bit integer	1	-128	127

Data Type	Description	Size (bytes)	Minimum Value	Maximum Value
BYTE_UNSIGND	Unsigned 8 bit integer	1	0	256
WORD_SIGNED	Signed 16 bit integer	2	-32,767	32,768
WORD_UNSIGND	Unsigned 16 bit integer	2	0	65,536
DWORD_SIGNED	Signed 32 bit integer	4	-2,147,483,648	2,147,483,647

7.1.4 Discrete/Boolean Data

Discrete Data Representation on PROFIBUS DP

Discrete (Boolean) data is generally represented on PROFIBUS DP as a specific bit, which may be packed into an array of bits, where each of 8 bits per byte represents a different discrete value.

Discrete data handling capability in the generic channel blocks

The generic blocks are configured with the necessary parameters to specify the particular byte and bit necessary to convert discrete I/O data from/to integer format.

7.1.5 Scope of Application

The information in this section is intended to provide the available information necessary to determine if the generic blocks are applicable for use with a particular device. Note that this determination is not easily made given that it also requires intimate knowledge of the input/output data message structure for the device of interest.

Application Constraints

The data representation of various PROFIBUS DP devices was evaluated in the design of the generic blocks, which can interface with most, but not all, PROFIBUS DP devices. The following constraints bound the scope of application for the use of the generic blocks:

- The generic blocks provide the CEE with I/O data of FLOAT64 and BOOLEAN data types only, for analog and discrete devices, respectively. Note that I/O parameters can be connected to blocks such as the TypeConvert block in order to effectively interpret other data types.
- Although analog/numeric data is exposed to the control process in the FLOAT64 data type, as described above, the generic blocks interpret/package all numerical data in integer format for transport from/to PROFIBUS. The following integer formats are supported for data transport by the generic channel blocks:
  - Signed or unsigned 16 bit integer (most commonly used)
  - Signed or unsigned 8 bit integer
  - Signed 32 bit integer
- The generic blocks interpret/package all discrete/Boolean data as an individual bit, where the byte and bit number can be specified. Conventional logic polarity is assumed, whereby a '1' corresponds to an On/True condition and a '0' corresponds to an Off/False condition. Note that the use of logic blocks, such as the NOT block, can be used to effectively invert the polarity of the logical conditions.
- The generic blocks are capable of interpreting I/O data that is of a fixed format; I/O data of variable format cannot be interpreted. An example of a fixed format device is a 4-channel AI module that provides an 8-byte input data structure, with 2 bytes representing each channel. An example of a variable format data structure is a 6-byte output data structure where the first two bytes specify a numerical parameter identifier (selects one of several parameters) and the remaining 4 bytes represent the value that is being stored.
- The generic output channel blocks do not provide back-initialization capability to any regulatory control blocks that are connected.

Configuration Limits and Considerations

The following configuration limits apply to the use of the generic PROFIBUS blocks:

- A maximum of up to 16 input channels and 16 output channel blocks can be associated with the device/module block.
- Each input and output channel is capable of mapping/interpreting up to 8 numerical values. At 16 channels per module, this provides a maximum of 128 numerical values per device/module.
- Each input and output channel is capable of mapping/interpreting up to 32 discrete values. At 16 channels per module, this provides a maximum of 512 discrete values per device/module.
- Input and output parameter names are fixed at the parameter names indicated in the following table. However, 24 character descriptors are configurable on each parameter.

	Input Channel Block	Output Channel Block
Numeric Parameter Name	PV[0-7]	OP[0-7]
Discrete Parameter Name	PVFL[0-31]	OPFL[0-31]

- All data sizes are indicated in units of bytes.
- All byte and bit offsets are zero based specifications. Therefore, the first byte of a data message is considered byte 0, not byte 1. A data message of 8 bytes in size would span bytes 0-7.
- All data offsets are indicated in units of bytes and are 'left justified', meaning that byte 0 is the byte at the lowest memory address location.
- All bit offsets are made relative to a particular byte, and therefore span the range 0 to 7. Bit offsets are 'right justified', meaning that when a byte is presented in binary numerical format, bit 0 is on the right side. In the following example, only bit 0 is set: 00000001
- Although the input and output channel blocks are capable of specifying and interpreting a discrete value in a single bit, the entire byte (which contains the referenced bit) is read and written by the input/output channel blocks, respectively. Although this is not a problem for inputs, it does present a problem for outputs. For example, if two different output channel blocks are used to write discrete output values that are contained within the same byte of the channel output data message, the execution order of the channel blocks and/or their containing Control Modules will determine which values are written to the device. Thus, it is recommended that all discrete output channel values that are contained within a specific byte be referenced from a single output channel block.
- Numerical inputs and outputs do not support underrange or overrange protection in the form of fail-safe behavior nor are alarms generated at such limits.

7.2 PBI_DEVICE block

The PBI_DEVICE block is the generic device/module block and represents the slave device/module of interest.

7.2.1 PBI_DEVICE block Channel Assignment

The PBI_DEVICE block allows assignment of up to 32 channels. The following considerations apply:

- Channels 0-15 are reserved for input channels (PBI_INCHAN).
- Channels 16-31 are reserved for output channels (PBI _OUTCHAN).
- Only a small subset of the 32 available channels is typically required to create an interface to a device. For example:
  - An input-only device would not utilize any output channels, nor would an output-only device utilize any input channels.
  - An input (or output) device that supplies (or accepts) only a single parameter would require the use of only one input (or output) channel.
- Because each input and output channel block is capable of processing up to 8 numeric and 32 discrete I/O values, the user has some amount of flexibility in the channel assignment process. For example, if a particular PROFIBUS device supplies four input numeric parameters, the configuration engineer could use as many as four input channel blocks (with one parameter processed on each block) or as few as only one input channel block (with four parameters processed on each block).

7.2.2 PBI_DEVICE block configuration

Certain properties and parameters of device/module blocks are common to all such blocks and have been listed above in the *Commonality of PROFIBUS I/O Device/Module Blocks* Section.

Additionally, the generic device/module block requires that the user input the size and format of the data for the input and output messages. It is imperative that the correct data sizes are input. Entering a value for the input/output messages that is too large or too small will not be detected and could be detrimental to the process. For example, if a particular device outputs 8 bytes of data, but 10 bytes are specified in the configuration of the device/module block, 10 bytes will be written into the PBIM assembly data each time the block executes, causing 2 bytes to be written into a memory area which (most likely) is the output data message for a different device.

When configuring the Experion blocks, all data sizes are specified in bytes, although the PROFIBUS configuration may specify data sizes in bytes or words.

For each slave module on PROFIBUS, raw data is transported as either a string of (8 bit) bytes or a string of (16 bit) words. This data format information is embedded in the GSD file and can be extracted from the PROFIBUS configuration available through the SST PROFIBUS Configuration Tool. Specific examples are cited in the parameter definitions below.

Parameter Name:	**DEVINPUTFMT**	
Specific to Block:	PBI_DEVICE	
Description:	Input Data Format	
Data Type:	Enumeration of DeviceFormat	
Range:	NOTUSED	0
	BYTE8	1
	WORD16	2
Default:	NOTUSED	
Config Load:	Yes	

Access Lock:	Application Developer	
Residence:	CEE	
Related Parameters:	DEVOUTPUTFMT, INCONNSIZE, OUTCONNSIZE	
Remarks:	See the PROFIBUS configuration for the module of interest, which lists the data size and format. The following are examples of the appropriate value of the DEVINPUTFMT parameter based on the PROFIBUS configuration:	
	PROFIBUS Configuration	*DEVINPUTFMT*
	In4words	Word16
	Out6words	NotUsed
	InOut2bytes	Byte8
	In2bytesOut1byte	Byte8

Parameter Name:	**DEVOUTPUTFMT**	
Specific to Block:	PBI_DEVICE	
Description:	Output Data Format	
Data Type:	Enumeration of DeviceFormat	
Range:	NOTUSED	0
	BYTE8	1
	WORD16	2
Default:	NOTUSED	
Config Load:	Yes	
Access Lock:	Application Developer	
Residence:	CEE	
Related Parameters:	DEVINPUTFMT, INCONNSIZE, OUTCONNSIZE	
Remarks:	See the PROFIBUS configuration for the module of interest, which lists the data size and format. The following are examples of the appropriate value of the DEVOUTPUTFMT parameter based on the PROFIBUS configuration:	
	PROFIBUS Configuration	*DEVINPUTFMT*
	In4words	NotUsed
	Out6words	Word16
	InOut2bytes	Byte8
	In2bytesOut1byte	Byte8

Parameter Name:	**INCONNSIZE**	
Specific to Block:	PBI_DEVICE	
Description:	Input Data Size in bytes	
Data Type:	INT32	
Range:	0 - 244	
Default:	0	
Config Load:	Yes	
Access Lock:	Application Developer	
Residence:	CEE	
Related Parameters:	DEVINPUTFMT, DEVOUTPUTFMT, OUTCONNSIZE	

Remarks:	See the PROFIBUS configuration for the module of interest, which lists the data size and format. The following are examples of the appropriate value of the INCONNSIZE parameter based on the PROFIBUS configuration:	
	PROFIBUS Configuration	*INCONNSIZE*
	In4words	8
	Out6words	0
	InOut2bytes	2
	In2bytesOut1byte	2

Parameter Name:	**OUTCONNSIZE**	
Specific to Block:	PBI_DEVICE	
Description:	Output Data Size in bytes	
Data Type:	INT32	
Range:	0 - 244	
Default:	0	
Config Load:	Yes	
Access Lock:	Application Developer	
Residence:	CEE	
Related Parameters:	DEVINPUTFMT, DEVOUTPUTFMT, INCONNSIZE	
Remarks:	See the PROFIBUS configuration for the module of interest, which lists the data size and format. The following are examples of the appropriate value of the OUTCONNSIZE parameter based on the PROFIBUS configuration:	
	PROFIBUS Configuration	*OUTCONNSIZE*
	In4words	0
	Out6words	12
	InOut2bytes	2
	In2bytesOut1byte	1

7.2.3 PBI_DEVICE block operation

Refer to *Operation* in the *Commonality of PROFIBUS I/O Device/Module Blocks* Section for operational characteristics and parameters.

7.3 Generic Input Channel Block

Refer to the *Commonality of PROFIBUS I/O Channel Blocks* Section for attributes that are common to all PROFIBUS channel blocks.

7.3.1 Generic Input Channel Block configuration

Channel Assignment

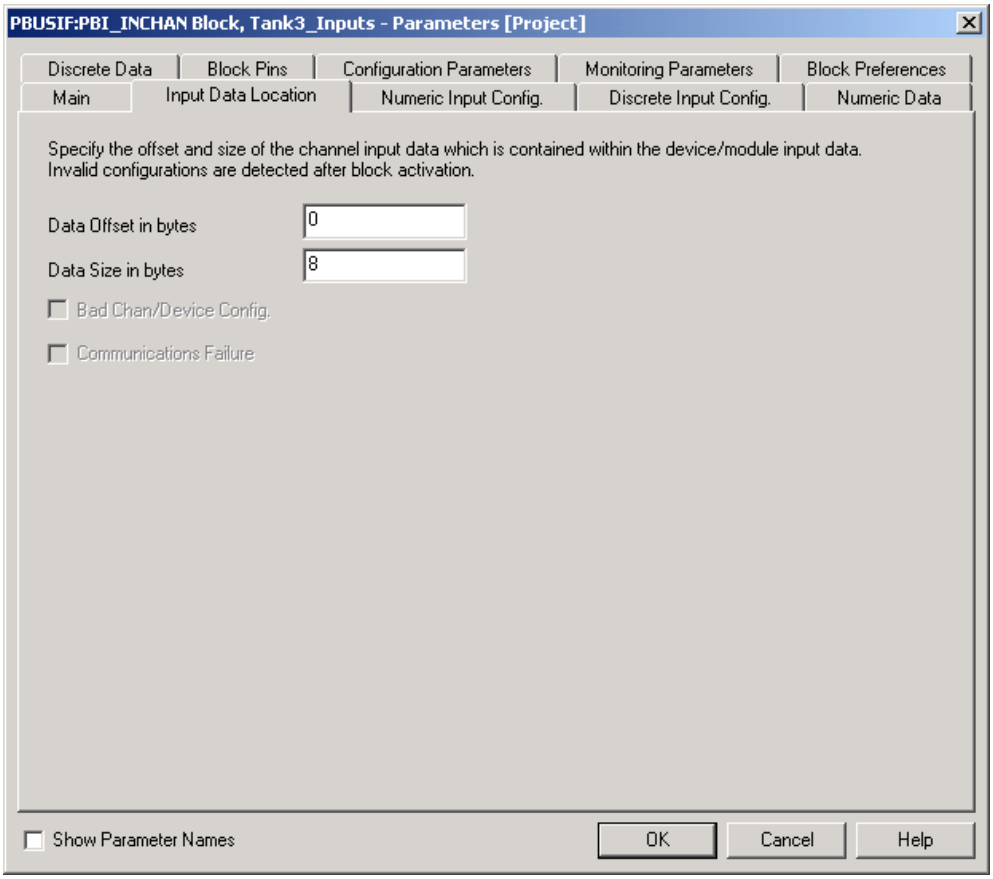
The generic input channel block (PBI_INCHAN) is assigned to a PBI_DEVICE block within the same CEE. Channels 0-15 of the PBI_DEVICE block are reserved for assignment by this block type. Like other Experion I/O modules, both the Device/module block and the CM containing the input or output channels must be assigned to the same CEE prior to assigning the channel to the device/module block.

Pin Exposure

Because the generic input channel block is capable of processing up to 8 numeric inputs and 32 discrete inputs or any combination thereof, the user must choose the appropriate parameters for pin exposure (for connection with other blocks) as executed on the Block Pins tab of the Control Builder form. One numeric input parameter and one discrete input parameter are exposed as block pins by default.

Specifying the Input Channel Data Message Location

The following is the configuration tab for the input channel data message location:



The input channel block extracts data from within the device block's input message structure, the size of which is specified during the configuration of the device block's INCONNSIZE parameters. An offset and a size parameter are specified to indicate the exact location of the data for this channel block within the device/module

input data message. Therefore, the data references made during the configuration of the input channel block cannot exceed the boundaries of the input message structure, which is the most common cause of configuration errors.

If the user enters inappropriate values for the data size or offset, which exceed the boundaries of the device/module block message, such errors are not detectable on load. However, upon activation of the containing CM, a 'configuration mismatch warning' is returned on the CM. Also, the block that has the error will have its 'Bad Channel/Device Config Flag' asserted.

Notes on the sample configuration form:

- Because 8 bytes are specified, it is expected that the associated generic device/module block be configured with at least 8 bytes of input data (INCONNSIZE >= 8).
- Because the offset is listed as 0, it is expected that the first 8 bytes of the device/module block's input data message is processed by this channel. Alternatively, if this channel block required only bytes 2 and 3, the data size would be configured as 2, and the offset would be configured as 2.

7.3.2 Parameters Related to Generic Input Channel Block Configuration

The following parameters are those that appear above on the *Input Data Location* form of CB.

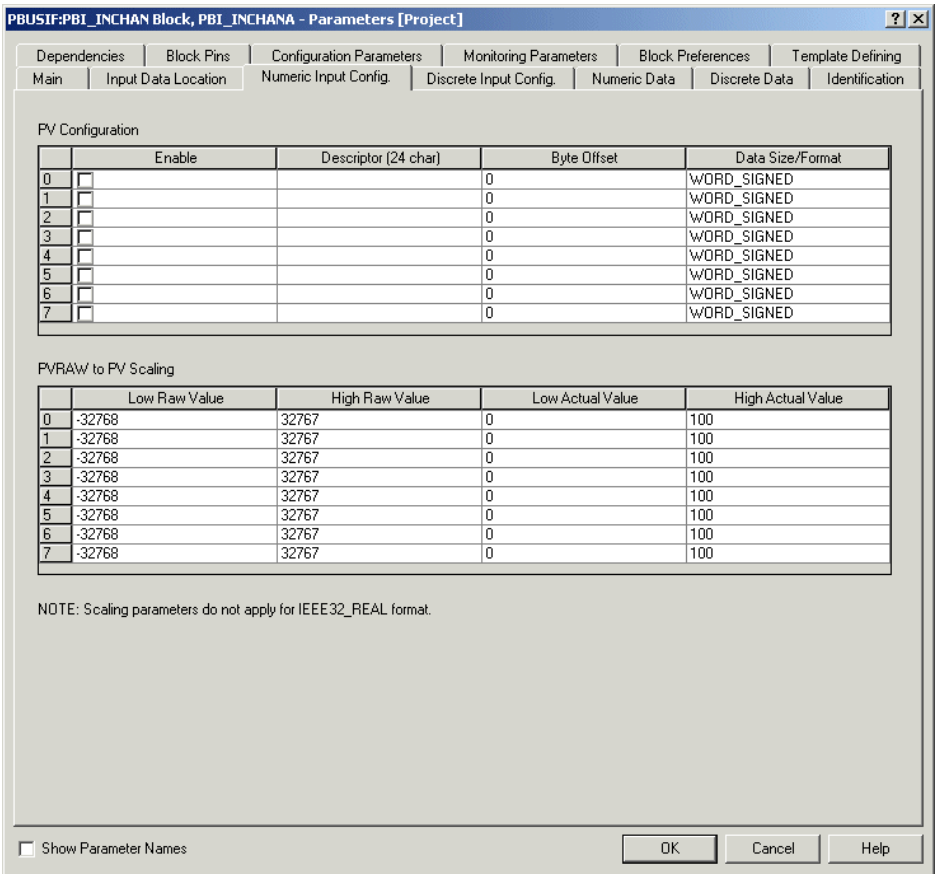
Parameter Name:	**DATAOFFSET**
Specific to Block:	PBI_INCHAN
Description:	Input Data Offset in bytes
Data Type:	INT32
Range:	0 to (PBI_DEVICE.INCONNSIZE - 1), 243 maximum
Default:	0
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	DATASIZE, INCONNSIZE (of PBI_DEVICE block)
Remarks:	This parameter specifies the starting location of the data referenced by this channel relative to the associated device/module block's input data message. DATAOFFSET + DATASIZE must be less than or equal to the INCONNSIZE of the associated device/module block. Invalid configurations are detected after block activation is attempted.

Parameter Name:	**DATASIZE**
Specific to Block:	PBI_INCHAN
Description:	Input Data Size in bytes
Data Type:	INT32
Range:	1 to (PBI_DEVICE.INCONNSIZE - DATAOFFSET), 244 maximum
Default:	8
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	DATAOFFSET, INCONNSIZE (of PBI_DEVICE block)

Remarks:	This parameter specifies the size (in bytes) of the data referenced by this channel, which is contained within the device/module block's input data message. DATAOFFSET + DATASIZE must be less than or equal to the INCONNSIZE of the associated device/module block. Invalid configurations are detected after block activation is attempted.

7.3.3 Numeric Input Data Configuration

Up to 8 numeric input parameters can be derived from the generic input channel. The configuration and scaling of the numeric input parameters is made in the *Numeric Input Configuration* tab that is shown below.



In the example configuration tab:

- The byte offset and data size specified on this form are relative to the channel block's data message, not the device/module block's data message. The entries on this form would indicate that the channel block has at least 8 bytes of input data.
- Only those indexes that are enabled will undergo configuration validation checking or execution processing.
- In the example, four of the eight available numeric parameters are enabled, and configured, indicating that this channel block will be used to input four separate input parameters, specifically for channels 1-4 of the analog input device of interest.
- The scaling parameters for the first two entries scale raw integer values into percent.
- The scaling parameters for the last two entries scale raw integer values into engineering units, which in this case is mA.
- In a normal configuration scenario, the supporting technical documentation for the device of interest would specify the raw values and associated actual values that are entered into the table.

7.3.4 Configuration Parameters Related to Numeric Input Data

Parameter Name:	**PVENABLE[0..7]**	
Specific to Block:	PBI_INCHAN	
Description:	Enable processing of the respective PV[x] parameter	
Data Type:	BOOLEAN	
Range:	Off(0)	Disable parameter
	On(1)	Enable parameter
Default:	Off(0)	
Config Load:	Yes	
Access Lock:	Application Developer	
Residence:	CEE	
Related Parameters:	PVOFFSET[..], PVRAWSIZE[..], PV[..]	
Remarks:	Invalid configuration entries for PVOFFSET[x] or PVRAWSIZE[x] will cause PVENABLE[x] to be automatically disabled during block load.	

Parameter Name:	**PVDESC[0..7]**	
Specific to Block:	PBI_INCHAN	
Description:	Descriptor of PV[x]	
Data Type:	24 Character Text String	
Range:	N/A	
Default:		
Config Load:	Yes	
Access Lock:	Application Developer	
Residence:	SR	
Related Parameters:	PV[..]	
Remarks:	Because the PV[x] parameter cannot be renamed, this text descriptor is useful in documenting the use of each enabled PV[x] .	

Parameter Name:	**PVOFFSET[0..7]**
Specific to Block:	PBI_INCHAN
Description:	Data offset in bytes from the start of the channel's input data to the starting location of the raw data representation of PV[x].
Data Type:	INT32
Range:	0 to (DATASIZE - 1), 243 maximum
Default:	0
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE

Related Parameters:	DATASIZE, PVRAWSIZE[..], PVENABLE[..], PV[..]
Remarks:	PVOFFSET + (Size of PVRAWSIZE in bytes) must less than or equal to the INCONNSIZE of the associated device/module block. Invalid combinations of PVOFFSET[x] and PVRAWSIZE[x] (which reference data outside the boundaries of the channel block data message) are detected during load and indicated as a'configuration mismatch warning', causing PVENABLE[x] to be disabled, ensuring no processing of PV[x].

Parameter Name:	**PVRAWSIZE[0..7]**				
Specific to Block:	PBI_INCHAN				
Description:	Raw integer data size and format				
Data Type:	Enumeration of PBI_Datasize				
Range:	The additional columns below indicate the physical size of each integer format and the associated minimum and maximum values for use in configuring the raw integer scaling parameters.				
	#	*Enum text*	*Size in bytes*	*Min. Raw Value*	*Max. Raw Value*
	0	BYTE_SIGNED	1	-128	127
	1	BYTE_UNSIGND	1	0	256
	2	WORD_SIGNED	2	-32,767	32,768
	3	WORD_UNSIGND	2	0	65,536
	4	DWORD_SIGNED	4	-2,147,483,648	2,147,483,647
	5	IEEE32_REAL	4	+/- 1.4 * E-45	+/- 3.4 * E38
Default:	0				
Config Load:	Yes				
Access Lock:	Application Developer				
Residence:	CEE				
Related Parameters:	DATASIZE, PVOFFSET[..], PVLORAW[..], PVHIRAW[..], PV[..]				
Remarks:	• PVOFFSET + (Size of PVRAWSIZE in bytes) must less than or equal to the INCONNSIZE of the associated device/module block. • Invalid combinations of PVOFFSET[x] and PVRAWSIZE[x] (which reference data outside the boundaries of the channel block data message) are detected during load and indicated as a'configuration mismatch warning', causing PVENABLE[x] to be disabled, ensuring no processing of PV[x]. • Scaling parameters are meaningless for the IEEE32_REAL format.				

Parameter Name:	**PVLORAW[0..7]**
Specific to Block:	PBI_INCHAN
Description:	Lower range limit of the integer input value.
Data Type:	INT32
Range:	See PVRAWSIZE[..], upon which this parameter is dependent.
Default:	-32,768
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	PVRAWSIZE[..], PVLOACT[..], PV[..]

Remarks:	• Input values less than PVLORAW[x] are invalid. • PVLOACT[x] is configured with the actual numeric value that is equivalent to PVLORAW[x].

Parameter Name:	**PVHIRAW[0..7]**
Specific to Block:	PBI_INCHAN
Description:	Upper range limit of the integer input value.
Data Type:	INT32
Range:	See PVRAWSIZE[..], upon which this parameter is dependent.
Default:	32,767
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	PVRAWSIZE[..], PVHIACT[..], PV[..]
Remarks:	Output values greater than PVHIRAW[x] are invalid. PVHIACT[x] is configured with the actual numeric value that is equivalent to PVHIRAW[x].

Parameter Name:	**PVLOACT[0..7]**
Specific to Block:	PBI_INCHAN
Description:	Low Actual Value This parameter represents the numeric equivalent of PVLORAW[x].
Data Type:	FLOAT64
Range:	
Default:	0.0
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	PVLORAW[..], PVHIACT[..], PV[..]
Remarks:	Although this parameter is usually less than PVHIACT[x] (representing the lowest numerical input value), if the inputs are scaled in a reverse fashion, this parameter can also be greater than PVHIACT[x] (thus representing the highest numerical input value).

Parameter Name:	**PVHIACT[0..7]**
Specific to Block:	PBI_INCHAN
Description:	High Actual Value This parameter represents the numeric equivalent of PVHIRAW[x].
Data Type:	FLOAT64
Range:	
Default:	100.0
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	PVHIRAW[..], PVLOACT[..], PV[..]

Remarks:	Although this parameter is usually greater than PVLOACT[x] (representing the highest numerical input value), if the inputs are scaled in a reverse fashion, this parameter can also be less than PVLOACT[x] (thus representing the lowest numerical input value).

7.3.5 Operational Parameters Related to Numeric Input Data

Parameter Name:	**PV[0..7]**
Specific to Block:	PBI_INCHAN
Description:	Process Input[x], Scaled This parameter represents the floating point Present Value (PV), as a scaled representation of the raw integer input value.
Data Type:	FLOAT64
Range:	
Default:	0.0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	PVENABLE[..], PVDESC[..], PVOFFSET[..], PVRAWSIZE[..], PVLORAW[..], PVHIRAW[..], PVLOACT[..], PVHIACT[..], PVRAW[..], PVSTS[..]
Remarks:	PV[x] is processed only if PVENABLE[x] is asserted. The PV connection is ‘intelligent' in that it will automatically pass value and status to any block that accepts status information, such as the P1 connection of a DATAACQ block or the PV connection of a PID block.

Parameter Name:	**PVRAW[0..7]**
Specific to Block:	PBI_INCHAN
Description:	Process Input[x] raw value This value is the raw field value that is transported via PROFIBUS. It is scaled by the block to calculate PV[x].
Data Type:	INT32
Range:	Varies based on PVRAWSIZE[..]
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	PV[..]
Remarks:	This parameter is primarily useful as a debugging parameter, but can be used directly as a control input, providing an input as an integer data type.

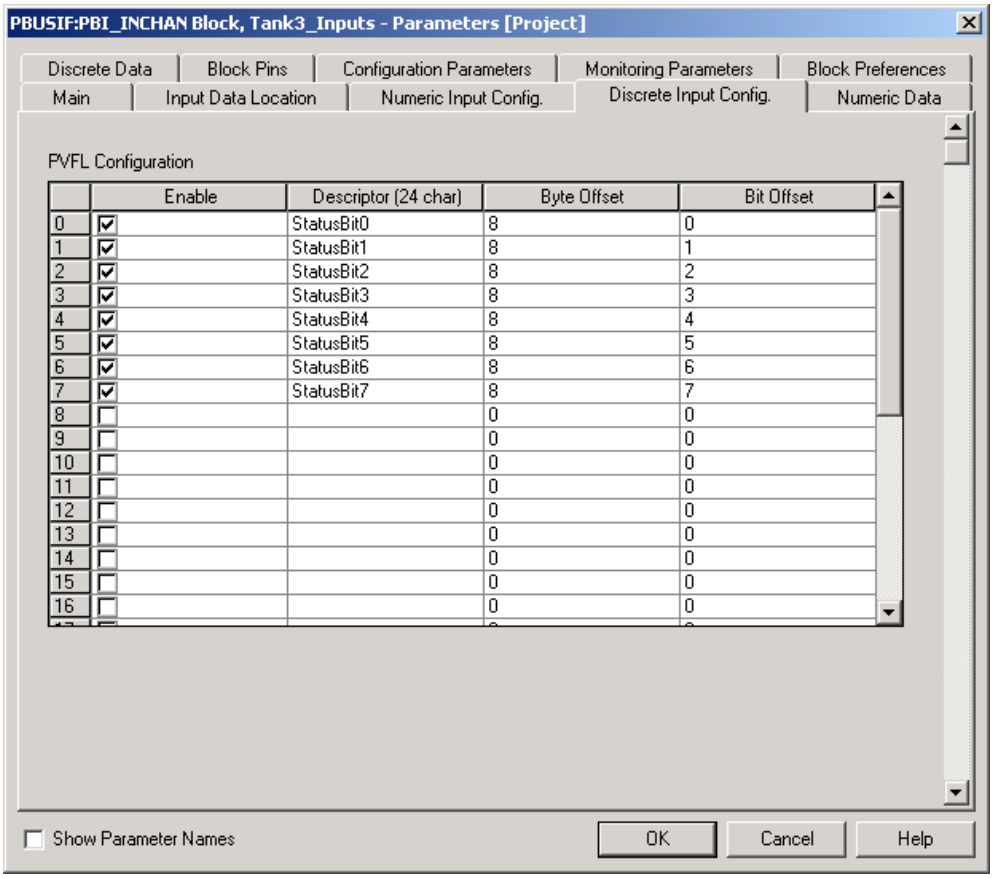
Parameter Name:	**PVSTS[0..7]**		
Specific to Block:	PBI_INCHAN		
Description:	Process Input[x] Status		
Data Type:	Enumeration of PVSTS		
Range:	0	Bad	
	1	Uncertain	(not used by this block)

	2	Normal	
	3	Manual	(not used by this block)
	4	InitAck	(not used by this block)
Default:	Bad		
Config Load:	No		
Access Lock:	View Only		
Residence:	CEE		
Related Parameters:	COMFAILFL, PVENABLE[..], PV[..]		
Remarks:	PVSTS[x] assumes the BAD state if either 1) BADINPUTSTS is asserted or 2) PVENABLE[x] is configured to the Off state.		

7.3.6 Discrete Input Data Configuration

Up to 32 discrete input parameters can be derived from the generic input channel.

The configuration and scaling of the discrete input parameters is made in the *Discrete Input Configuration tab*, which is shown below.



In the example configuration tab:

- The byte offset specified on this form is relative to the channel block's data message, not the device/module block's data message. The entries on this form would indicate that the channel block has at least 9 bytes of input data, given that the bits specified are within byte 8 (of a 0 based byte array).
- Only those indexes that are enabled will undergo configuration validation checking or execution processing.
- In the example, 8 of the 32 available discrete input parameters are enabled and configured.

- The supporting technical documentation for the device indicates the specific meaning of each of the status bits.

7.3.7 Configuration Parameters Related to Discrete Input Data

Parameter Name:	**BADPVFLSTS[0..31]**	
Specific to Block:	PBI_INCHAN	
Description:	Bad Communication Status Indicator for PVFL[x]	
Data Type:	BOOLEAN	
Range:	Off(0)	Good status
	On(1)	Bad status
Default:	Off(0)	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	PVFL[..], FLENABLE[..]	
Remarks:	This bad status indicator will become asserted if the channel block is not actively receiving valid input data from the input device for any reason, specifically if either 1) BADINPUTSTS is asserted or 2) FLENABLE[x] is configured to the Off state.	

Parameter Name:	**FLENABLE[0..31]**	
Specific to Block:	PBI_INCHAN	
Description:	Enable processing of the respective PVFL[x] parameter	
Data Type:	BOOLEAN	
Range:	Off(0)	Disable parameter
	On(1)	Enable parameter
Default:	Off(0)	
Config Load:	Yes	
Access Lock:	Application Developer	
Residence:	CEE	
Related Parameters:	FLBYTEOFFSET[..], FLBITOFFSET[..], PVFL[..]	
Remarks:	Invalid configuration entries for FLBYTEOFFSET will cause PVENABLE[x] to be internally disabled during block load.	

Parameter Name:	**FLDESC[0..31]**	
Specific to Block:	PBI_INCHAN	
Description:	Descriptor of PVFL[x]	
Data Type:	24 Character Text String	
Range:	N/A	
Default:		
Config Load:	Yes	
Access Lock:	Application Developer	
Residence:	SR	

Related Parameters:	PVFL[..]
Remarks:	Because the PVFL[x] parameter cannot be renamed, this text descriptor is useful in documenting the use of each enabled PVFL[x].

Parameter Name:	**FLBYTEOFFSET[0..31]**
Specific to Block:	PBI_INCHAN
Description:	Data offset in bytes from the start of the channel's input data to the location of the byte containing the bit that represents PVFL[x].
Data Type:	INT32
Range:	0 to (DATASIZE - 1), 243 maximum
Default:	0
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	DATASIZE, FLENABLE[x], PVFL[..]
Remarks:	Invalid entries (which reference data outside the boundaries of the channel block data message) are detected during load and indicated as a'configuration mismatch warning', causing FLENABLE[x] to be disabled, ensuring no processing of PVFL[x].

Parameter Name:	**FLBITOFFSET[0..31]**
Specific to Block:	PBI_INCHAN
Description:	Bit number that represents PVFL[x] within the specified byte of interest.
Data Type:	INT32
Range:	0 to 7
Default:	0
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	DATASIZE, FLENABLE[..], PVFL[..]
Remarks:	Bit numbers are 'right justified', meaning that the lowest bit number is at the right side of the byte. For example the following binary sequence has only bit 0 asserted: 00000001

7.3.8 Operational Parameters Related to Discrete Input Data

Parameter Name:	**PVFL[0..31]**
Specific to Block:	PBI_INCHAN
Description:	Process Input Flag[x] This parameter represents the Boolean (flag) Present Value, as extracted from the specified bit of the specified byte of interest.
Data Type:	BOOLEAN
Range:	Off(0)
	On(1)
Default:	Off
Config Load:	No

Access Lock:	View Only
Residence:	CEE
Related Parameters:	BADPVFLSTS[..], FLENABLE[..], FLDESC[..], FLBYTEOFFSET[..], FLBITOFFSET[..].
Remarks:	PVFL[x] is processed only if FLENABLE[x] is asserted. The PV connection is ‘intelligent' in that it will automatically pass value and status to any block that accepts status information, such as the P1 connection of a DATAACQ block or the PV connection of a PID block.

7.4 Generic Output Channel Block

Refer to the *Commonality of PROFIBUS I/O Channel Blocks* Section for attributes that are common to all PROFIBUS channel blocks.

7.4.1 Generic Output Channel Block configuration

Channel Assignment

The generic output channel block (PBI_OUTCHAN) is assigned to a PBI_DEVICE block within the same CEE. Channels 16-32 of the PBI_DEVICE block are reserved for assignment by this block type. Like other Experion I/O modules, both the Device/module block and the CM containing the input or output channels must be assigned to the same CEE prior to assigning the channel to the device/module block.

Pin Exposure

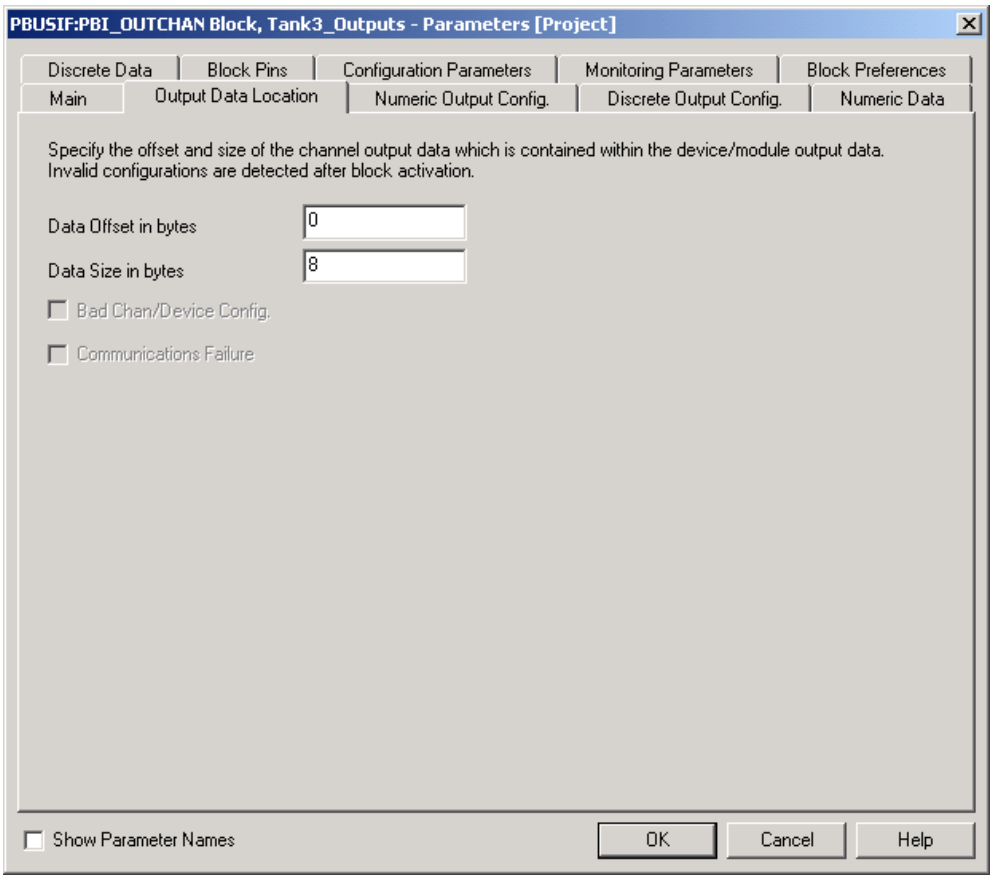
Because the generic output channel block is capable of processing up to 8 numeric outputs and 32 discrete outputs or any combination thereof, the user must choose the appropriate parameters for pin exposure (for connection with other blocks) as executed on the Block Pins tab of the Control Builder form. One numeric output parameter and one discrete output parameter are exposed as block pins by default.

Access Level Constraints on Control Outputs

A view-only access lock is enforced on the discrete and numeric output parameters of the PBI_OUTCHAN block. Therefore, the operator/engineer cannot change the value of these parameters directly. Any output parameter that is required to be capable of being changed by the operator or engineer, should be exposed as a block pin and connected to another block.

Specifying the Output Channel Data Message Location

The following is the configuration tab for the output channel data message location:



The output channel block extracts data from within the device block's output message structure, the size of which is specified during the configuration of the device block's OUTCONNSIZE parameters. An offset and a size parameter are specified to indicate the exact location of the data for this channel block within the device/module output data message. Therefore, the data references made during the configuration of the output channel block cannot exceed the boundaries of the output message structure, which is the most common cause of configuration errors.

If the user enters inappropriate values for the data size or offset, which exceed the boundaries of the device/module block message, such errors are not detectable on load. However, upon activation of the containing CM, a 'configuration mismatch warning' is returned on the CM. Also, the block that has the error will have its'Bad Channel/Device Config Flag' asserted.

Notes on the sample configuration form:

- Because 8 bytes are specified, it is expected that the associated generic device/module block be configured with at least 8 bytes of output data (OUTCONNSIZE >= 8).
- Because the offset is listed as 0, it is expected that the first 8 bytes of the device/module block's output data message is processed by this channel. Alternatively, if this channel block required only bytes 2 and 3, the data size would be configured as 2, and the offset would be configured as 2.

7.4.2 Parameters Related to Generic Output Channel Block Configuration

The following parameters are those that appear above on the *Output Data Location* form of CB.

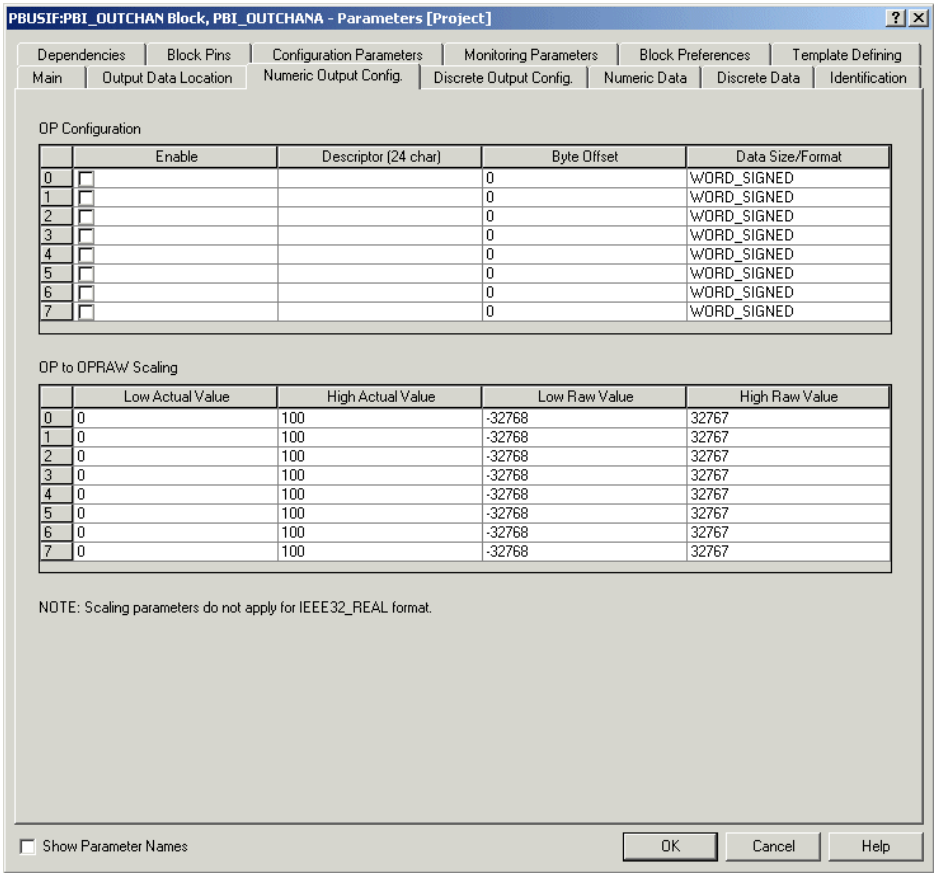
Parameter Name:	**DATAOFFSET**
Specific to Block:	PBI_OUTCHAN
Description:	Output Data Offset in bytes

Data Type:	INT32
Range:	0 to (PBI_DEVICE.INCONNSIZE - 1), 243 maximum
Default:	0
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	DATASIZE, OUTCONNSIZE (of PBI_DEVICE block)
Remarks:	This parameter specifies the starting location of the data referenced by this channel relative to the associated device/module block's output data message. DATAOFFSET + DATASIZE must be less than or equal to the OUTCONNSIZE of the associated device/module block. Invalid configurations are detected after block activation is attempted.

Parameter Name:	**DATASIZE**
Specific to Block:	PBI_OUTCHAN
Description:	Output Data Size in bytes
Data Type:	INT32
Range:	1 to (PBI_DEVICE.OUTCONNSIZE - DATAOFFSET), 244 maximum
Default:	8
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	DATAOFFSET, OUTCONNSIZE (of PBI_DEVICE block)
Remarks:	This parameter specifies the size (in bytes) of the data referenced by this channel, which is contained within the device/module block's output data message. DATAOFFSET + DATASIZE must be less than or equal to the OUTCONNSIZE of the associated device/module block. Invalid configurations are detected after block activation is attempted.

7.4.3 Numeric Output Data Configuration

Up to 8 numeric output parameters can be derived from the generic output channel. The configuration and scaling of the numeric output parameters is made in the *Numeric Output Configuration* tab that is shown below.



In the example configuration tab:

- The byte offset and data size specified on this form are relative to the channel block's data message, not the device/module block's data message. The entries on this form would indicate that the channel block has at least 4 bytes of output data.
- Only those indexes that are enabled will undergo configuration validation checking or execution processing.
- In the example, two of the eight available numeric parameters are enabled, and configured, indicating that this channel block will be used to input two separate output parameters, specifically for channels 1-2 of the analog output device of interest.
- The scaling parameters for the first entry cause the conversion of a percentage into a raw integer value.
- The scaling parameters for the second entry cause the conversion of an engineering unit (voltage) into a raw integer value.
- In a normal configuration scenario, the supporting technical documentation for the device of interest would specify the raw values and associated actual values that are entered into the table.

7.4.4 Configuration Parameters Related to Numeric Output Data

Parameter Name:	**OPENABLE[0..7]**	
Specific to Block:	PBI_OUTCHAN	
Description:	Enable processing of the respective OP[x] parameter	
Data Type:	BOOLEAN	
Range:	Off(0)	Disable parameter
	On(1)	Enable parameter

Default:	Off(0)
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	OPOFFSET[..], OPRAWSIZE[..], OP[..]
Remarks:	Invalid configuration entries for OPOFFSET[x] or OPRAWSIZE[x] will cause OPENABLE[x] to be automatically disabled during block load.

Parameter Name:	**OPDESC[0..7]**
Specific to Block:	PBI_OUTCHAN
Description:	Descriptor of OP[x]
Data Type:	24 Character Text String
Range:	N/A
Default:	
Config Load:	Yes
Access Lock:	Application Developer
Residence:	SR
Related Parameters:	OP[..]
Remarks:	Because the OP[x] parameter cannot be renamed, this text descriptor is useful in documenting the use of each enabled OP[x].

Parameter Name:	**OPOFFSET[0..7]**
Specific to Block:	PBI_OUTCHAN
Description:	Data offset in bytes from the start of the channel's input data to the starting location of the raw data representation of OP[x].
Data Type:	INT32
Range:	0 to (DATASIZE - 1), 243 maximum
Default:	0
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	DATASIZE, OPRAWSIZE[..], OPENABLE[..], OP[..]
Remarks:	OPOFFSET + (Size of OPRAWSIZE in bytes) must less than or equal to the OUTCONNSIZE of the associated device/module block. Invalid combinations of OPOFFSET[x] and OPRAWSIZE[x] (which reference data outside the boundaries of the channel block data message) are detected during load and indicated as a'configuration mismatch warning', causing PVENABLE[x] to be disabled, ensuring no processing of OP[x].

Parameter Name:	**OPRAWSIZE[0..7]**
Specific to Block:	PBI_OUTCHAN
Description:	Raw integer data size and format
Data Type:	Enumeration of PBI_Datasize

Range:	The additional columns below indicate the physical size of each integer format and the associated minimum and maximum values for use in configuring the raw integer scaling parameters.

#	*Enum text*	*Size in bytes*	*Min. Raw Value*	*Max. Raw Value*
0	BYTE_ SIGNED	1	-128	127
1	BYTE_ UNSIGND	1	0	256
2	WORD_ SIGNED	2	-32,767	32,768
3	WORD_ UNSIGND	2	0	65,536
4	DWORD_ SIGNED	4	-2,147,483,648	2,147,483,647
5	IEEE32_ REAL	4	+/- 1.4 * E-45	+/- 3.4 * E38

Default:	0
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	DATASIZE, OPOFFSET[..], OPLORAW[..], OPHIRAW[..], OP[..]
Remarks:	• OPOFFSET + (Size of OPRAWSIZE in bytes) must less than or equal to the OUTONNSIZE of the associated device/module block. • Invalid combinations of OPOFFSET[x] and OPRAWSIZE[x] (which reference data outside the boundaries of the channel block data message) are detected during load and indicated as a 'configuration mismatch warning', causing OPENABLE[x] to be disabled, ensuring no processing of OP[x]. • Scaling parameters are meaningless for the IEEE32_REAL format.

Parameter Name:	**OPLORAW[0..7]**
Specific to Block:	PBI_OUTCHAN
Description:	Lower range limit of the integer output value.
Data Type:	INT32
Range:	See OPRAWSIZE[..], upon which this parameter is dependent.
Default:	-32,768
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	OPRAWSIZE[..], OPLOACT[..], OP[..]
Remarks:	• Input values less than OPLORAW[x] are invalid. • OPLOACT[x] is configured with the actual numeric value that is equivalent to OPLORAW[x].

Parameter Name:	**OPHIRAW[0..7]**

Specific to Block:	PBI_OUTCHAN
Description:	Upper range limit of the integer output value.
Data Type:	INT32
Range:	See OPRAWSIZE[..], upon which this parameter is dependent.
Default:	32,767
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	OPRAWSIZE[..], OPHIACT[..], OP[..]
Remarks:	• Output values greater than OPHIRAW[x] are invalid. • OPHIACT[x] is configured with the actual numeric value that is equivalent to OPHIRAW[x].

Parameter Name:	**OPLOACT[0..7]**
Specific to Block:	PBI_OUTCHAN
Description:	Low Actual Value This parameter represents the numeric equivalent of OPLORAW[x].
Data Type:	FLOAT64
Range:	
Default:	0.0
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	OPLORAW[..], OPHIACT[..], OP[..]
Remarks:	Although this parameter is usually less than OPHIACT[x] (representing the lowest numerical input value), if the inputs are scaled in a reverse fashion, this parameter can also be greater than OPHIACT[x] (thus representing the highest numerical input value).

Parameter Name:	**OPHIACT[0..7]**
Specific to Block:	PBI_OUTCHAN
Description:	High Actual Value This parameter represents the numeric equivalent of OPHIRAW[x].
Data Type:	FLOAT64
Range:	
Default:	100.0
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	OPHIRAW[..], OPLOACT[..], OP[..]
Remarks:	Although this parameter is usually greater than OPLOACT[x] (representing the highest numerical input value), if the inputs are scaled in a reverse fashion, this parameter can also be less than OPLOACT[x] (thus representing the lowest numerical input value).

7.4.5 Operational Parameters Related to Numeric Output Data

Parameter Name:	**BADOPSTS[0..7]**		
Specific to Block:	PBI_OUTCHAN		
Description:	Bad Output[x] Status Flag		
Data Type:	BOOLEAN		
Range:	0	Off	OP status is good
	1	On	OP status is bad
Default:	Bad		
Config Load:	No		
Access Lock:	View Only		
Residence:	CEE		
Related Parameters:	COMFAILFL, OPENABLE[..]		
Remarks:	BADOPSTS[x] is asserted if either 1) BADOUTPUTSTS is asserted or 2) PVENABLE[x] is configured to the Off state.		

Parameter Name:	**OP[0..7]**
Specific to Block:	PBI_OUTCHAN
Description:	Output Value[x] This parameter represents the floating-point output value.
Data Type:	FLOAT64
Range:	
Default:	0.0
Config Load:	No
Access Lock:	Engineer
Residence:	CEE
Related Parameters:	BADOPSTS[..], OPENABLE[..], OPDESC[..], OPOFFSET[..], OPRAWSIZE[..], OPLORAW[..], OPHIRAW[..], OPLOACT[..], OPHIACT[..], OPRAW[..].
Remarks:	OP[x] is processed only if OPENABLE[x] is asserted. The OP connection does not support back initialization, or automatic passage of status.

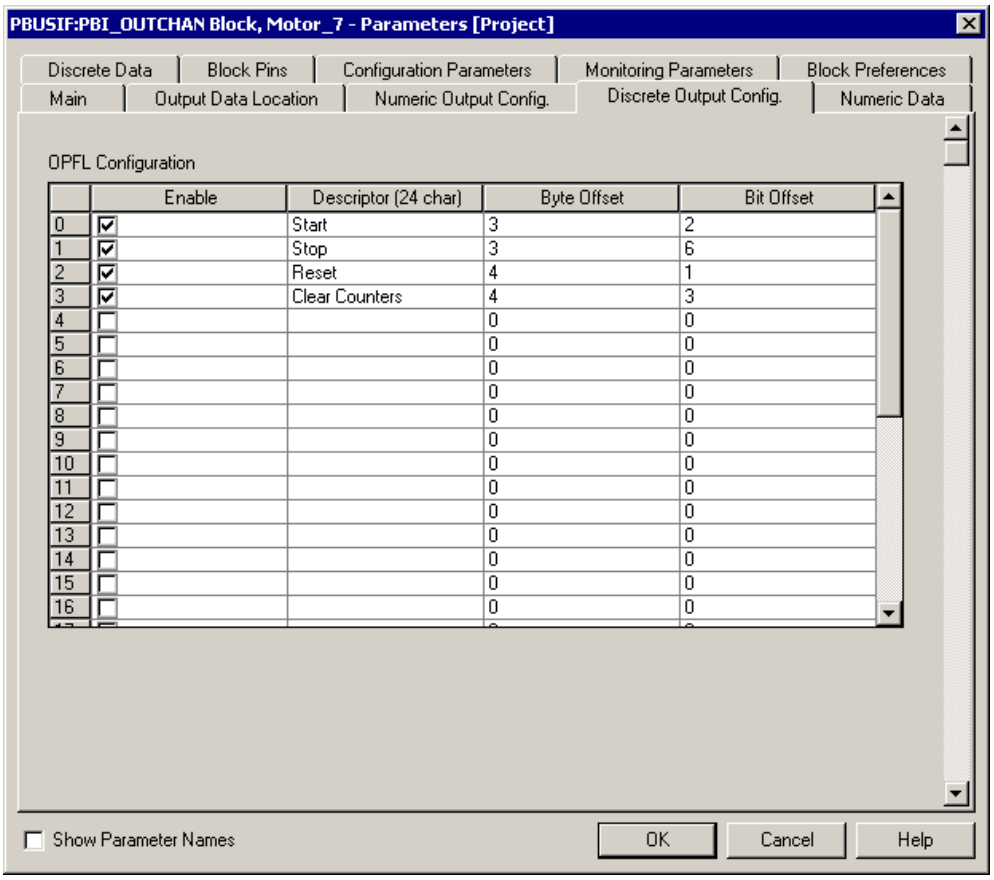
Parameter Name:	**OPRAW[0..7]**
Specific to Block:	PBI_OUTCHAN
Description:	Output Value, as a scaled integer representation of OP[x] This value is the raw field value that is transported through PROFIBUS. It is scaled by the block from OP[x].
Data Type:	INT32
Range:	Varies based on OPRAWSIZE[..]
Default:	0
Config Load:	No
Access Lock:	Engineer
Residence:	CEE

Related Parameters:	OP[..]
Remarks:	This parameter is primarily useful as a debugging parameter, but can be used directly as a control output, providing an output as an integer data type.

7.4.6 Discrete Output Data Configuration

Up to 32 discrete output parameters can be derived from the generic output channel.

The configuration and scaling of the discrete output parameters is made in the *Discrete Output Configuration* tab, which is shown below.



In the example configuration tab:

- The byte offset specified on this form is relative to the channel block's data message, not the device/module block's data message. The entries on this form would indicate that the channel block has at least 5 bytes of input data, given that the bits specified are within bytes 3 and 4 (of a 0 based byte array).
- Only those indexes that are enabled will undergo configuration validation checking or execution processing.
- In the example, 4 of the 32 available discrete output parameters are enabled and configured.
- The supporting technical documentation for the device indicates the specific meaning of each of the command parameter bits.

7.4.7 Configuration Parameters Related to Discrete Output Data

Parameter Name:	**FLENABLE[0..31]**
Specific to Block:	PBI_OUTCHAN

Description:	Enable processing of the respective OPFL[x] parameter	
Data Type:	BOOLEAN	
Range:	Off(0)	Disable parameter
	On(1)	Enable parameter
Default:	Off(0)	
Config Load:	Yes	
Access Lock:	Application Developer	
Residence:	CEE	
Related Parameters:	FLBYTEOFFSET[..], FLBITOFFSET[..], OPFL[..]	
Remarks:	Invalid configuration entries for FLBYTEOFFSET will cause OPENABLE[x] to be internally disabled during block load.	

Parameter Name:	**FLDESC[0..31]**
Specific to Block:	PBI_OUTCHAN
Description:	Descriptor of OPFL[x]
Data Type:	24 Character Text String
Range:	N/A
Default:	
Config Load:	Yes
Access Lock:	Application Developer
Residence:	SR
Related Parameters:	OPFL[..]
Remarks:	Because the OPFL[x] parameter cannot be renamed, this text descriptor is useful in documenting the use of each enabled OPFL[x].

Parameter Name:	**FLBYTEOFFSET[0..31]**
Specific to Block:	PBI_OUTCHAN
Description:	Data offset in bytes from the start of the channel's input data to the location of the byte containing the bit that represents OPFL[x].
Data Type:	INT32
Range:	0 to (DATASIZE - 1), 243 maximum
Default:	0
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	DATASIZE, FLENABLE[x], OPFL[..]

Remarks:	Invalid entries (which reference data outside the boundaries of the channel block data message) are detected during load and indicated as a 'configuration mismatch warning', causing FLENABLE[x] to be disabled, ensuring no processing of OPFL[x].

Parameter Name:	**FLBITOFFSET[0..31]**
Specific to Block:	PBI_OUTCHAN
Description:	Bit number that represents OPFL[x] within the specified byte of interest.
Data Type:	INT32
Range:	0 to 7
Default:	0
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	DATASIZE, FLENABLE[..], OPFL[..]
Remarks:	Bit numbers are 'right justified', meaning that the lowest bit number is at the right side of the byte. For example the following binary sequence has only bit 0 asserted: 00000001

7.4.8 Operational Parameters Related to Discrete Output Data

Parameter Name:	**OPFL[0..31]**	
Specific to Block:	PBI_OUTCHAN	
Description:	Process Output Flag[x] This parameter represents the Boolean (flag) Output Value, which is inserted into the specified bit of the specified byte of interest.	
Data Type:	BOOLEAN	
Range:	Off(0)	
	On(1)	
Default:	Off	
Config Load:	No	
Access Lock:	Engineer	
Residence:	CEE	
Related Parameters:	BADOPFLSTS[..], FLENABLE[..], FLDESC[..], FLBYTEOFFSET[..], FLBITOFFSET[..].	
Remarks:	OPFL[x] is processed only if FLENABLE[x] is asserted.	

Parameter Name:	**BADOPFLSTS[0..31]**	
Specific to Block:	PBI_OUTCHAN	
Description:	Bad Communication Status Indicator for OPFL[x]	
Data Type:	BOOLEAN	
Range:	Off(0)	Good status
	On(1)	Bad status
Default:	Off(0)	
Config Load:	No	
Access Lock:	View Only	

Residence:	CEE
Related Parameters:	OPFL[..], FLENABLE[..]
Remarks:	This bad status indicator will become asserted if the channel block's output data is not actively being transferred to the output device for any reason, specifically if either 1) BADOUTPUTSTS is asserted or 2) FLENABLE[x] is configured to the Off state.

8 SIMATIC I/O Module Function Blocks

Related topics

8.1 SIMATIC I/O Module Function Blocks overview

A set of device-specific function blocks is included for interfacing to the Siemens Simatic I/O family. The Simatic I/O is a rail-based, modular I/O family that utilizes the Siemens ET200M, which acts as the PROFIBUS station, and 'gateway' to PROFIBUS.

8.1.1 Block Template Names

Four I/O device module function blocks have been developed with the following template names and descriptors:

- SIMATIC_AI - Siemens Simatic Analog Input Module
- SIMATIC_AO - Siemens Simatic Analog Output Module
- SIMATIC_DI - Siemens Simatic Digital Input Module
- SIMATIC_DO - Siemens Simatic Digital Output Module

This I/O family also includes a set of mixed (input and output modules). The Experion block set does not include device-specific function block support for such modules.

8.1.2 One Size Fits All

Unlike conventional I/O module blocks, the Simatic IOM blocks allow the user to configure the number of channels that are supported on the actual module (NUMCHANSSUP parameter), thus allowing each block to represent multiple physical modules. Depending on the choice of the actual module from within the Simatic family, the following data lists the possible valid selections for the number of channels supported for each of the module types:

- AI - 8, 4, 2
- AO - 4, 2
- DI - 32, 16, 8, 4
- DO - 32, 16, 8, 4

Because the selection for the number of channels supported is internally used by the block to calculate the size of its expected input/output data, it is important that the number of channels is configured in agreement with the end device.

Despite the selection for the number of channels supported, the module block allows the assignment of up to the maximum number of channels. If a user were to assign and configure a 9^{th} channel to a module which is configured to allow a maximum of 8 channels, that channel can safely be assigned and loaded, but will be capable of communicating any data to/from the field device.

8.1.3 Utilizes 'Standard' I/O Channel blocks

The 'standard' I/O channel blocks (located in the IOCHANNEL) library are used/assigned with/to the SIMATIC I/O module blocks. Because the standard I/O channel blocks were designed for use with the standard Experion I/O, there are some parameters on the channel blocks that are not applicable to the channel block when it is assigned to a PROFIBUS module. Parameter requests for such parameters will appropriately return an error when requested.

8.1.4 SIMATIC I/O Module Function Blocks configuration

The only configuration parameter that is common to the blocks is the NUMCHANSSUP parameter that is explained above. For the analog input and output module, additional configuration input is required and is described in the subsections below. For the discrete/digital modules, the configuration of the number of channels is the only configuration that is necessary.

8.2 The Simatic Digital Input Module Function Block

8.2.1 Simatic Digital Input Module Function Block functional description

The Simatic Digital Input Module Function Block provides a configurable number of channels of discrete (binary) input data. The maximum number of channels of discrete data that this module can be configured to provide is 32.

Digital Input Channel Function Block Interface

The Simatic Digital Input Module Function Block supports a set of interfaces to the standard Experion Digital Input Channel Function Block. This interface provides both a BOOLEAN input value and a BOOLEAN signal indicating the status (GOOD vs. BAD) of the input value. The status is set to BAD when any condition exists that prevents timely update of the input value. Examples include Digital Input Module Function Block state = INACTIVE or a loss of communication to the field device.

8.2.2 Parameters Specific to the Simatic Digital Input Module Function Block

The following parameters are specific to the Simatic Digital Input Module Function Block:

Function	Parameter Name	Description
PVVAL	Process Value	This is an arrayed parameter. Each element of the array represents a single BOOLEAN input value from the corresponding Simatic Digital Input device. This is a READ-ONLY parameter. The absolute maximum size of this array is 32 elements; however, once the function block is configured and loaded, the size of this array is restricted to the value of MAXCHANS.
BADPVSTS	Bad PV Status	This is an arrayed parameter. Each element of the array represents a single BOOLEAN status value that indicates whether or not the corresponding element of the PVVAL array is valid. TRUE = BAD FALSE = GOOD This is a READ-ONLY parameter.
NUMCHANSSUP	Number of Channels Configured	This parameter defines the number of discrete data channels supported by the Simatic Digital Input device that this function block will represent. It may only have a value of 4, 8, 16, 32 This is an INTEGER parameter that the user provides at the time that the function block is initially configured and loaded. Once the block is loaded, the value of this parameter cannot be changed.

8.2.3 Simatic Digital Input Module Function Block configuration

The proper configuration of the number of channels supported (NUMCHANNSUP) parameter and the assignment of input channel blocks are the only configuration operations that are required.

8.2.4 Simatic Digital Input Module Function Block execution

Input Processing

The Simatic Digital Input Module Function Block executes as an independent block at the base period of the containing CEE.

The remainder of the processing described here only occurs if the function block is in the ACTIVE state.

The following is performed in order …

The status of the communication path to the device is checked

- If the communication path is **not** intact …
  - a notification is generated
  - BADPVSTS for all channels is asserted TRUE indicating ‘Bad' data
- If the communication path is intact …
  - The most recent discrete data is obtained from the associated PBIM Block
  - This input data is assigned to the corresponding function block parameters
  - BADPVSTS is set FALSE for all channels

Status Processing

This block currently provides no device specific or channel specific status and fault data.

The block does note and process communication fault data. Status of communication with the target device is determined in conjunction with the associated PBIM Block. The following conditions will result in a communication fault:

- this Simatic Digital Input Function Block loses communication with its associated PBIM Block
- the associated PBIM Block loses communication with its associated PROFIBUS Interface Module
- the associated PROFIBUS Interface Module losses communication with or detects a major fault on the target device

8.3 The Simatic Digital Output Module Function Block

8.3.1 Simatic Digital Output Module Function Block Functional Description

The Simatic Digital Output Module Function Block provides a configurable number of channels of discrete (binary) output data. The maximum number of channels of discrete data that this module can be configured to provide is 32.

Digital Output Channel Function Block Interface

The Simatic Digital Output Module Function Block supports a set of interfaces to the standard Experion Digital Output Channel Function Block. This interface provides the following:

- a BOOLEAN output value
- a BOOLEAN input value that represents an ‘echo back' value meant to reflect the output value posted to the device.

  (Since there is no echo back value from Simatic Digital Output devices, the Digital Output Module Function Block simply reflects the current output value assigned to the channel back through this signal)
- a BOOLEAN signal indicating the status (GOOD vs. BAD) of the input value. The status is set to BAD when any condition exists that prevents timely update of the input value. Examples include Digital Output Module Function Block state = INACTIVE or a loss of communication to the field device.

8.3.2 Parameters Specific to the Simatic Digital Output Module Function Block

The following parameters are specific to the Simatic Digital Output Module Function Block:

Function	Parameter Name	Description
DOMSO	Output Value	This is an arrayed parameter. Each element of the array represents a single BOOLEAN output value. This value can only be written by a Digital Output Channel Function Block. The absolute maximum size of this array is 32 elements; however, once the function block is configured and loaded, the size of this array is restricted to the value assigned to MAXCHANS.
PVVAL	Process Value	This is an arrayed parameter. Each element of the array represents a single BOOLEAN input value representing an ‘Echo Back' value. This is a READ-ONLY parameter. The absolute maximum size of this array is 32 elements; however, once the function block is configured and loaded, the size of this array is restricted to the value assigned to MAXCHANS.
BADPVSTS	Bad PV Status	This is an arrayed parameter. Each element of the array represents a single BOOLEAN status value that indicates whether or not the corresponding element of the PVVAL array is valid. TRUE = BAD FALSE = GOOD This is a READ-ONLY parameter.

Function	Parameter Name	Description
NUMCHANSSUP	Number of Channels Configured	This parameter defines the number of discrete data channels supported by the Simatic Digital Output device that this function block will represent. It may only have a value of 4, 8, 16, 32 This is an INTEGER parameter that the user provides at the time that the function block is initially configured and loaded. Once the block is loaded, the value of this parameter cannot be changed.

8.3.3 Simatic Digital Output Module Function Block configuration

The proper configuration of the number of channels supported (NUMCHANNSUP) parameter and the assignment of input channel blocks are the only configuration operations that are required.

8.3.4 Simatic Digital Output Module Function Block execution

Input Processing

The Simatic Digital Output Module Function Block executes as an independent block at the base period of the containing CEE.

The remainder of the processing described here only occurs if the function block is in the ACTIVE state.

The following is performed in order …

The status of the communication path to the device is checked

- If the communication path is *not* intact …
  - a notification is generated
  - BADPVSTS for all channels is asserted TRUE indicating ‘Bad' data
- If the communication path is intact …
  - The current value of each output channel is assigned to its corresponding input echo-back channel.
  - BADPVSTS is set FALSE for all channels

NOTE: No real input data is received from the Simatic Digital Output device itself. The Simatic Digital Output device does not support such echoback data.

Status Processing

This block currently provides no device specific or channel specific status and fault data.

The block does note and process communication fault data. Status of communication with the target device is determined in conjunction with the associated PBIM Block. The following conditions will result in a communication fault:

- this Simatic Digital Output Function Block loses communication with its associated PBIM Block
- the associated PBIM Block loses communication with its associated PROFIBUS Interface Module
- the associated PROFIBUS Interface Module losses communication with or detects a major fault on the target device

Output Processing

Following input processing as described above, output values for all channels are written as a single word to the PBIM Block. This function block forwards the data to the target Simatic Output device.

WARNING: In the absence of any true echoback data from the device, back initialization of outputs following a disruption in communication between a Experion controller and a Simatic Digital Output device cannot be

guaranteed to be ‘bumpless'. In the face of a loss of communication to a Simatic Digital Output device, the user should take precautions to ensure safety before re-connecting and activating a Simatic Digital Output device.

8.4 The Simatic Analog Input Module Function Block

8.4.1 Simatic Analog Input Module Function Block Functional Description

The Simatic Analog Input Module Function Block provides a configurable number of channels of analog input data. The maximum number of channels of analog data that this module can be configured to provide is thirty-two (32). The format of this data (signal type and range) is specified during device and module function block configuration.

Analog Input Channel Function Block Interface

The Simatic Analog Input Module Function Block supports an interface to the standard Experion Analog Input Channel Function Block. This interface provides both a FLOAT input value and a BOOLEAN signal indicating the status (GOOD vs. BAD) of the input value. The status is set to BAD when any condition exists that prevents timely update of the input value. Examples include Analog Input Module Function Block state = INACTIVE or a loss of communication to the field device.

8.4.2 Parameters Specific to the Simatic Analog Input Module Function Block

The following parameters are specific to the Simatic Analog Input Module Function Block:

Function	Parameter Name	Description
PVVAL	Process Value	This is an arrayed parameter. Each element of the array represents a single FLOAT input value. This value is the result of processing performed by this module function block upon a corresponding ‘raw' value received from the device. This processing typically converts A/D counts (provided by the device) to a value representing percentage of full range. This value is NaN if communication with the associated Simatic Analog Input device fails this Module Function Block is INACTIVE an overrange or underrange condition is detected This is a READ-ONLY parameter.
BADPVSTS	Bad PV Status	This is an arrayed parameter. Each element of the array represents a single BOOLEAN status value that indicates whether or not the corresponding element of the PVVAL array is valid. TRUE = BAD FALSE = GOOD This is a READ-ONLY parameter.
PVEU	PV in Engineering Units	This parameter represents the present value in engineering units. In the case of resistance and temperature inputs, PV and PVEU are identical.
PVRAW	Raw PV	This is an arrayed parameter. Each element of the array represents a single integer value representing the ‘raw' input value as received from the associated device. Typically, this value represents A/D counts.

Function	Parameter Name	Description
UNDERRANGE	Underrange	This is an arrayed parameter. Each element of the array represents a single BOOLEAN value which is set if the input value drops below the detectable threshold supported by the input device, which is less than, or in some cases = to 0%. Otherwise, this value is FALSE. This is a READ-ONLY parameter.
OVERRANGE	Overrange	This is an arrayed parameter. Each element of the array represents a single BOOLEAN value which is set if the input value goes above the reportable limit supported by the input device, which is greater than, or in some cases = to 100%. Otherwise this value is FALSE. This is a READ-ONLY parameter.
MEASUREMODE	Measurement Mode	This is an arrayed enumeration parameter. Each element of the array contains a single enumeration that specifies a signal type and range. Possible values are cited below.
NUMCHANSSUP	Number of Channels Supported	This parameter defines the number of analog data channels supported by the Simatic Analog Input device that this function block will represent. It may only have a value of 2, 4, 8 This is an INTEGER parameter that the user provides at the time that the function block is initially configured and loaded. Once the block is loaded, the value of this parameter cannot be changed.

8.4.3 Simatic Analog Input Module Function Block Configuration

The Simatic Analog Input Channel block requires that the user select the input sensor type for each of the channels that are in use. For example, if the module in use supports two channels, but only one of those channels is in use, then only the sensor type of the channel in use needs to be configured. Unsupported and unused channels should remain configured with the default value of DEACTIVATED. The following table lists the possible choices for the MEASUREMODE parameter and the corresponding signal type and range.

Enumeration	Signal Type	Range
Deactivated	This turns off the specified channel. Inputs on this channel are not processed. This value should be chosen for all unused channels.	
Neg25_25mV	Voltage	-25 to +25 milliVolts
Neg50_50mV	Voltage	-50 to +50 milliVolts
Neg80_80mV	Voltage	-80 to +80 milliVolts
Neg250_250mV	Voltage	-250 to +250 milliVolts
Neg500_500mV	Voltage	-500 to +500 milliVolts
Neg1_1V	Voltage	-1 to +1 Volts
Neg2p5_2p5V	Voltage	-2.5 to +2.5 Volts
Neg5_5V	Voltage	-5 to +5 Volts
Neg10_10V	Voltage	-10 to +10 Volts
1_5V	Voltage	+1 to +5 Volts
0_10V	Voltage	0 to +10 Volts
Neg3p2_3p2mA	Current	-3.2 to +3.2 mA

Enumeration	Signal Type	Range
Neg10_10mA	Current	-10 to +0 mA
Neg20_20mA	Current	-20 to +20 mA
0_20mA	Current	0 to 20 mA
4_20mA	Current	4 to 20 mA
150ohms	Resistance	150 Ohms
300ohms	Resistance	300 Ohms
600ohms	Resistance	600 Ohms
10kohms	Resistance	10 kOhms
Pt100_std	RTD (Temperature)	Refer to device documentation
Pt100_clim	RTD (Temperature)	Refer to device documentation
Pt200_std	RTD (Temperature)	Refer to device documentation
Pt200_clim	RTD (Temperature)	Refer to device documentation
Ni100_std	RTD (Temperature)	Refer to device documentation
Ni100_clim	RTD (Temperature)	Refer to device documentation
ThermoCpl_B	Thermocouple (Temperature)	Refer to device documentation
ThermoCpl_N	Thermocouple (Temperature)	Refer to device documentation
ThermoCpl_E	Thermocouple (Temperature)	Refer to device documentation
ThermoCpl_R	Thermocouple (Temperature)	Refer to device documentation
ThermoCpl_S	Thermocouple (Temperature)	Refer to device documentation
ThermoCpl_J	Thermocouple (Temperature)	Refer to device documentation
ThermoCpl_L	Thermocouple (Temperature)	Refer to device documentation
ThermoCpl_T	Thermocouple (Temperature)	Refer to device documentation
ThermoCpl_K	Thermocouple (Temperature)	Refer to device documentation
ThermoCpl_U	Thermocouple (Temperature)	Refer to device documentation

8.4.4 Simatic Analog Input Module Function Block execution

Input Processing

The Simatic Analog Input Module Function Block executes as an independent block at the base period of the containing CEE.

The remainder of the processing described here only occurs if the function block is in the ACTIVE state.

The following is performed in order …

The status of the communication path to the device is checked

- If the communication path is *not* intact …
  - a notification is generated
  - BADPVSTS for all channels is asserted TRUE indicating ‘Bad' data
- If the communication path is intact …
  - The most recent analog data is obtained from the associated PBIM Block
  - This input data is assigned to the corresponding function block parameters
  - BADPVSTS is set FALSE for all channels

Status Processing

This block currently provides no device specific or channel specific status and fault data.

The block does note and process communication fault data. Status of communication with the target device is determined in conjunction with the associated PBIM Block. The following conditions will result in a communication fault:

- this Simatic Analog Input Function Block loses communication with its associated PBIM Block
- the associated PBIM Block loses communication with its associated PROFIBUS Interface Module
- the associated PROFIBUS Interface Module losses communication with or detects a major fault on the target device

8.5 The Simatic Analog Output Module Function Block

8.5.1 Simatic Analog Output Module Function Block Functional Description

The Simatic Analog Output Module Function Block provides a configurable number of channels of analog output data. The maximum number of channels of analog data that this module can be configured to provide is 32. The format of this data (signal type and range) is specified during device and module function block configuration.

Analog Output Channel Function Block Interface

The Simatic Analog Output Module Function Block supports an interface to the standard Experion Analog Output Channel Function Block. This interface provides the following:

- A FLOAT output value provided by the Analog Output Channel Function Block to the Simatic Analog Output Module Function Block. This represents the output value that is to be sent to the associated Simatic Analog Output device
- A FLOAT input value provided by this Module Function Block to the Analog Output Channel Function Block. This value, while originating in the Simatic Analog Output Module Function Block, represents the ‘echoback' value from this channel to the higher level control blocks in the system that use this channel.

  NOTE: The Simatic Analog Output device does not provide an ‘echoback' value from the device itself.
- A BOOLEAN signal indicating the status (GOOD vs. BAD) of the echoback value. The status is set to BAD when any condition exists that prevents timely communication with the Simatic Analog Output device. Examples include Analog Output Module Function Block state = INACTIVE or a loss of communication to the field device.

8.5.2 Parameters Specific to the Simatic Analog Output Module Function Block

The following parameters are specific to the Simatic Analog Output Module Function Block:

Function	Parameter Name	Description
OPFINAL	Output Value	This is an arrayed parameter. Each element of the array represents a single FLOAT output value represented as a percentage of full engineering range. This parameter may only be written via the Analog Output Channel Function Block.
OPECHO	Echo Back Value	This is an arrayed parameter. Each element of the array represents a single FLOAT input value represented as a percentage of full engineering range. The last output signal value posted to the associated device is reflected in this parameter. This is a READ-ONLY parameter.
SIGNALTYPE	Signal Type	This is an arrayed enumeration parameter. Each element of the array contains a single enumeration that specifies a signal type and range. Possible values are documented below. The user specifies the value of each element of this parameter at the time that the function block is initially configured and loaded. Once the block is loaded, the value of the elements of this parameter cannot be changed.

Function	Parameter Name	Description
NUMCHANSSUP	Number of Channels Supported	This parameter defines the number of analog data channels supported by the Simatic Analog Output device that this function block will represent. It may only have a value of 2 or 4 This is an INTEGER parameter that the user provides at the time that the function block is initially configured and loaded. Once the block is loaded, the value of this parameter cannot be changed.

8.5.3 Simatic Analog Output Module Function Block Configuration

The Simatic Analog Input Channel block requires that the user select the output type for each of the channels that are in use. For example, if the module in use supports two channels, but only one of those channels is in use, then only the output type of the channel in use needs to be configured. Unsupported and unused channels should remain configured with the default value of DEACTIVATED. The following table lists the possible choices for the SIGNALTYPE parameter and the corresponding signal type and range.

Enumeration	Signal Type	Range
Deactivated	This turns off the specified channel. Outputs on this channel are not processed. This value should be chosen for all unused channels.	
1To5V	Voltage	+1 to +5 Volts
0To10V	Voltage	0 to +10 Volts
Neg10To10V	Voltage	-10 to +10 Volts
0To20mA	Current	0 to 20 mA
4To20mA	Current	4 to 20 mA
Neg20To20mA	Current	-20 to +20 mA

8.5.4 Simatic Analog Output Module Function Block execution

Input Processing

The Simatic Analog Output Module Function Block executes as an independent block at the base period of the containing CEE.

The remainder of the processing described here only occurs if the function block is in the ACTIVE state.

The following is performed in order …

The status of the communication path to the device is checked

- If the communication path is *not* intact …
  - a notification is generated
  - a flag is set indicating that the last output data sent to the device, as reflected in the parameter OPECHO, is BAD
- If the communication path is intact …
  - The current value of each output channel is assigned to its corresponding input echo-back channel.
  - a flag is set indicating that the last output data sent to the device, as reflected in the parameter OPECHO, is GOOD

NOTE: No real input data is received from the Simatic Analog Output device itself. The Simatic Analog Output device does not support such echoback data

Status Processing

This block currently provides no device specific or channel specific status and fault data.

The block does note and process communication fault data. Status of communication with the target device is determined in conjunction with the associated PBIM Block. The following conditions will result in a communication fault:

- this Simatic Analog Output Function Block loses communication with its associated PBIM Block
- the associated PBIM Block loses communication with its associated PROFIBUS Interface Module
- the associated PROFIBUS Interface Module losses communication with or detects a major fault on the target device

Output Processing

Following input processing as described above, output values for all channels are written as an array of words to the PBIM Block. This function block forwards the data to the target Simatic Analog Output device.

WARNING: In the absence of any true echoback data from the device, back initialization of outputs following a disruption in communication between a Experion controller and a Simatic Analog Output device cannot be guaranteed to be ‘bumpless'. In the face of a loss of communication to a Simatic Analog Output device, the user should take precautions to ensure safety before re-connecting and activating a Simatic Analog Output device.

9 PROFIBUS Encoder Function Blocks

The PTO (PROFIBUS Trade Organization) has defined a standard profile for Encoders. The blocks described in this section have been developed based upon this specification and can be used with encoders from various manufacturers that adhere to it.

The encoder function block set consists of the following blocks:

- Encoder Device Block
- Encoder Input Channel Block
- Encoder Output Channel Block

Related topics

“Encoder Device Block” on page 120
“Encoder Input Channel Block” on page 123
“Encoder Output Channel Block” on page 125

9.1 Encoder Device Block

Related topics

“Encoder Device Block description” on page 120
“Encoder Device Block Channel Assignment” on page 120
“Encoder Device Block configuration” on page 120
“Encoder Device Block Configuration Parameters” on page 121
“Encoder Device Block Operation” on page 122

9.1.1 Encoder Device Block description

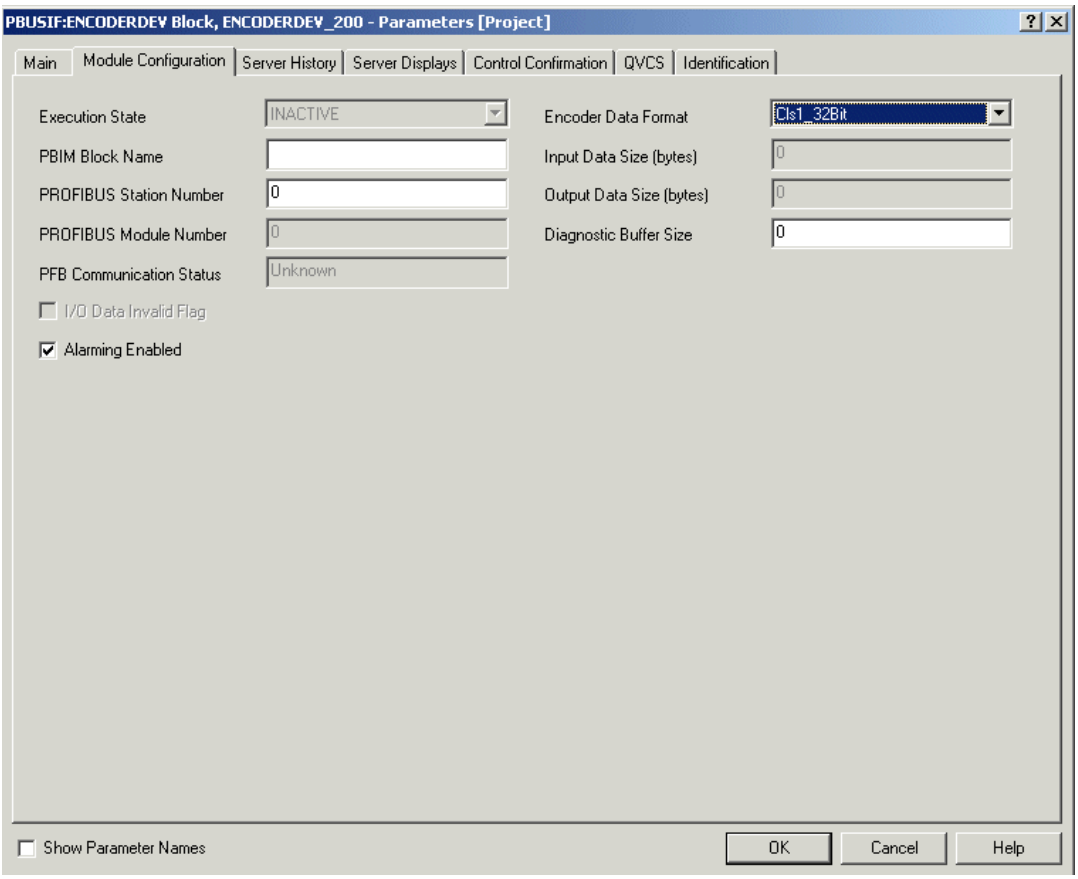
The block template name of the encoder device block is: ENCODERDEV. This block represents the encoder device itself. Process data is made available through input and output channel blocks that are associated with the encoder device block.

9.1.2 Encoder Device Block Channel Assignment

The encoder device block is capable of hosting only two channel blocks. Channel 0 is reserved for an encoder input channel block, and channel 1 is reserved for assignment by an encoder output channel block. The encoder output channel block is not required in all configurations, specifically, where the encoder is configured as an input-only device.

9.1.3 Encoder Device Block configuration

The Module Configuration tab for the encoder device block is pictured below.



Notes on the diagram:

- The 'PBIM Block Name' and 'PROFIBUS Station Number' fields represent the SRCBLOCK and STATION parameters respectively. These parameters, as well as the Diagnostic Buffer Size (DIAGBUFSIZE) parameter, are common to all PROFIBUS device/module blocks and are described in Commonality of PROFIBUS I/O Device/Module Blocks Section.
- Because the Encoder station supports only a single module, the module number is not configurable, and is set to 0 by default.
- The input and output data sizes are not configurable directly. Instead, these parameters are calculated after block load based upon the selection for 'Encoder Data Format', which is described below in the parameter definition for ENCFORMAT.

9.1.4 Encoder Device Block Configuration Parameters

The only device specific configuration parameter for the encoder device block is ENCFORMAT.

Parameter Name:	**ENCFORMAT**		
Specific to Block:	ENCODERDEV		
Description:	Encoder configuration format		
Data Type:	Enumeration of ENCODERFORMAT		
Range:	#	*Enum text*	*Description*
	1	Cls1_16Bit	Input only - 16 bit
	2	Cls1_32Bit	Input only - 32 bit

	3	Cls2_16Bit	Input/output - 16 bit
	4	Cls2_32Bit	Input/output - 32 bit
Default:	0		
Config Load:	Yes		
Access Lock:	Application Developer		
Residence:	CEE		
Related Parameters:			
Remarks:	When using class 1 functionality only, there is no need for the assignment of an output channel block.		

9.1.5 Encoder Device Block Operation

Refer to *Operation* in the *Commonality of PROFIBUS I/O Device/Module Blocks* Section, which describes the operational characteristics common to all PROFIBUS device/module blocks.

The detailed operational characteristics and parameters of the Encoder device are accessed through the encoder input and output channel blocks.

9.2 Encoder Input Channel Block

Refer to the *Commonality of PROFIBUS I/O Channel Blocks* Section for attributes that are common to all PROFIBUS channel blocks.

Related topics

"Encoder Input Channel Block description" on page 123
"Encoder Input Channel Block Configuration" on page 123
"Encoder Input Channel Block Operation" on page 123

9.2.1 Encoder Input Channel Block description

The Encoder Input Channel block template name is: ENCODERIN. This input channel block is responsible for providing the control interface to all input parameters from the encoder device, specifically the position of the encoder device. The ENCODERIN block can be assigned only to channel 0 of the Encoder Device block.

9.2.2 Encoder Input Channel Block Configuration

Besides the channel assignment, block execution order, and any desired change in the parameter or pin exposure on the block, the Encoder Input Channel block requires no specific configuration operations.

9.2.3 Encoder Input Channel Block Operation

The following parameters describe the operational parameters of the encoder function block, specifically the position parameter and associated status.

Parameter Name:	**POSITION**
Specific to Block:	ENCODERIN
Description:	Position This parameter represents the floating-point position value (PV), as received from the encoder device.
Data Type:	FLOAT64
Range:	
Default:	0.0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	POSSTS, POSITIONRAW
Remarks:	The POSITION connection is 'intelligent' in that it will automatically pass value and status to any block that accepts status information, such as the P1 connection of a DATAACQ block or the PV connection of a PID block.

Parameter Name:	**POSITIONRAW**
Specific to Block:	ENCODERIN
Description:	Position Raw This parameter represents encoder position as an unsigned 32 bit integer, as received from the encoder device.

Data Type:	UINT32
Range:	
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	POSSTS, POSITION, ENCODEROUTCHAN.FEEDBACKVAL
Remarks:	This parameter is typically used in connecting to the FEEDBACKVAL connection of the encoder output channel block, for use in automatic reset of the preset enable.

Parameter Name:	**POSSTS**		
Specific to Block:	ENCODERIN		
Description:	Position Status		
Data Type:	Enumeration of PVSTS		
Range:	0	Bad	
	1	Uncertain	(not used by this block)
	2	Normal	
	3	Manual	(not used by this block)
	4	InitAck	(not used by this block)
Default:	Bad		
Config Load:	No		
Access Lock:	View Only		
Residence:	CEE		
Related Parameters:	POSITON, COMFAILFL		
Remarks:	POSSTS assumes the BAD state if BADINPUTSTS is asserted		

Parameter Name:	**ENCFORMAT**		
Specific to Block:	ENCODERIN		
Description:	Encoder configuration format		
Data Type:	Enumeration of ENCODERFORMAT		
Range:	*#*	*Enum text*	*Description*
	1	Cls1_16Bit	Input only - 16 bit
	2	Cls1_32Bit	Input only - 32 bit
	3	Cls2_16Bit	Input/output - 16 bit
	4	Cls2_32Bit	Input/output - 32 bit
Default:	0		
Config Load:	Yes		
Access Lock:	View Only		
Residence:	CEE		
Related Parameters:			
Remarks:	This view only parameter represents the configuration format of the associated Encoder device/module block.		

9.3 Encoder Output Channel Block

Refer to the *Commonality of PROFIBUS I/O Channel Blocks* Section for attributes that are common to all PROFIBUS channel blocks.

Related topics

“Encoder Output Channel Block description” on page 125
“Encoder Output Channel Block Configuration” on page 125
“Encoder Output Channel Block Operation” on page 125

9.3.1 Encoder Output Channel Block description

The Encoder Output Channel block template name is: ENCODEROUT. This output channel block is responsible for providing the control interface to all input parameters from the encoder device. The ENCODERIN block can be assigned only to channel 1 of the Encoder Device block. This block is not required when the encoder device is configured as an input-only device (Class 1). The BADCONFIGFL parameter will remain asserted if an output channel is loaded, active and assigned to an encoder device that is not configured as a Class 2 device.

The function of the encoder output channel block is to provide a means by which to reset the encoder to a known input position value.

9.3.2 Encoder Output Channel Block Configuration

The output channel block is equipped with an input connection (FEEDBACKVAL), which is intended to be connected to the POSITIONRAW of the associated encoder input channel. The purpose of this connection is to automatically reset the Boolean 'Preset Enable' command when the input value received from the encoder input channel equals the preset value which has been sent by the output channel. If this connection is not made, the operator or logic that is controlling the operation of the encoder output channel is responsible for resetting the Preset Enable. The configuration strategy should take into consideration the function of the PRESETENB parameter, and supporting FEEDBACKVAL connection, in the event that supporting logic, or sequencing is required.

Besides the channel assignment, block execution order, and any desired change in the parameter or pin exposure on the block, the Encoder Output Channel block requires no specific configuration operations.

9.3.3 Encoder Output Channel Block Operation

The following parameters describe the operational parameters of the encoder function block, specifically the position parameter and associated status. Although position is communicated as an integer via PROFIBUS, it is represented as a floating-point value to the CEE.

Parameter Name:	**PRESETVAL**
Specific to Block:	ENCODEROUT
Description:	Preset Value This parameter represents the floating point Preset Value, which is used as the starting value for POSITION.
Data Type:	INT32
Range:	
Default:	0
Config Load:	No

Access Lock:	Operator
Residence:	CEE
Related Parameters:	PRESETENB, ENCODERIN.POSITION
Remarks:	The POSITION connection is ‘intelligent' in that it will automatically pass value and status to any block that accepts status information, such as the P1 connection of a DATAACQ block or the PV connection of a PID block.

Parameter Name:	**PRESETENB**		
Specific to Block:	ENCODEROUT		
Description:	Preset Enable When asserted, instructs the encoder device to accept the PRESETVAL as the current value for POSITION. Preset Enable should be asserted in a momentary fashion, and only when the encoder is stopped.		
Data Type:	BOOLEAN		
Range:	0	Off	Disable Preset
	1	On	Enable Preset
Default:	Off		
Config Load:	No		
Access Lock:	Engineer		
Residence:	CEE		
Related Parameters:	PRESETVAL, ENCODERIN.POSITION		
Remarks:	PRESETENB should be reset to Off when the POSITION value (of the input channel) = PRESETVAL. This reset occurs automatically if the FEEDBACKVAL is connected to the POSITONRAW of the input channel. Otherwise, PRESETENB does not reset automatically.		

Parameter Name:	**FEEDBACKVAL**
Specific to Block:	ENCODEROUT
Description:	Feedback position value This parameter accepts connection to the POSITIONRAW parameter of the encoder input channel, and is used for automatic reset of the PRESETENB.
Data Type:	UINT32
Range:	
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	PRESETVAL, PRESETENB, ENCODERINCHAN.POSITIONRAW
Remarks:	

Parameter Name:	**ENCFORMAT**
Specific to Block:	ENCODEROUT
Description:	Encoder configuration format
Data Type:	Enumeration of ENCODERFORMAT

Range:	#	*Enum text*	*Description*
	1	Cls1_16Bit	Input only - 16 bit
	2	Cls1_32Bit	Input only - 32 bit
	3	Cls2_16Bit	Input/output - 16 bit
	4	Cls2_32Bit	Input/output - 32 bit
Default:	0		
Config Load:	Yes		
Access Lock:	View Only		
Residence:	CEE		
Related Parameters:			
Remarks:	This view only parameter represents the configuration format of the associated Encoder device/module block.		

10 PROFIDRIVE Function Blocks

Related topics

“PROFIDRIVE Function Blocks overview” on page 130
“PROFIDRIVE Device Block” on page 131
“PROFIDRIVE Input Channel Block” on page 135
“Output Channel Block” on page 146

10.1 PROFIDRIVE Function Blocks overview

The PTO (PROFIBUS Trade Organization) has defined a standard PROFIDRIVE profile for motor drives and frequency converters. These are motor control devices allowing monitoring and control of lots of parameters (e.g. speed, torque, current, temperature, etc) required for industrial motor applications. The blocks described in this section have been developed based upon this specification and can be used with drives and converters from various manufacturers that adhere to it.

Attention

Some of the parameters and Control Builder forms that are referenced in this section may have been modified after the publication of this document.

10.1.1 Template Names

The block template names for the PROFIDRIVE blocks are the following:

- ProfiDriveDev - PROFIDRIVE Device Block
- ProfiDriveIn - PROFIDRIVE Input Channel Block
- ProfiDriveOut - PROFIDRIVE Output Channel Block

10.2 PROFIDRIVE Device Block

Related topics

10.2.1 PROFIDRIVE Device Block Description

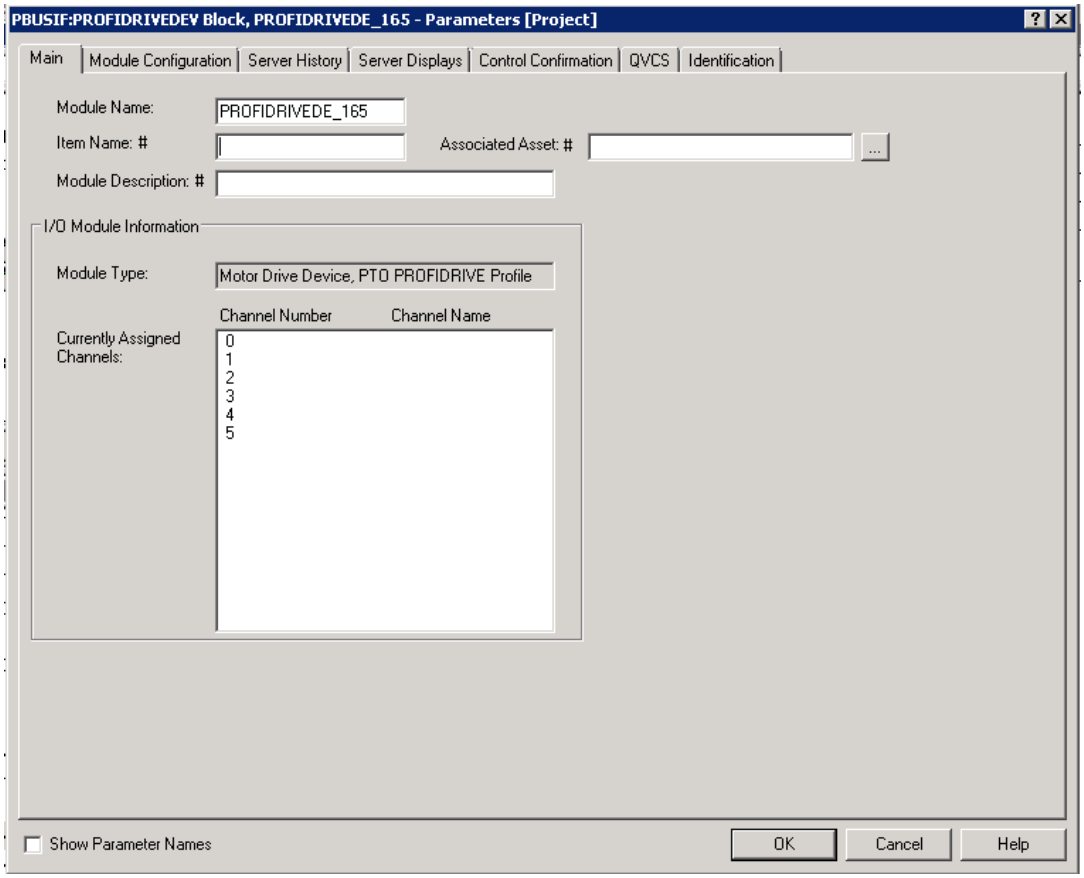
The block template name of the PROFIDRIVE device block is: PROFIDRIVEDEV. This block represents the PROFIDRIVE device itself. Process data is made available through input and output channel blocks that are associated with the PROFIDRIVE device block.

10.2.2 PROFIDRIVE Device Block Channel Assignment

The PROFIDRIVE device block is capable of hosting only two channel blocks. Channel 0 is reserved for assignment of the PROFIDRIVE input channel block, and channel 1 is reserved for assignment of the PROFIDRIVE output channel block.

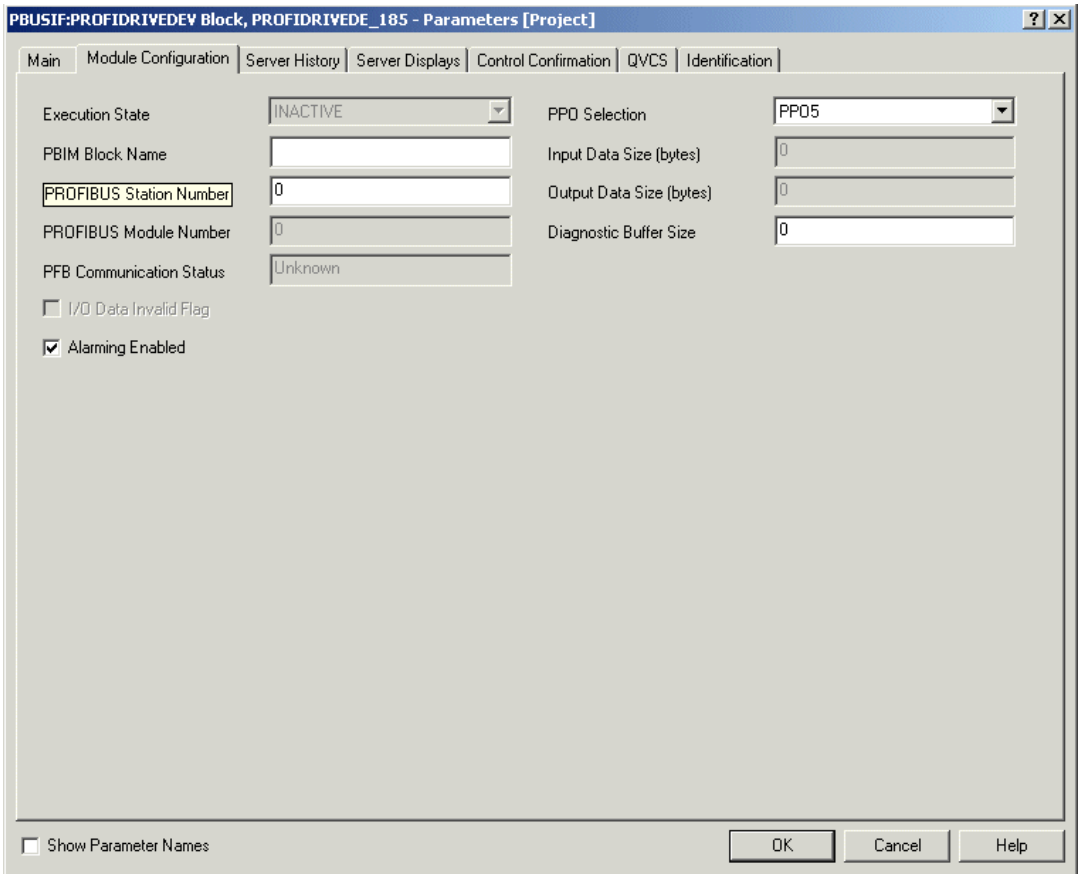
10.2.3 PROFIDRIVE Device Block Configuration

The Main tab for the PROFIDRIVE device block is pictured below.



The Associated Asset field represents an asset selected from those configured in the Enterprise Model Database to set the Scope of Responsibility (SOR) for the point.

The Module Configuration tab for the PROFIDRIVE device block is pictured below.



Notes on the diagram:

- The 'PBIM Block Name' and 'PROFIBUS&Station Number' fields represent the SRCBLOCK and STATION parameters respectively. These parameters, as well as the Diagnostic Buffer Size (DIAGBUFSIZE) parameter, are common to all PROFIBUS device/module blocks and are described in *Commonality of PROFIBUS I/O Device/Module Blocks* Section.
- Because the PROFIDRIVE station supports only a single module, the module number is not configurable, and is set to 0 by default.
- The input and output data sizes are not configurable directly. Instead, these parameters are calculated after block load based upon 'PPO Selection', which is described below in the parameter definition for PPOTYPE. PPO is the acronym for Process Parameter Object, which describes the format and size of the input/output data message, consisting of parameter block and process data or process data only.

10.2.4 PROFIDRIVE Device Block Configuration Parameters

The only device specific configuration parameter for the PROFIDRIVE device block is PPOTYPE.

Parameter Name:	**PPOTYPE**		
Specific to Block:	PROFIDRIVEIN, PROFIDRIVEOUT		
Description:	Format and size of input/output data message		
Data Type:	Enumeration of PROFIDRIVEPPO		
Range:	#	*Enum text*	*Description*
	1	PPO1	Parameter block, Status word, 1 Process data element

	2	PPO2	Parameter block, Status word, 5 Process data elements
	3	PPO3	Status word, 1 Process data element
	4	PPO4	Status word, 5 Process data elements
	5	PPO5	Parameter block, Status word, 9 Process data elements
Default:	PPO5		
Config Load:	Yes		
Access Lock:	Application Developer		
Residence:	CEE		
Related Parameters:			
Remarks:			

The parameter PPOTYPE applies to the input as well as the output channel block. Both channel blocks access this parameter while communicating with the device block. Since PPOTYPE determines contents and structure of the input and output data message, the format of both messages is identical. The same number of bytes that are read from the PROFIDRIVE device (current input data) are also sent to the device (current output data). Furthermore, if e.g. the speed parameter input value is read as process data element 2 of the input message, the speed setpoint must also be send to the device as process data element 2 of the output message.

10.2.5 PROFIDRIVE Device Block Operation

Refer to *Operation* in the *Commonality of PROFIBUS I/O Device/Module Blocks* Section, which describes the operational characteristics common to all PROFIBUS device/module blocks.

The detailed operational characteristics and parameters of the PROFIDRIVE device are accessed through the PROFIDRIVE input and output channel blocks.

10.3 PROFIDRIVE Input Channel Block

Refer to the Commonality of PROFIBUS I/O Channel Blocks Section for attributes that are common to all PROFIBUS channel blocks.

Related topics

“PROFIDRIVE Input Channel Block Description” on page 135
“PROFIDRIVE Input Channel Block Configuration” on page 135
“PROFIDRIVE Input Channel Block Status Word tab” on page 136
“Configuration Parameters for the Status Word Tab” on page 136
“Parameter/Process Data Project tab” on page 137
“PROFIDRIVE Input Channel Block Configuration Parameters for the Parameter/Process Data tab” on page 138
“PROFIDRIVE Input Channel Block Scaling Process Data” on page 138
“Configuration Parameters for the Input Scaling tab” on page 139
“PROFIDRIVE Input Channel Block Operation” on page 141
“Status Word Monitoring tab” on page 141
“Operational Parameters for the Status Word tab” on page 141
“Parameter/Process Data Monitoring tab” on page 142
“Operational Parameters for the Parameters/Process Data Monitoring tab” on page 143

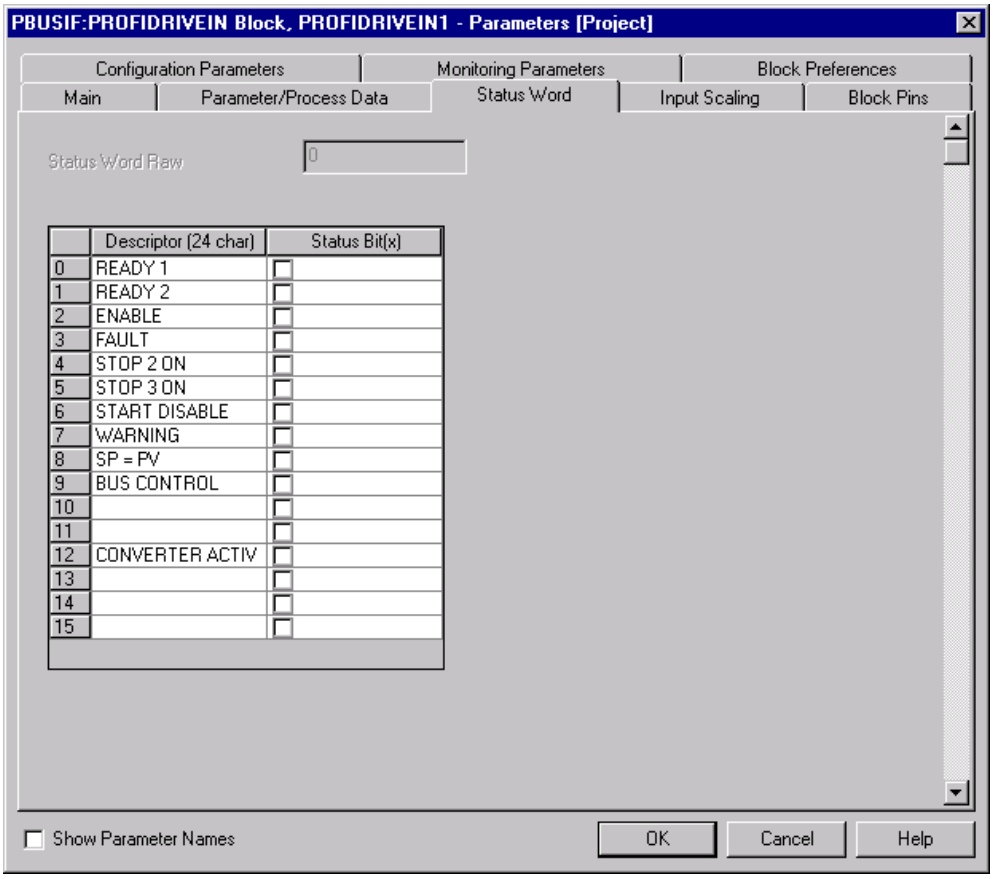
10.3.1 PROFIDRIVE Input Channel Block Description

The PROFIDRIVE Input Channel block template name is: PROFIDRIVEIN. This input channel block is responsible for providing the control interface to all input parameters from the PROFIDRIVE device. The PROFIDRIVEIN block can be assigned only to channel 0 of the PROFIDRIVE Device block.

10.3.2 PROFIDRIVE Input Channel Block Configuration

Besides the channel assignment, block execution order, and any desired change in the parameter or pin exposure on the block, the PROFIDRIVE Input Channel block requires configuration on the Status Word tab, the Parameter/ Process Data tab and the Input Scaling tab.

10.3.3 PROFIDRIVE Input Channel Block Status Word tab



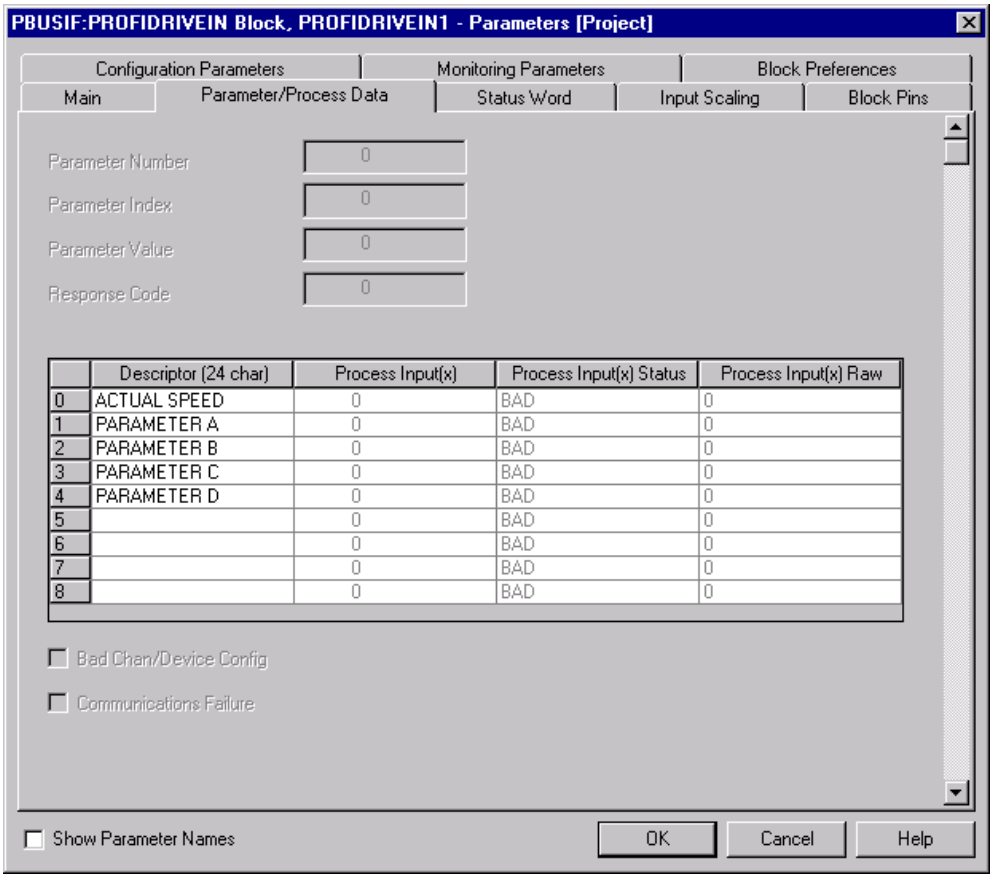
This tab displays the Status Word as a 16bit unsigned integer raw value parameter (SWRAW), the corresponding 16 Status Bits as Boolean parameters (SWBIT[x]) and their corresponding descriptors as 24char. string parameters (SWDESC[x]). The purpose of these descriptors is to identify the use of individual Status Bits, since the SWBIT[x] parameters cannot be renamed. Only the descriptors are configurable on this tab.

10.3.4 Configuration Parameters for the Status Word Tab

Parameter Name:	**SWDESC[0..15]**
Specific to Block:	PROFIDRIVEIN
Description:	Descriptor of SWBIT[x]
Data Type:	24 Character Text String
Range:	N/A
Default:	
Config Load:	Yes
Access Lock:	No access permitted
Residence:	SR
Related Parameters:	SWBIT[..]

Remarks:	Because the SWBIT[x] parameters cannot be renamed, this text descriptor is useful in documenting the use of each SWBIT[x].

10.3.5 Parameter/Process Data Project tab



The upper part of this tab displays the Parameter Data (PARNUM, PARIDX, PARVAL, RESPCODE) along with their specified use as defined by the PROFIDRIVE profile. The lower part displays the Process Data items (PD[x], PDSTS[x], PDRAW[x]) and their corresponding descriptors as 24char. string parameters (PDDESC[x]). The purpose of these descriptors is to identify the use of individual Process Data items, since the parameters PD[x], PDSTS[x] and PDRAW[x] cannot be renamed. Because these Process Data items are selectable and vary for different applications and devices it is required to present them in this array like fashion. Only the descriptors are configurable on this tab.

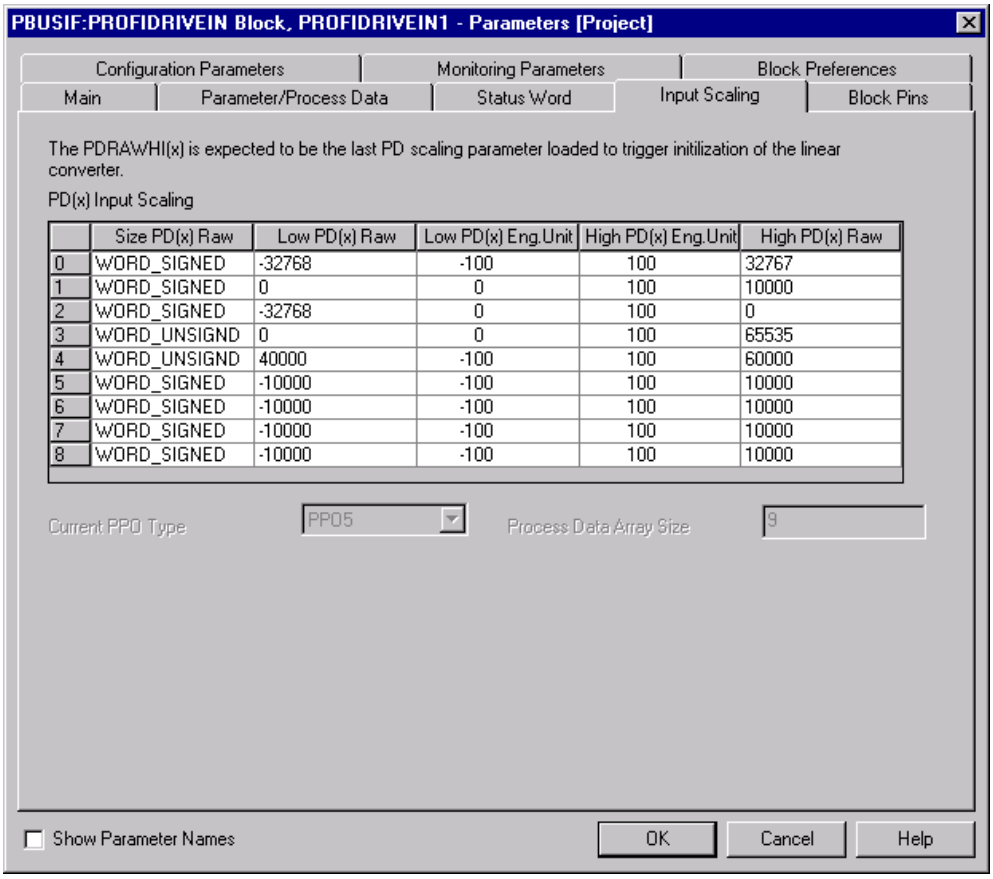
For configuration purposes this tab exposes the maximum number of elements, independent of the currently selected PPO type. Only after activation has the Input Channel block knowledge of the PPO type and will expose only the relevant data.

The descriptors are not repeated on the process data scaling tab, but are related to the scaling parameters as well as to the process data parameters.

10.3.6 PROFIDRIVE Input Channel Block Configuration Parameters for the Parameter/Process Data tab

Parameter Name:	**PDDESC[0..8]**
Specific to Block:	PROFIDRIVEIN
Description:	Descriptor of PDBIT[x]
Data Type:	24 Character Text String
Range:	N/A
Default:	
Config Load:	Yes
Access Lock:	No access permitted
Residence:	SR
Related Parameters:	PD[..], PDSTS[..], PDRAW[..], PDRAWSIZE[..],PDRAWLO[..], PDRAWHI[..], PDEULO[..], PDEUHI[..]
Remarks:	Because the PD[x] parameters cannot be renamed, this text descriptor is useful in documenting the use of each PD[x].

10.3.7 PROFIDRIVE Input Channel Block Scaling Process Data



This tab displays the size of the raw data received from the device and the scaling parameters for Process Data. For configuration purposes this tab exposes the maximum number of elements, independent of the currently selected PPO type. Only after activation has the Input Channel block knowledge of the PPO type and will expose only the relevant data.

The raw data size parameter (PDRAWSIZE[x]) allows for selecting one of the following types:

Data Type	Description	Size (bytes)	Minimum Value	Maximum Value
WORD_SIGNED	Signed 16 bit integer	2	-32,767	32,768
WORD_UNSIGND	Unsigned 16 bit integer	2	0	65,536

Upon entry, the scaling parameters PDRAWLO[x], PDRAWHI[x] are limit checked against PDRAWSIZE[x]. Along with PDEULO[x], PDEUHI[x] these 4 parameters will normally be obtained from technical documentation for the specific PROFIDRIVE device. PDRAWHI[x] is expected to be the last scaling parameter loaded to trigger the initialization of the linear converter.

10.3.8 Configuration Parameters for the Input Scaling tab

Parameter Name:	**PDRAWSIZE[0..8]**				
Specific to Block:	PROFIDRIVEIN				
Description:	Raw integer data size and format				
Data Type:	Enumeration of ProfiDrive_DataSize				
Range:	The additional columns below indicate the physical size of each integer format and the associated minimum and maximum values for use in configuring the raw integer scaling parameters.				
	#	*Enum text*	*Size in bytes*	*Min. Raw Value*	*Max. Raw Value*
	0	WORD_SIGNED	2	-32,767	32,768
	1	WORD_UNSIGND	2	0	65,536
Default:	WORD_SIGNED				
Config Load:	Yes				
Access Lock:	Application Developer				
Residence:	CEE				
Related Parameters:	PDRAWLO [..], PDRAWHI [..], PD[..]				
Remarks:	PDRAWSIZE [x] should be entered prior to PDRAWLO [x], PDRAWHI [x] to allow for limit checking and clamping of these range parameters.				

Parameter Name:	**PDRAWLO[0..8]**
Specific to Block:	PROFIDRIVEIN
Description:	Lower range limit of the integer input value.
Data Type:	INT32
Range:	See PDRAWSIZE[..], upon which this parameter is dependent.
Default:	-10,000
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	PDRAWSIZE[..], PDEULO[..], PD[..]

Remarks:	Upon entry PDRAWLO [x] will be limit checked against PDRAWSIZE [x]

Parameter Name:	**PDRAWHI[0..8]**
Specific to Block:	PROFIDRIVEIN
Description:	Upper range limit of the integer input value.
Data Type:	INT32
Range:	See PDRAWSIZE[..], upon which this parameter is dependent.
Default:	10,000
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	PDRAWSIZE[..], PDEUHI[..], PD[..]
Remarks:	Upon entry PDRAWHI [x] will be limit checked against PDRAWSIZE [x] PDRAWHI[x] is expected to be the last scaling parameter loaded to trigger the initialization of the linear converter.

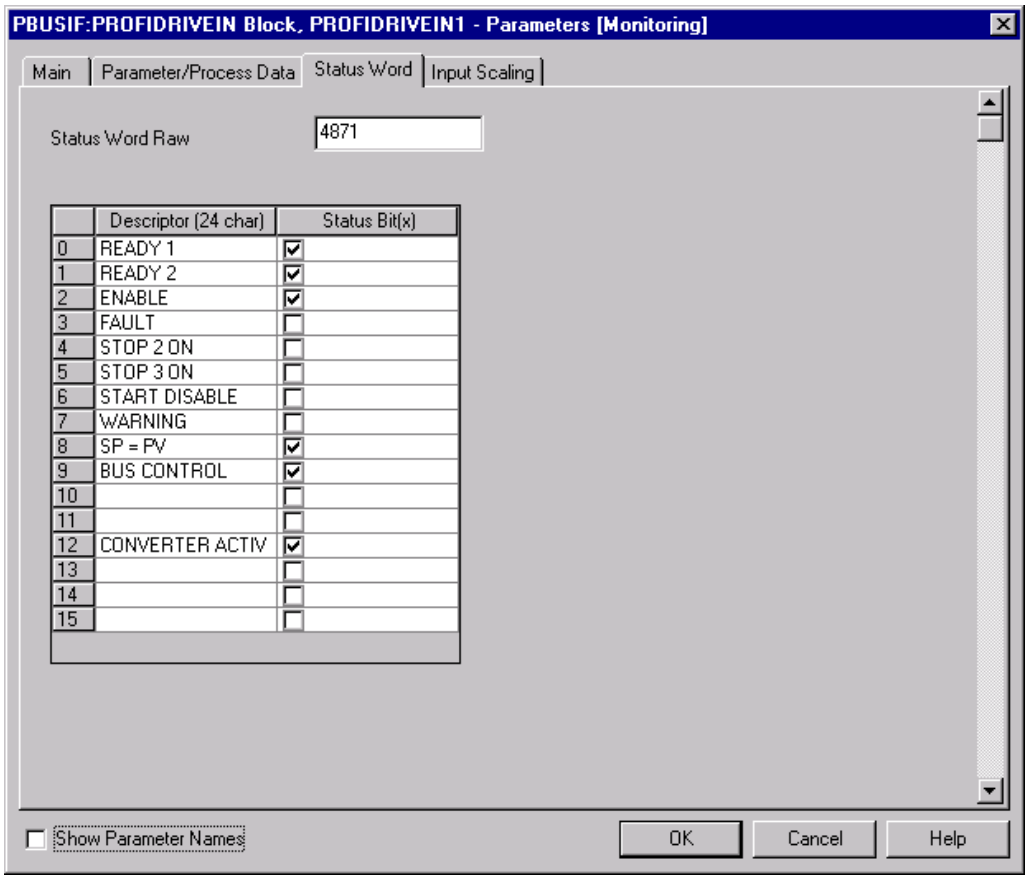
Parameter Name:	**PDEULO[0..8]**
Specific to Block:	PROFIDRIVEIN
Description:	Engineering Unit Low This parameter represents the numeric equivalent of PDRAWLO[x].
Data Type:	FLOAT64
Range:	
Default:	-100.0
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	PDRAWLO[..], PDEUHI[..], PD[..]
Remarks:	Although this parameter is usually less than PDEUHI[x] (representing the lowest numerical input value), if the inputs are scaled in a reverse fashion, this parameter can also be greater than PDEUHI[x] (thus representing the highest numerical input value).

Parameter Name:	**PDEUHI[0..8]**
Specific to Block:	PROFIDRIVEIN
Description:	Engineering Unit High This parameter represents the numeric equivalent of PDRAWHI[x].
Data Type:	FLOAT64
Range:	
Default:	100.0
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	PDRAWHI[..], PDEULO[..], PD[..]

Remarks:	Although this parameter is usually greater than PDEULO[x] (representing the highest numerical input value), if the inputs are scaled in a reverse fashion, this parameter can also be less than PDEULO[x] (thus representing the lowest numerical input value).

10.3.9 PROFIDRIVE Input Channel Block Operation

10.3.10 Status Word Monitoring tab



This monitoring tab displays a live Status Word as a 16bit unsigned integer raw value parameter (SWRAW) and the corresponding 16 Status Bits as Boolean parameters (SWBIT[x]). The corresponding descriptors (SWDESC[x]) have been entered in project mode.

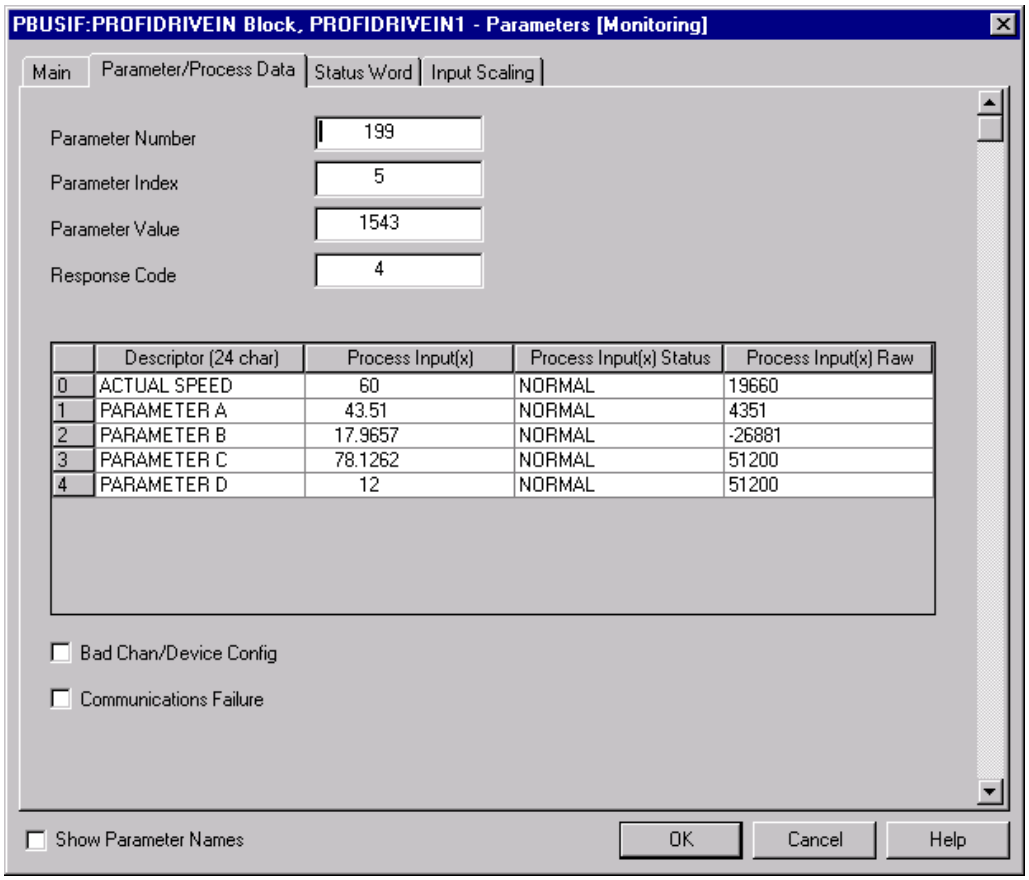
10.3.11 Operational Parameters for the Status Word tab

Parameter Name:	**SWRAW**
Specific to Block:	PROFIDRIVEIN
Description:	Status Word
Data Type:	INT32
Range:	0....65535
Default:	0
Config Load:	No

Access Lock:	View Only
Residence:	CEE
Related Parameters:	SWBIT[..]
Remarks:	Parameter SWRAW is the 16bit status received from the device.

Parameter Name:	**SWBIT[0..15]**
Specific to Block:	PROFIDRIVEIN
Description:	Status Word Bit
Data Type:	BOOLEAN
Range:	OFF(0), ON(1)
Default:	OFF
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	SWRAW
Remarks:	Boolean parameter SWBIT[x] corresponds to bit position 'x' of SWRAW.

10.3.12 Parameter/Process Data Monitoring tab



This parameter/process data monitoring tab displays live data typical for the PPO2 format selection: parameter data and 5 process data elements.

The upper part displays the parameters PARNUM, PARIDX, PARVAL, RESPCODE. Parameter 199[5] has currently a value of 1543, and the request completed successfully with response code 4. A device specific user manual is required to interpret the data. Note that all parameters are unscaled raw data.

The lower part displays 5 process data elements PD[x], PDSTS[x] and PDRAW[x]. The status for all data items is normal. PD[x] is the scaled equivalent of PDRAW[x] according to the scaling parameters entered on the Input Scaling tab in project mode.

The corresponding descriptors PDDESC[x] have been entered in project mode.

10.3.13 Operational Parameters for the Parameters/Process Data Monitoring tab

Parameter Name:	**PARNUM**
Specific to Block:	PROFIDRIVEIN
Description:	Parameter Number
Data Type:	INT32
Range:	0....2047
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	PARIDX, PARVAL, RESPCODE
Remarks:	Parameter PARNUM is the echo of the Parameter Number currently in use by the parameter request.

Parameter Name:	**PARIDX**
Specific to Block:	PROFIDRIVEIN
Description:	Parameter Number Index
Data Type:	INT32
Range:	0....65535
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	PARNUM, PARVAL, RESPCODE
Remarks:	Parameter PARIDX is the echo of the Parameter Number Index currently in use by the parameter request.

Parameter Name:	**PARVAL**
Specific to Block:	PROFIDRIVEIN
Description:	Parameter Value
Data Type:	FLOAT64
Range:	0....4 294 967 295 (UINT32 range)
Default:	0
Config Load:	No
Access Lock:	View Only

Residence:	CEE
Related Parameters:	PARNUM, PARIDX, RESPCODE
Remarks:	Parameter PARVAL is the actual value of PARNUM[PARIDX] currently in use by the parameter request. If the parameter RESPCODE returns a request failure, PARVAL holds the error code.

Parameter Name:	**RESPCODE**
Specific to Block:	PROFIDRIVEIN
Description:	Parameter Request Response Code
Data Type:	INT32
Range:	0....15
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	PARNUM, PARIDX, PARVAL
Remarks:	Parameter RESPCODE is the status of the current parameter request, indicating success or failure. If RESPCODE returns a request failure, parameter PARVAL holds the error code.

Parameter Name:	**PD[0..8]**
Specific to Block:	PROFIDRIVEIN
Description:	Process Input[x], Scaled This parameter is the scaled floating point representation of the raw integer input value.
Data Type:	FLOAT64
Range:	
Default:	0.0
Config Load:	No
Access Lock:	View Only
Residence:	CEE
Related Parameters:	PDDESC[..], PDRAW[..], PDSTS[..], PDRAWSIZE[..], PDRAWLO[..], PDRAWHI[..], PDEULO[..], PDEUHI[..]
Remarks:	The PV connection is ‘intelligent' in that it will automatically pass value and status to any block that accepts status information, such as the P1 connection of a DATAACQ block or the PV connection of a PID block.

Parameter Name:	**PDRAW[0..8]**
Specific to Block:	PROFIDRIVEIN
Description:	Process Input[x] raw value This value is the raw field value that is transported via PROFIBUS. It is scaled by the block to calculate PD[x].
Data Type:	INT32
Range:	Varies based on PDRAWSIZE[..]
Default:	0
Config Load:	No

Access Lock:	View Only
Residence:	CEE
Related Parameters:	PD[..]
Remarks:	

Parameter Name:	**PDSTS[0..8]**		
Specific to Block:	PROFIDRIVEIN		
Description:	Process Input[x] Status		
Data Type:	Enumeration of PVSTS		
Range:	0	Bad	
	1	Uncertain	(not used by this block)
	2	Normal	
	3	Manual	(not used by this block)
	4	InitAck	(not used by this block)
Default:	Bad		
Config Load:	No		
Access Lock:	View Only		
Residence:	CEE		
Related Parameters:	COMFAILFL, PD[..]		
Remarks:	PDSTS[x] assumes the BAD state if COMMFAILFL is asserted.		

10.4 Output Channel Block

Refer to the *Commonality of PROFIBUS I/O Channel Blocks* Section for attributes that are common to all PROFIBUS channel blocks.

Related topics

“Output Channel Block description” on page 146
“Output Channel Block Configuration” on page 146
“Control Word Project tab” on page 147
“Configuration Parameters for the Control Word tab” on page 147
“Output Channel Block Parameter/Process Data project tab” on page 148
“Output Channel Block Configuration Parameters for the Parameter/Process Data tab” on page 148
“Output Channel Block Scaling Process Data” on page 149
“Configuration Parameters for the Output Scaling tab” on page 150
“Output Channel Block Operation” on page 151
“Output Channel Block Control Word Monitoring tab” on page 152
“Operational Parameters for the Control Word tab” on page 152
“Output Channel Block Parameter/Process Data Monitoring tab” on page 153
“Output Channel Block Operational Parameters for the Parameter/Process Data Monitoring tab” on page 154

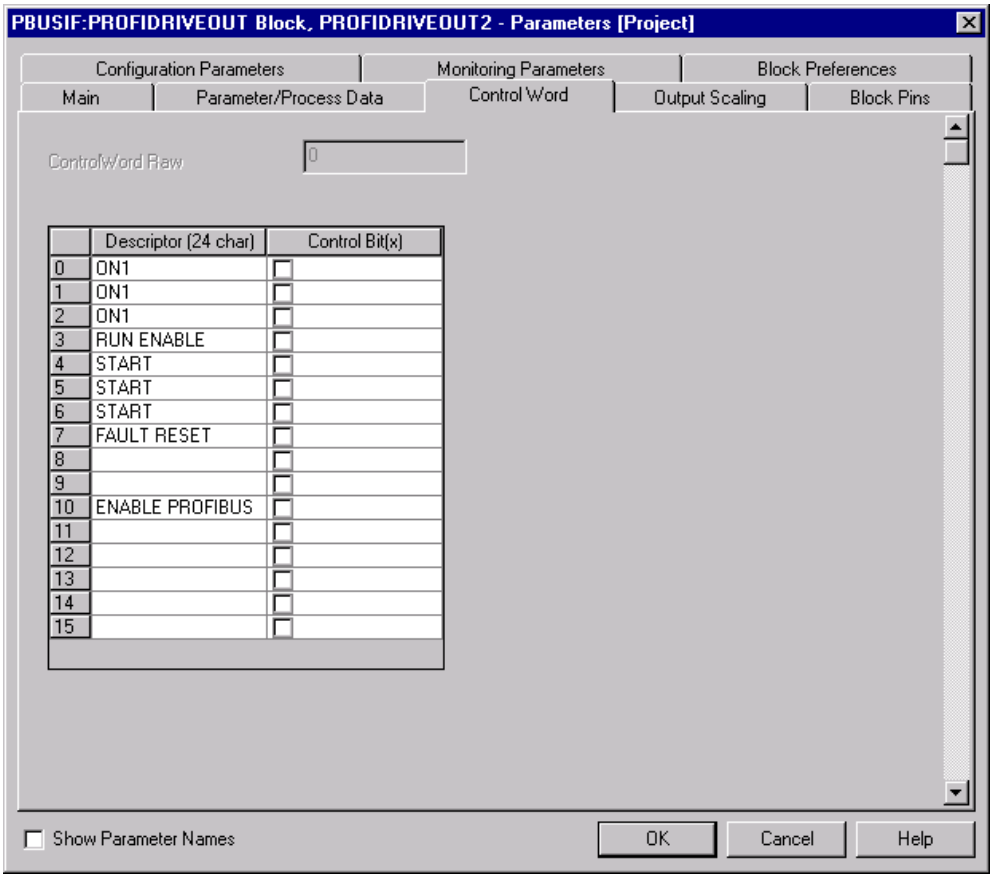
10.4.1 Output Channel Block description

The PROFIDRIVE Output Channel block template name is: PROFIDRIVEOUT. This output channel block is responsible for providing the control interface to all output parameters to the PROFIDRIVE device. The PROFIDRIVEOUT block can be assigned only to channel 1 of the PROFIDRIVE Device block.

10.4.2 Output Channel Block Configuration

Besides the channel assignment, block execution order, and any desired change in the parameter or pin exposure on the block, the PROFIDRIVE Output Channel block requires configuration on the Control Word tab, the Parameter/ Process Data tab and the Output Scaling tab.

10.4.3 Control Word Project tab



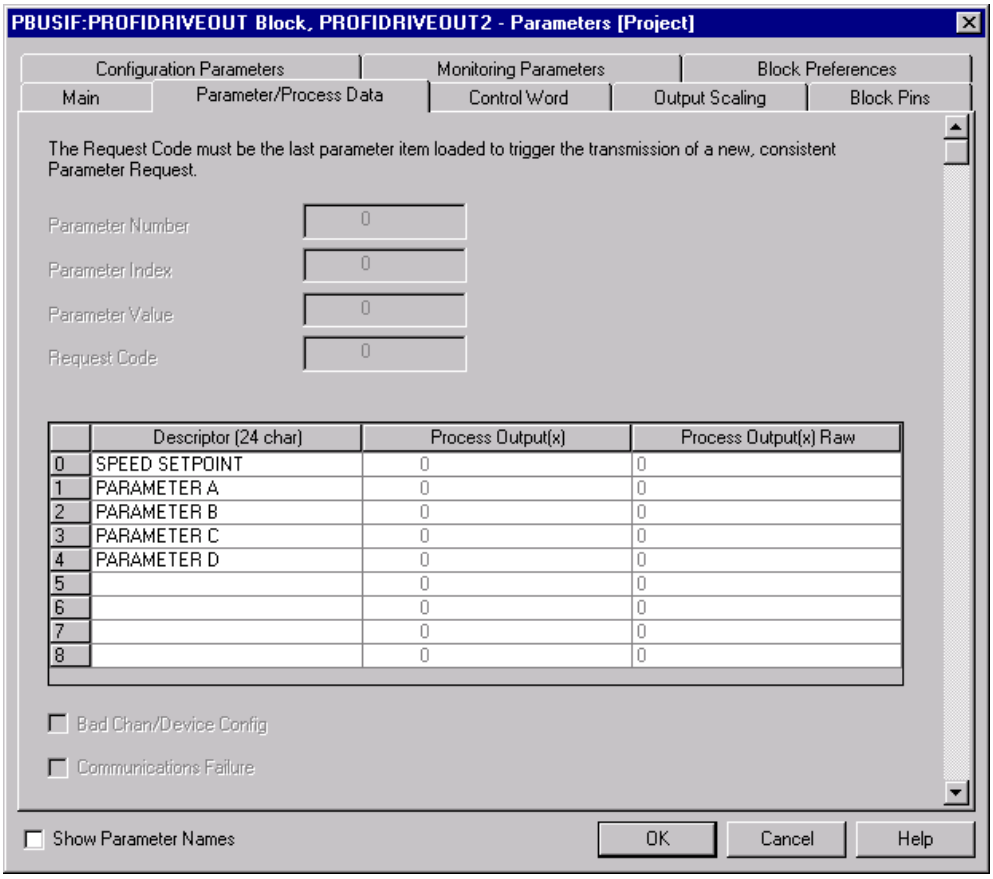
This tab displays the Control Word as a 16bit unsigned integer raw value parameter (CWRAW), the corresponding 16 Control Bits as Boolean parameters (CWBIT[x]) and their corresponding descriptors as 24char. string parameters (CWDESC[x]). The purpose of these descriptors is to identify the use of individual Control Bits, since the CWBIT[x] parameters cannot be renamed. Only the descriptors are configurable on this tab.

10.4.4 Configuration Parameters for the Control Word tab

Parameter Name:	**CWDESC[0..15]**
Specific to Block:	PROFIDRIVEOUT
Description:	Descriptor of CWBIT[x]
Data Type:	24 Character Text String
Range:	N/A
Default:	
Config Load:	Yes
Access Lock:	No access permitted
Residence:	SR
Related Parameters:	CWBIT[..]

Remarks:	Because the CWBIT[x] parameters cannot be renamed, this text descriptor is useful in documenting the use of each CWBIT[x].

10.4.5 Output Channel Block Parameter/Process Data project tab



The upper part of this tab displays the Parameter Data (PARNUM, PARIDX, PARVAL, REQUCODE) along with their specified use as defined by the PROFIDRIVE profile. The lower part displays the Process Data items (PD[x], PDRAW[x]) and their corresponding descriptors as 24char. string parameters (PDDESC[x]). The purpose of these descriptors is to identify the use of individual Process Data items, since the parameters PD[x], PDSTS[x] and PDRAW[x] cannot be renamed. Because these Process Data items are selectable and vary for different applications and devices it is required to present them in this array like fashion. Only the descriptors are configurable on this tab.

For configuration purposes this tab exposes the maximum number of elements, independent of the currently selected PPO type. Only after activation has the Input Channel block knowledge of the PPO type and will expose only the relevant data.

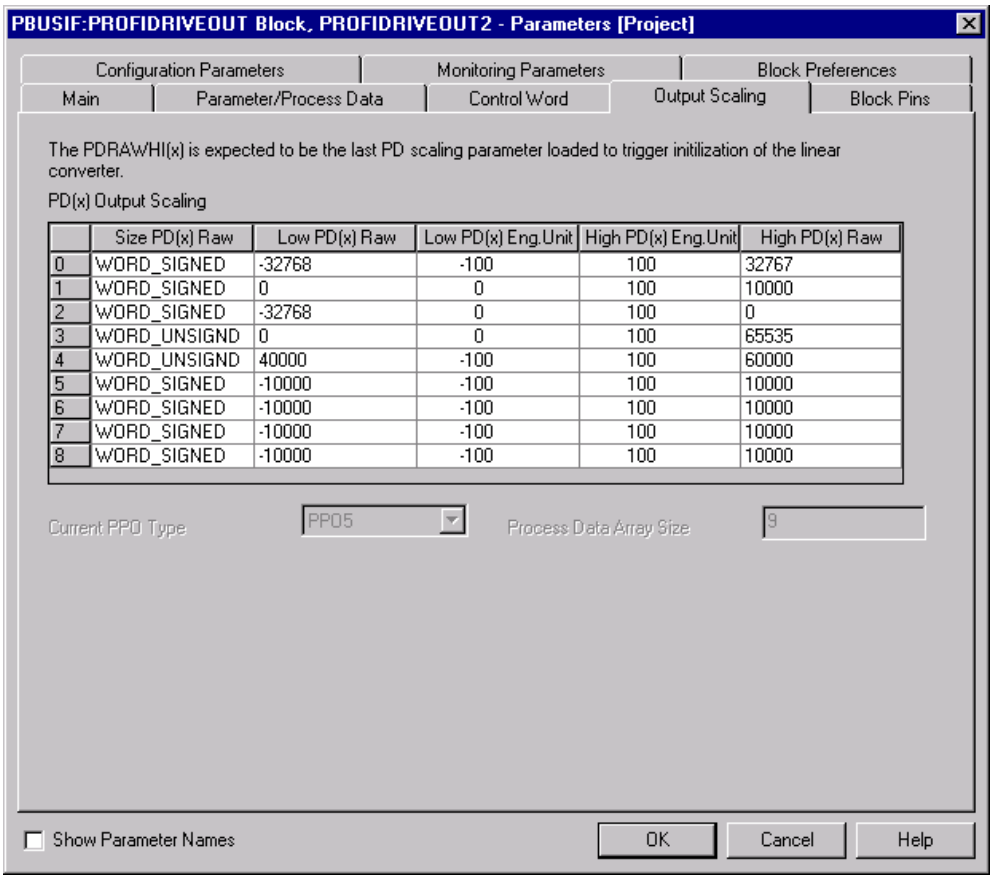
The descriptors are not repeated on the process data scaling tab, but are related to the scaling parameters as well as to the process data parameters.

10.4.6 Output Channel Block Configuration Parameters for the Parameter/Process Data tab

Parameter Name:	**PDDESC[0..8]**
Specific to Block:	PROFIDRIVEOUT
Description:	Descriptor of PDBIT[x]

Data Type:	24 Character Text String
Range:	N/A
Default:	
Config Load:	Yes
Access Lock:	No access permitted
Residence:	SR
Related Parameters:	PD[..], PDRAW[..], PDRAWSIZE[..],PDRAWLO[..], PDRAWHI[..], PDEULO[..], PDEUHI[..]
Remarks:	Because the PD[x] parameter cannot be renamed, this text descriptor is useful in documenting the use of each PD[x].

10.4.7 Output Channel Block Scaling Process Data



This tab displays the size of the raw data received from the device and the scaling parameters for Process Data. For configuration purposes this tab exposes the maximum number of elements, independent of the currently selected PPO type. Only after activation has the Input Channel block knowledge of the PPO type and will expose only the relevant data.

The raw data size parameter (PDRAWSIZE[x]) allows for selecting one of the following types:

Data Type	Description	Size (bytes)	Minimum Value	Maximum Value
WORD_SIGNED	Signed 16 bit integer	2	-32,767	32,768
WORD_UNSIGND	Unsigned 16 bit integer	2	0	65,536

Upon entry the scaling parameters PDRAWLO[x], PDRAWHI[x] are limit checked against PDRAWSIZE[x]. Along with PDEULO[x], PDEUHI[x] these 4 parameters will normally be obtained from technical documentation for the specific PROFIDRIVE device. PDRAWHI[x] is expected to be the last scaling parameter loaded to trigger the initialization of the linear converter.

10.4.8 Configuration Parameters for the Output Scaling tab

Parameter Name:	**PDRAWSIZE[0..8]**				
Specific to Block:	PROFIDRIVEOUT				
Description:	Raw integer data size and format				
Data Type:	Enumeration of ProfiDrive_DataSize				
Range:	The additional columns below indicate the physical size of each integer format and the associated minimum and maximum values for use in configuring the raw integer scaling parameters.				
	#	*Enum text*	*Size in bytes*	*Min. Raw Value*	*Max. Raw Value*
	0	WORD_SIGNED	2	-32,767	32,768
	1	WORD_UNSIGND	2	0	65,536
Default:	WORD_SIGNED				
Config Load:	Yes				
Access Lock:	Application Developer				
Residence:	CEE				
Related Parameters:	PDRAWLO [..], PDRAWHI [..], PD[..]				
Remarks:	PDRAWSIZE [x] should be entered prior to PDRAWLO [x], PDRAWHI [x] to allow for limit checking and clamping of these range parameters.				

Parameter Name:	**PDRAWLO[0..8]**
Specific to Block:	PROFIDRIVEOUT
Description:	Lower range limit of the integer input value.
Data Type:	INT32
Range:	See PDRAWSIZE[..], upon which this parameter is dependent.
Default:	-10,000
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	PDRAWSIZE[..], PDEULO[..], PD[..]
Remarks:	Upon entry PDRAWLO [x] will be limit checked against PDRAWSIZE [x]

Parameter Name:	**PDRAWHI[0..8]**
Specific to Block:	PROFIDRIVEOUT
Description:	Upper range limit of the integer input value.
Data Type:	INT32
Range:	See PDRAWSIZE[..], upon which this parameter is dependent.
Default:	10,000

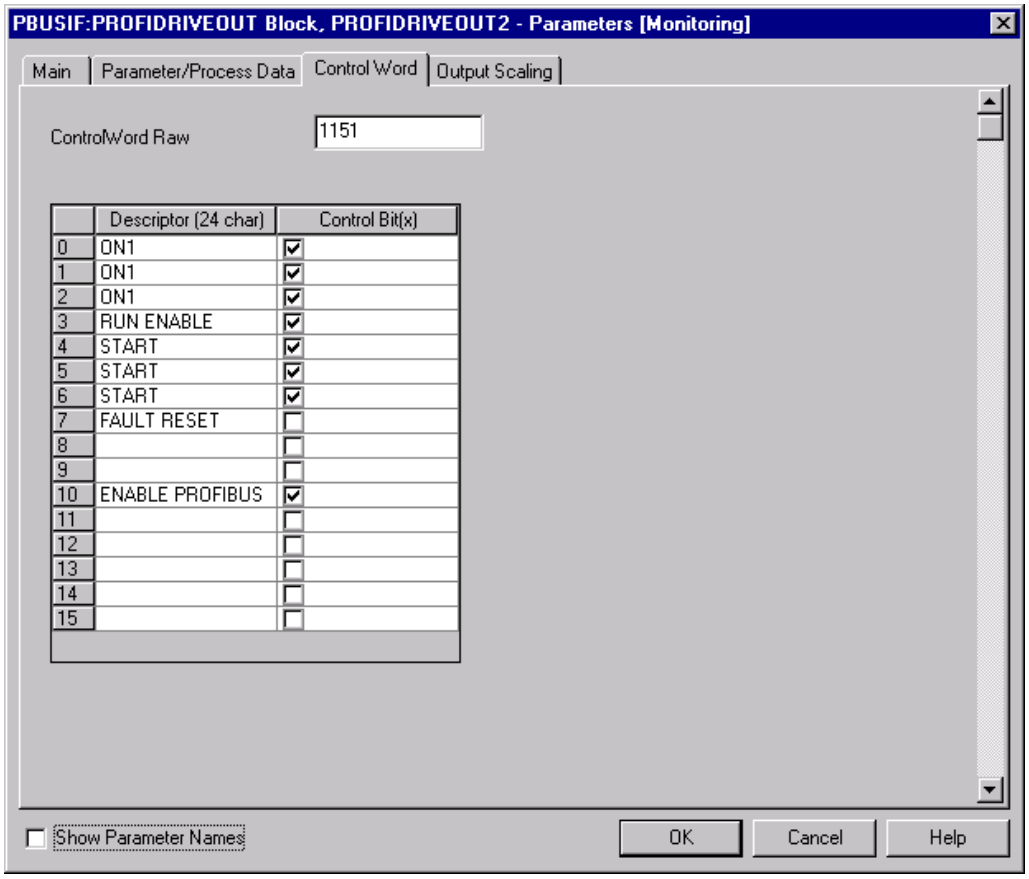
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	PDRAWSIZE[..], PDEUHI[..], PD[..]
Remarks:	Upon entry PDRAWHI [x] will be limit checked against PDRAWSIZE [x] PDRAWHI[x] is expected to be the last scaling parameter loaded to trigger the initialization of the linear converter.

Parameter Name:	**PDEULO[0..8]**
Specific to Block:	PROFIDRIVEOUT
Description:	Engineering Unit Low This parameter represents the numeric equivalent of PDRAWLO[x].
Data Type:	FLOAT64
Range:	
Default:	-100.0
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	PDRAWLO[..], PDEUHI[..], PD[..]
Remarks:	Although this parameter is usually less than PDEUHI[x] (representing the lowest numerical input value), if the inputs are scaled in a reverse fashion, this parameter can also be greater than PDEUHI[x] (thus representing the highest numerical input value).

Parameter Name:	PDEUHI[0..8]
Specific to Block:	**PROFIDRIVEOUT**
Description:	Engineering Unit High This parameter represents the numeric equivalent of PDRAWHI[x].
Data Type:	FLOAT64
Range:	
Default:	100.0
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	PDRAWHI[..], PDEULO[..], PD[..]
Remarks:	Although this parameter is usually greater than PDEULO[x] (representing the highest numerical input value), if the inputs are scaled in a reverse fashion, this parameter can also be less than PDEULO[x] (thus representing the lowest numerical input value).

10.4.9 Output Channel Block Operation

10.4.10 Output Channel Block Control Word Monitoring tab



This monitoring tab displays a live Control Word as a 16bit unsigned integer raw value parameter (CWRAW) and the corresponding 16 Control Bits as Boolean parameters (CWBIT[x]). The corresponding descriptors (CWDESC[x]) have been entered in project mode.

The Control Word can be updated through either the individual Boolean parameters or the UINT16 sized raw parameter. Each parameter entry method updates also the associated parameter.

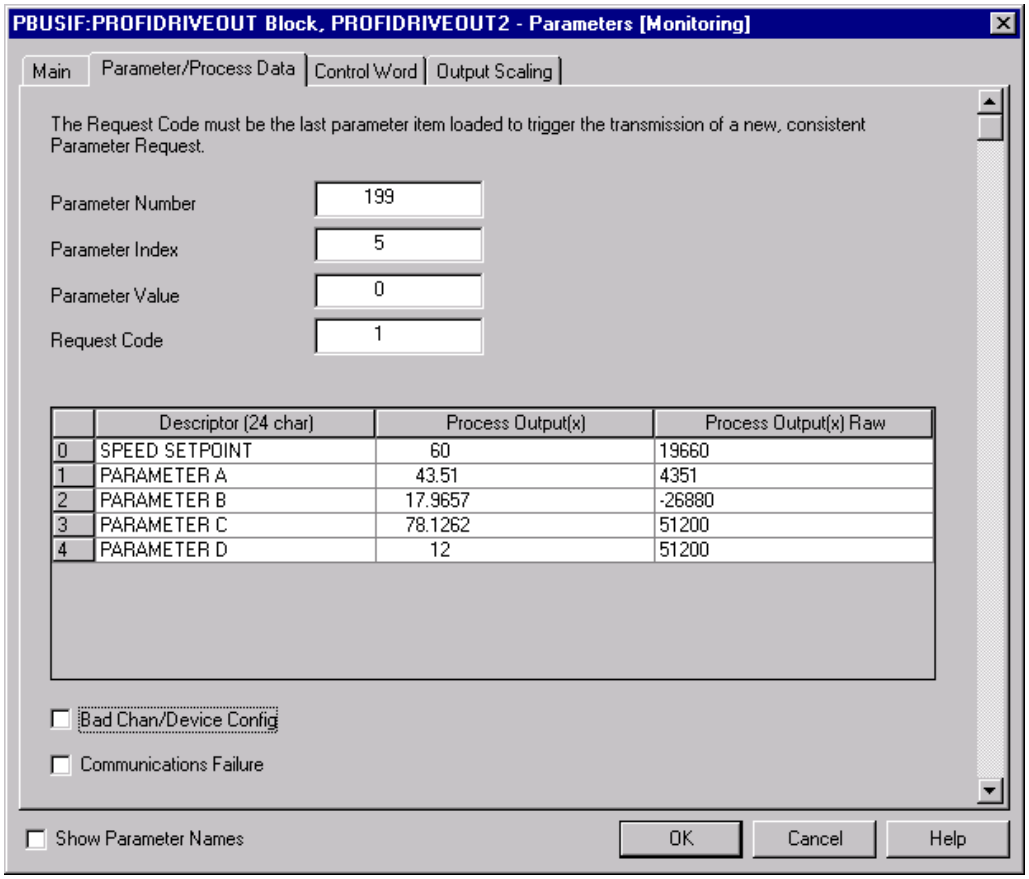
10.4.11 Operational Parameters for the Control Word tab

Parameter Name:	**CWRAW**
Specific to Block:	PROFIDRIVEOUT
Description:	Control Word
Data Type:	INT32
Range:	0....65535
Default:	0
Config Load:	No
Access Lock:	Engineer
Residence:	CEE
Related Parameters:	CWBIT[..]

Remarks:	Parameter CWRAW is the 16bit control word sent to the device.

Parameter Name:	**CWBIT[0..15]**
Specific to Block:	PROFIDRIVEOUT
Description:	Control Word Bit
Data Type:	BOOLEAN
Range:	OFF(0), ON(1)
Default:	OFF
Config Load:	No
Access Lock:	Engineer
Residence:	CEE
Related Parameters:	CWRAW
Remarks:	Boolean parameter CWBIT[x] corresponds to bit position 'x' of CWRAW.

10.4.12 Output Channel Block Parameter/Process Data Monitoring tab



This parameter/process data monitoring tab displays live data typical for the PPO2 format selection: parameter data and 5 process data elements.

The upper part displays a parameter request using parameters PARNUM, PARIDX, PARVAL, REQUCODE. This set of parameters is dependant on each other. Only if this request entity is complete and consistent in itself must the output channel block send this new request to the PROFIDRIVE device. Consistency is reached by

using parameter REQUCODE as an indicator for request completion. After having entered PARNUM, PARIDX and PARVAL will the entry of REQUCODE initiate the transmission of a new request.

Since REQUCODE is set to 1 (read request for this device), the request is for the current value of parameter 199[5]. In case of a read request PARVAL will be ignored. A device specific user manual is required to interpret the data. Note that all parameters are unscaled raw data.

The lower part displays data entered in 5 process data elements PD[x] and PDRAW[x]. Any value entered into PD[x] will be limit checked against its PDEULO and PDEUHI engineering units and clamped to the respective limit. According to its scaling parameters PD[x] will be converted to its raw data equivalent PDRAW[x]. The corresponding descriptors PDDESC[x] have been entered in project mode.

10.4.13 Output Channel Block Operational Parameters for the Parameter/Process Data Monitoring tab

Parameter Name:	**PARNUM**
Specific to Block:	PROFIDRIVEOUT
Description:	Parameter Number
Data Type:	INT32
Range:	0....2047
Default:	0
Config Load:	No
Access Lock:	Engineer
Residence:	CEE
Related Parameters:	PARIDX, PARVAL, REQUCODE
Remarks:	PARNUM is the Parameter Number used in the current parameter request.

Parameter Name:	**PARIDX**
Specific to Block:	PROFIDRIVEOUT
Description:	Parameter Number Index
Data Type:	INT32
Range:	0....65535
Default:	0
Config Load:	No
Access Lock:	Engineer
Residence:	CEE
Related Parameters:	PARNUM, PARVAL, REQUCODE
Remarks:	PARIDX is the index for the Parameter Number used in the current parameter request.

Parameter Name:	**PARVAL**
Specific to Block:	PROFIDRIVEOUT
Description:	Parameter Value
Data Type:	FLOAT64
Range:	0....4 294 967 295 (UINT32 range)
Default:	0
Config Load:	No
Access Lock:	Engineer

Residence:	CEE
Related Parameters:	PARNUM, PARIDX, REQUCODE
Remarks:	PARVAL is the value to be sent to PARNUM[PARIDX], if parameter write request. PARVAL is ignored, if parameter read request.

Parameter Name:	**REQUCODE**
Specific to Block:	PROFIDRIVEOUT
Description:	Parameter Request Code
Data Type:	INT32
Range:	0....15
Default:	0
Config Load:	No
Access Lock:	Engineer
Residence:	CEE
Related Parameters:	PARNUM, PARIDX, PARVAL
Remarks:	REQUCODE is expected to be the last parameter loaded within the set of request parameters, to trigger the transmission of a new consistent Parameter Request.

Parameter Name:	**PD[0..8]**
Specific to Block:	PROFIDRIVEOUT
Description:	Process Output[x], Scaled
Data Type:	FLOAT64
Range:	
Default:	0.0
Config Load:	No
Access Lock:	Engineer
Residence:	CEE
Related Parameters:	PDDESC[..], PDRAW[..], PDRAWSIZE[..], PDRAWLO[..], PDRAWHI[..], PDEULO[..], PDEUHI[..]
Remarks:	

Parameter Name:	**PDRAW[0..8]**
Specific to Block:	PROFIDRIVEOUT
Description:	Process Output[x] raw value This value is the raw data equivalent of PD[x]. It is the field value that is transmitted to the PROFIDRIVE device. It is calculated by the block using the corresponding scaling parameters.
Data Type:	INT32
Range:	Varies based on PDRAWSIZE[..]
Default:	0
Config Load:	No
Access Lock:	View Only
Residence:	CEE

Related Parameters:	PDDESC[..], PD[..], PDRAWSIZE[..], PDRAWLO[..], PDRAWHI[..], PDEULO[..], PDEUHI[..]
Remarks:	

11 BIZERBA ST Function Blocks

Related topics

“BIZERBA ST Function Blocks overview” on page 158
“BIZERBA ST Device Block” on page 159
“BIZERBA ST Input Channel Block” on page 163
“BIZERBA ST Output Channel Block” on page 171
“BIZERBA ST Parameter Reference” on page 177

11.1 BIZERBA ST Function Blocks overview

Related topics

“About Bizerba ST” on page 158
“BIZERBA ST Function Blocks Template Names” on page 158

11.1.1 About Bizerba ST

The Bizerba Weighing Terminal ST provides status information about the weighing process itself and the progress of function requests sent to the terminal. You can configure up to six process variables for weight results, such as gross weight, net weight, and tare, or other values depending on the industrial application.

11.1.2 BIZERBA ST Function Blocks Template Names

The following function blocks described in this section are based upon the fixed input/output (I/O) telegram sizes used by the Bizerba ST device. The following table identifies the block template name used to identify the block in the Control Builder application.

Block Description	Block Template Name in Control Builder
BIZERBA ST Device Block	BIZERBASTDEV
BIZERBA ST Input Channel Block	BIZERBASTIN
BIZERBA ST Output Channel Block	BIZERBASTOUT

11.2 BIZERBA ST Device Block

Related topics

“BIZERBA ST Device Block description” on page 159
“BIZERBA ST Device Block Channel Assignment” on page 159
“Diagnostic Message Support” on page 159
“Configuring BIZERBA ST device block” on page 159
“BIZERBA ST device operational characteristics” on page 162

11.2.1 BIZERBA ST Device Block description

The BIZERBA ST (BIZERBASTDEV) block represents the device itself. It provides process data through the input (BIZERBASTIN) and output (BIZERBASTOUT) channel blocks that are associated with this device block.

11.2.2 BIZERBA ST Device Block Channel Assignment

The BIZERBA ST device block is capable of hosting only two process I/O channel blocks. Channel 0 is reserved for assignment of the BIZERBA ST input channel block (BIZERBASTIN), and channel 1 is reserved for assignment of the BIZERBA ST output channel block (BIZERBASTOUT).

11.2.3 Diagnostic Message Support

All device IOM blocks support the processing of event-initiated diagnostic data messages send by the Profibus slave device. If these messages are to be interpreted, the assignment of optional diagnostic channel blocks is required in addition to the regular process I/O channel blocks. Each of these diagnostic input channel blocks is capable of processing up to 8 numeric and 32 discrete diagnostic data items. The number of available diagnostic channel blocks varies from one device to the next, depending on the maximum diagnostic message size specified in the GSD-file. During channel assignment, the number of diagnostic channels allowed for a specific device IOM, can be viewed in the channel assignment window.

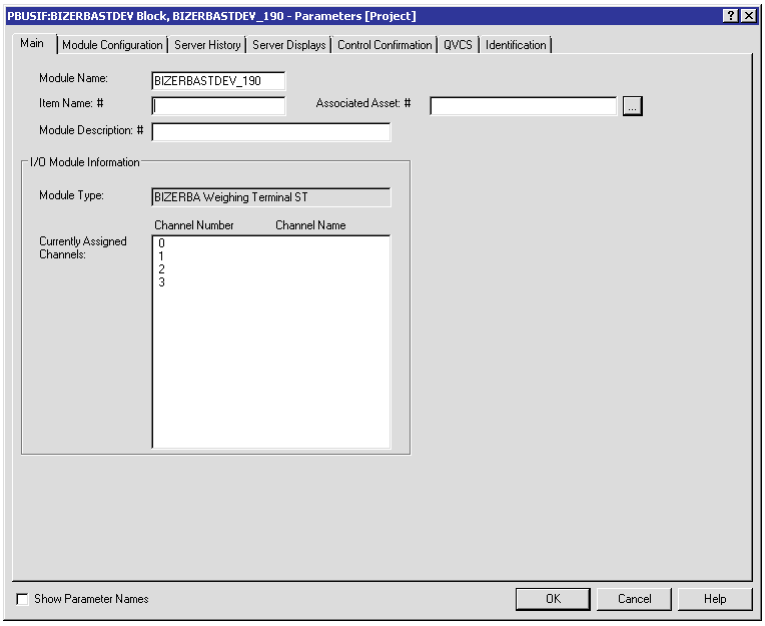
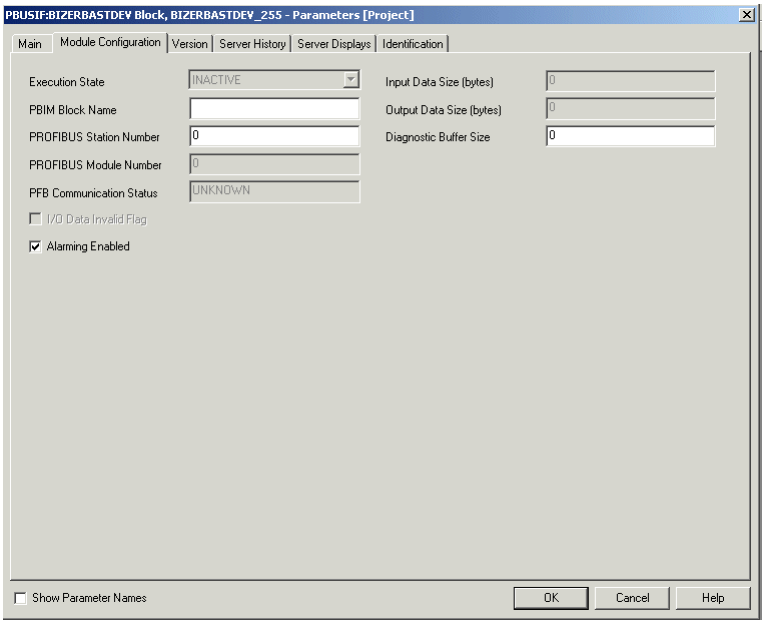
Channels 2-3 are reserved for a maximum of 2 diagnostic channels (PBDIAGNOSTIC).

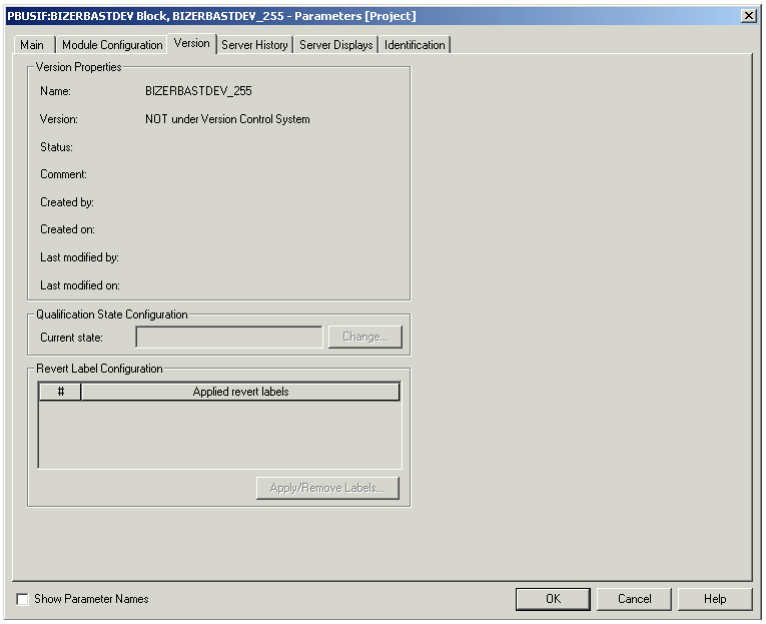
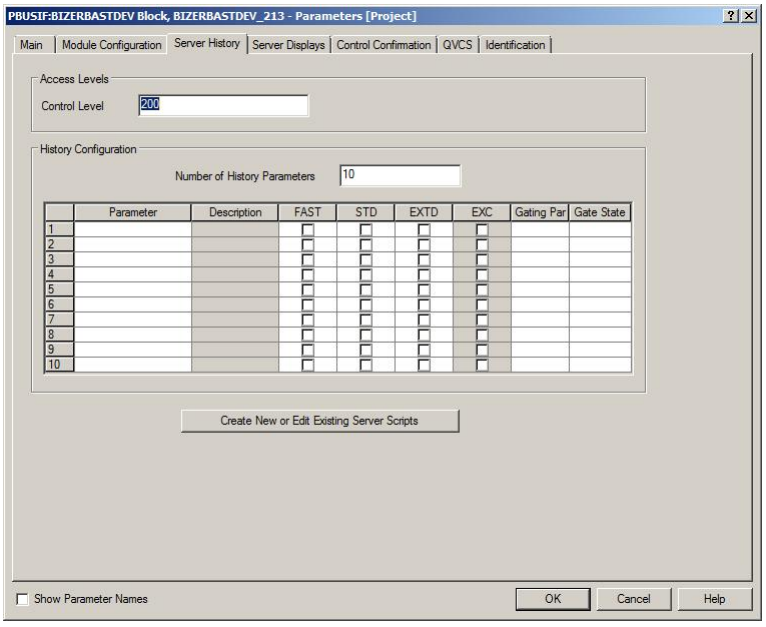
The maximum message size for diagnostic data is 6 bytes, coded as read-only parameter DIAGBUFSIZE.

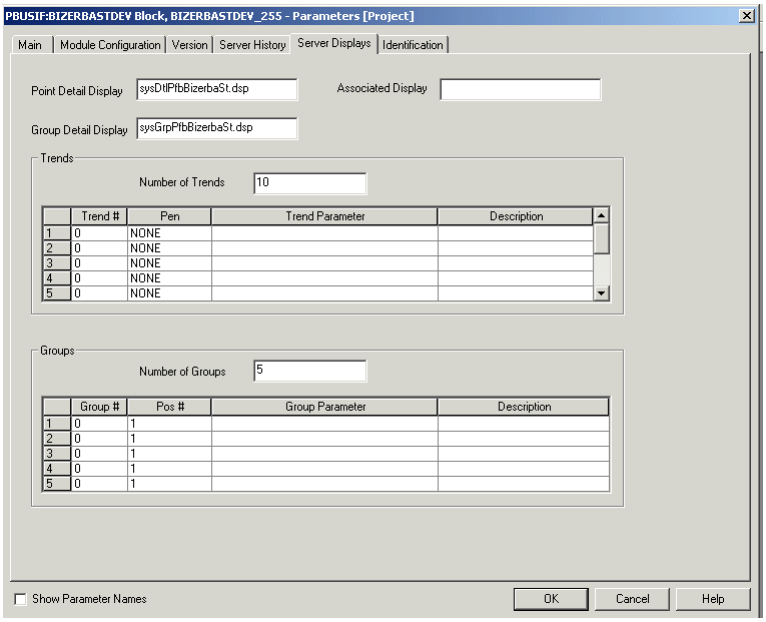
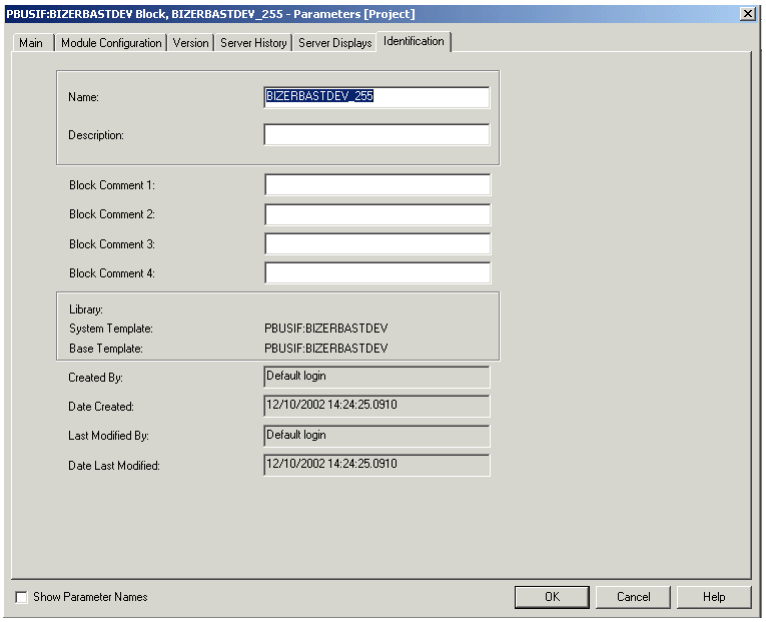
For more information on Profibus diagnostic messages and the diagnostic channel block refer to “PROFIBUS Diagnostic Data Messages”.

11.2.4 Configuring BIZERBA ST device block

You configure a Bizerba ST device (BIZERBASTDEV) block the same way you configure any other block in Control Builder. You can click and drag the BIZERBASTDEV icon from the Library tab to the Project tab to add it to a control strategy in Project. The following table shows the configuration forms associated with a BIZERBASTDEV block that include specific parameters for viewing and configuring the device within the system. Use the online help to access a definition for a given parameter or search for a parameter definition within this document.

If Configuration Form Tab Is . . .	Then, use it to . . .
Main  	Enter a unique name for the device and your own personal description for it. In addition, also select an asset from those configured in the Enterprise Model Database to set the Scope of Responsibility (SOR) for the point. View module type and current channel assignments.
Module Configuration  	Enter module related configuration data including the 'PBIM Block Name' and 'PROFIBUS Station Number' fields that represent the SRCBLOCK and STATION parameters, respectively. These parameters, as well as the Diagnostic Buffer Size (DIAGBUFSIZE) parameter, are common to all PROFIBUS device/module blocks and are described in the “Commonality of PROFIBUS I/O Device/Module Blocks” on page 65 section. Because the BIZERBA ST device supports only a single module, the module number is not configurable, and is set to 0 by default. The input and output data sizes for the BIZERBA ST device are fixed and not configurable.

If Configuration Form Tab Is . . .	Then, use it to . . .
Version  	View information pertinent to the Version Control System. You must have a Version Control System license to use this form.
Server History  	Configure server data pertinent to data access and history collection.

If Configuration Form Tab Is . . .	Then, use it to . . .
Server Displays 	Configure server data pertinent to displaying data in Station.
Identification 	View and configure data associated with the block template. You must have a template license to use this form.

11.2.5 BIZERBA ST device operational characteristics

Refer to “PROFIBUS I/O Device/Module Blocks operation” on page 68 in the Commonality of PROFIBUS I/O Device/Module Blocks section for a description of operational characteristics that are common to all PROFIBUS device/module blocks.

The detailed operational characteristics and parameters of the BIZERBA ST device are accessed through the BIZERBA ST input and output channel blocks.

11.3 BIZERBA ST Input Channel Block

Related topics

“About the BIZERBA ST Input Channel Block” on page 163
“Input data message format” on page 163
“Configuring BIZERBASTIN block” on page 163

11.3.1 About the BIZERBA ST Input Channel Block

The BIZERBA ST Input Channel (BIZERBASTIN) provides the control interface to all input parameters from the BIZERBA ST device. You can only assign the BIZERBASTIN block to channel 0 of the BIZERBA ST Device block.

Refer to the “Commonality of PROFIBUS I/O Channel Blocks” on page 71 section for attributes that are common to all PROFIBUS channel blocks.

11.3.2 Input data message format

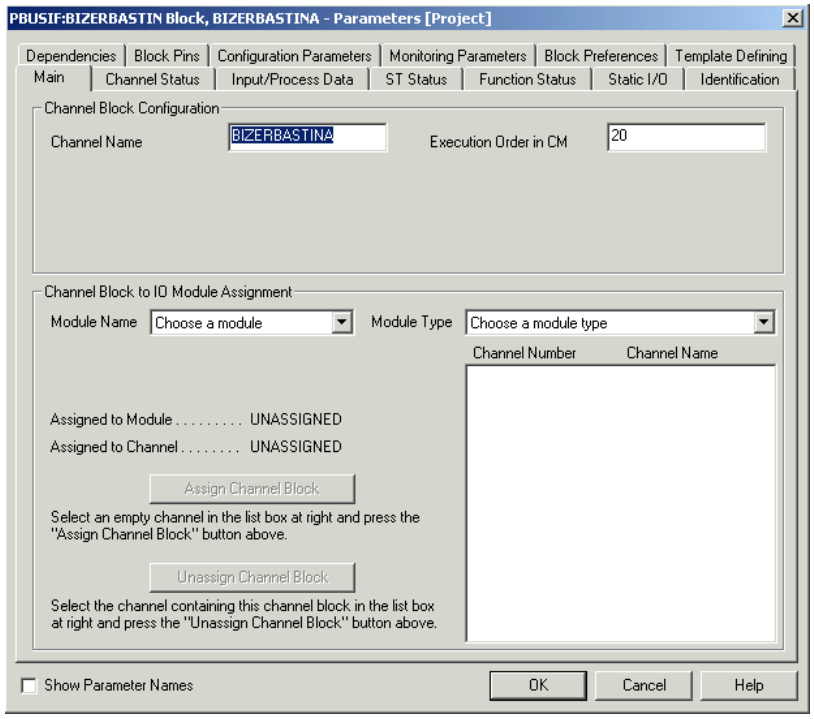
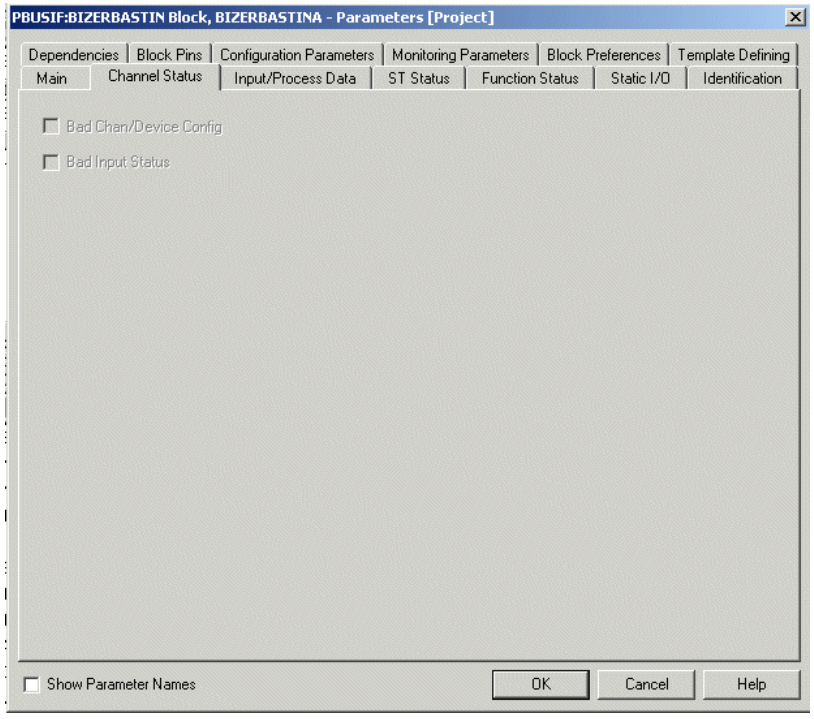
The input data message is 32-Bytes long and has the following fixed format.

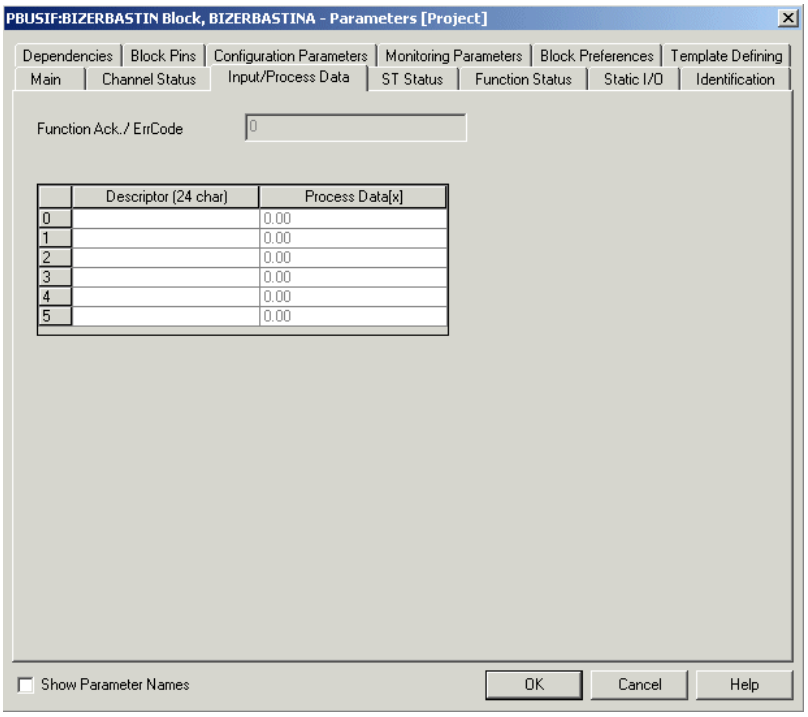
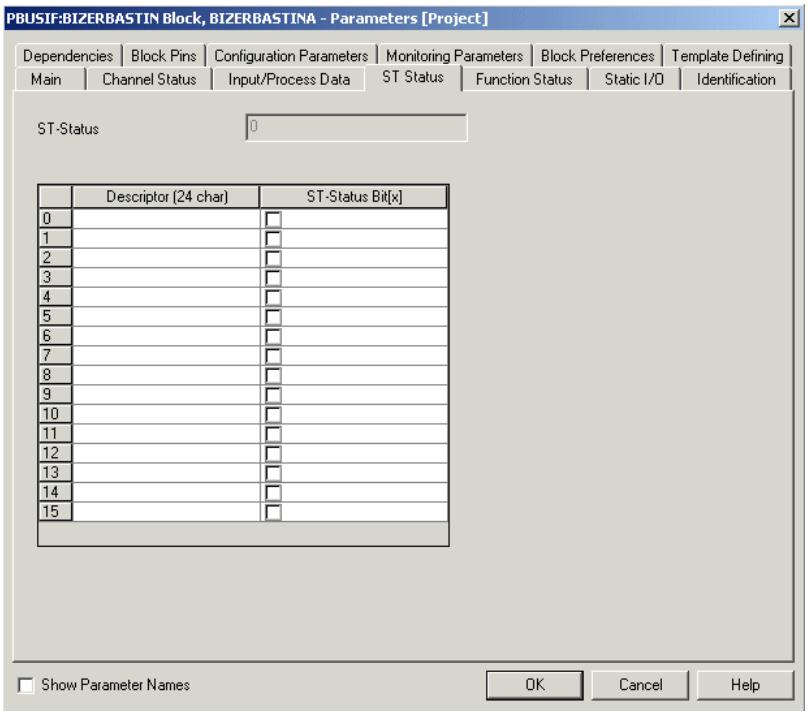
Parameter	Format	Length in Bytes
error code / function acknowledge	unsigned integer numeric value	2
ST-status	16 discrete values	2
function status	16 discrete values	2
static inputs	8 discrete values	1
static outputs	8 discrete values	1
process value 1	IEEE real numeric value	4
process value 2	IEEE real numeric value	4
....	....	....
process value 6	IEEE real numeric value	4

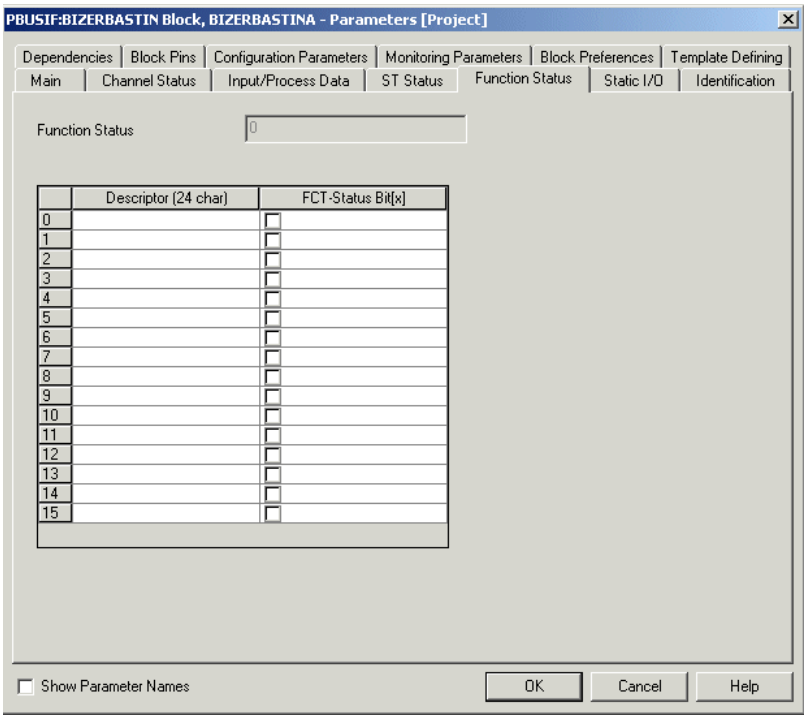
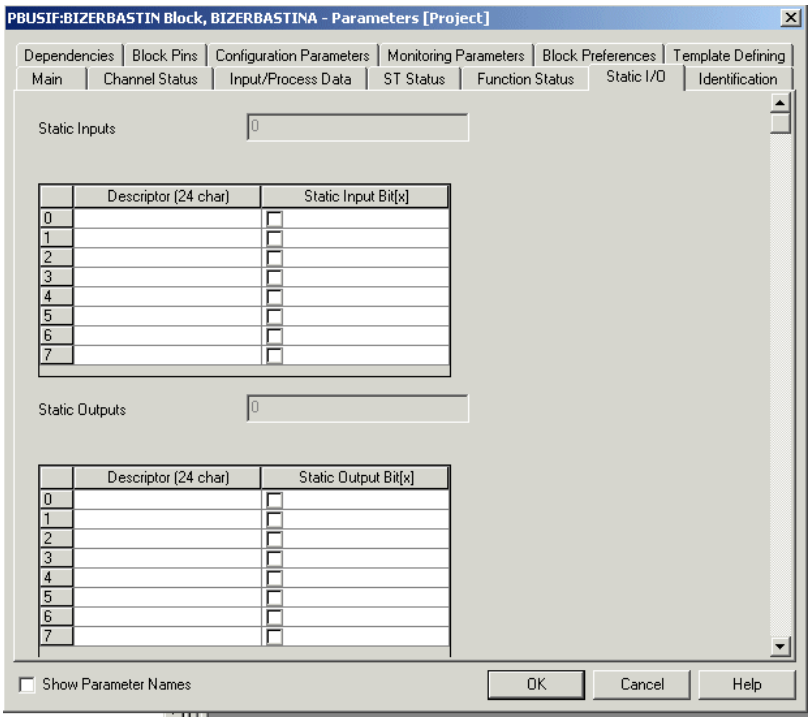
11.3.3 Configuring BIZERBASTIN block

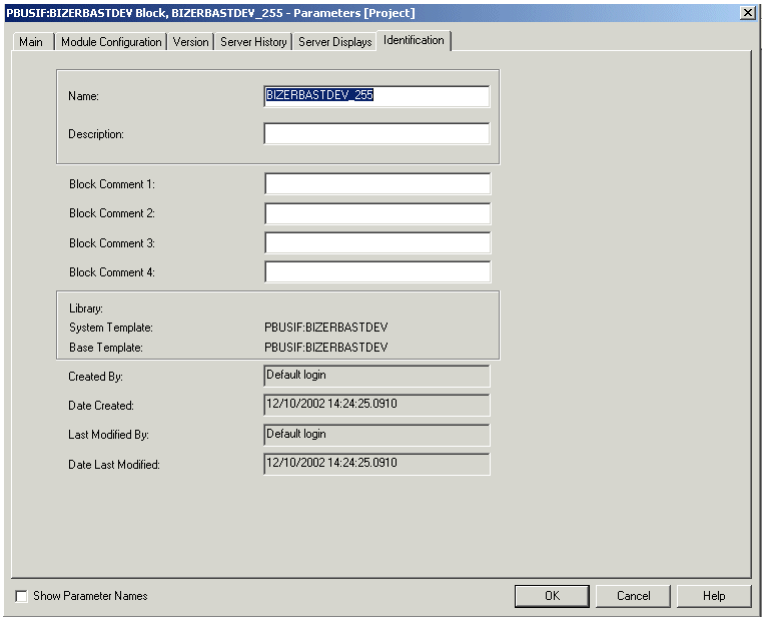
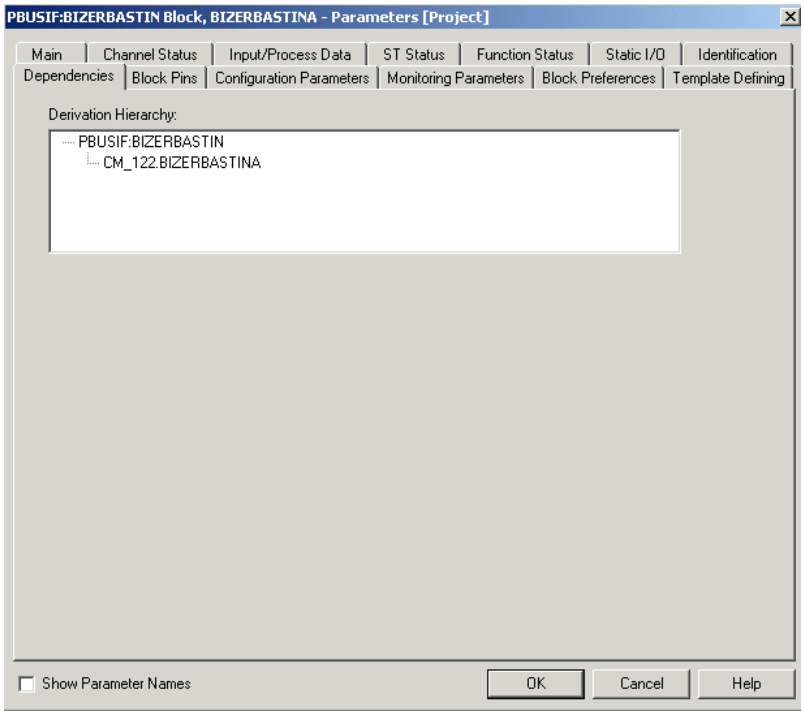
You configure a Bizerba ST input (BIZERBASTIN) block the same way you configure any other block in Control Builder. You can click and drag the BIZERBASTIN icon from the Library tab to an open Control Module chart to add it to a control strategy in Project. The following table shows the configuration forms associated with a BIZERBASTIN block that include specific parameters for viewing and configuring the device within the system. Use the online help to access a definition for a given parameter or search for a parameter definition within this document.

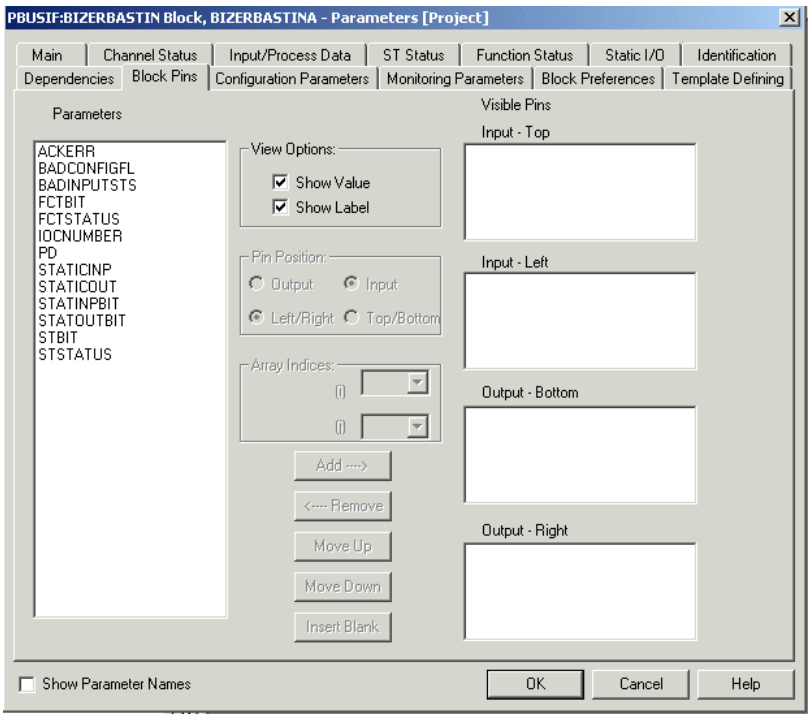
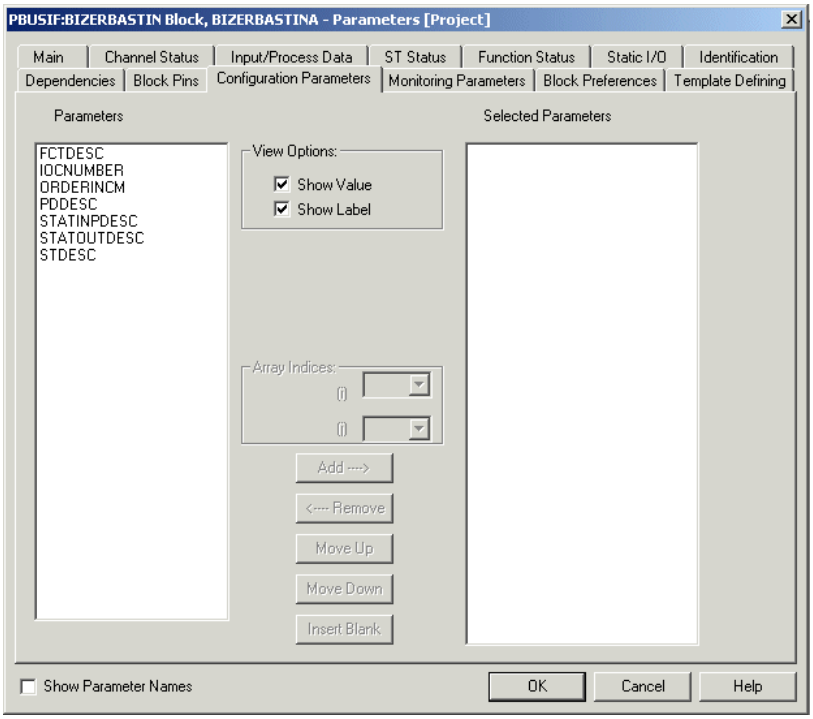
Tip
Besides the channel assignment, block execution order, and any desired change in the parameter or pin exposure on the block, the BIZERBA ST Input Channel block requires no configuration. Since the process data is configurable at the terminal and the status information is either depending on the application or configurable as well, descriptors have been provided for operator convenience.

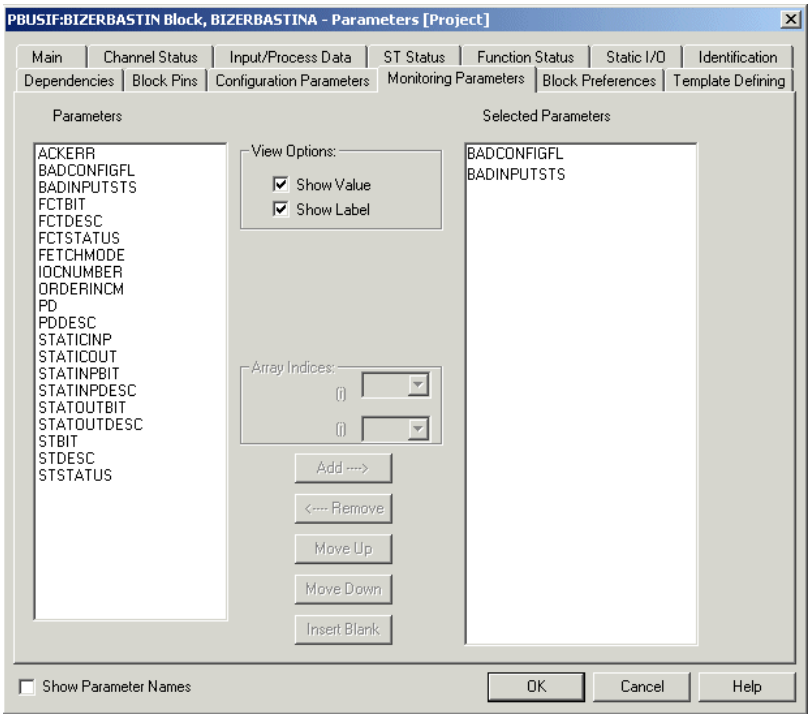
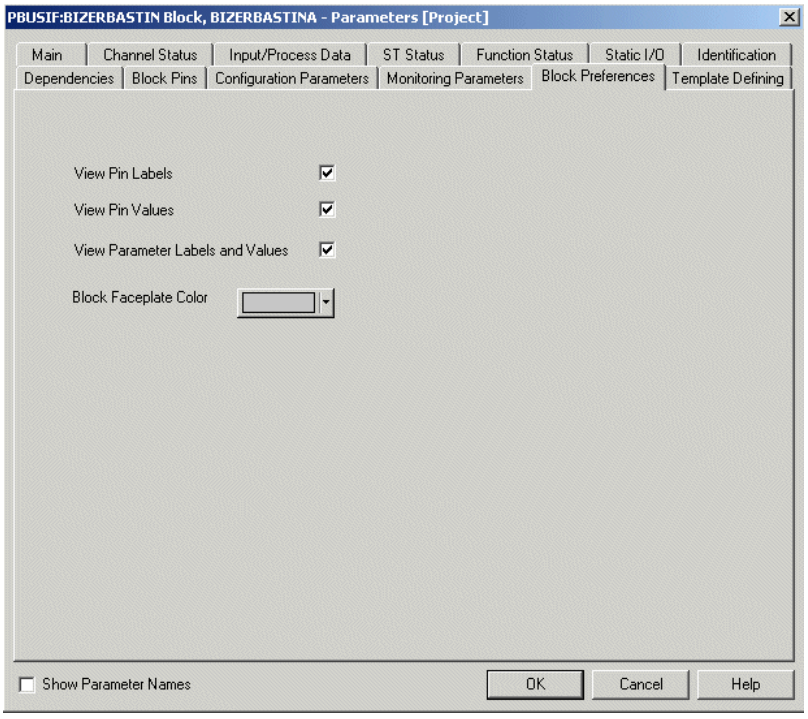
If Configuration Form Tab Is . . .	Then, use it to . . .
Main  	Enter a unique name for the device and desired block execution order. Assign or unassign channels for given module.
Channel Status  	View channel status in Monitor mode.

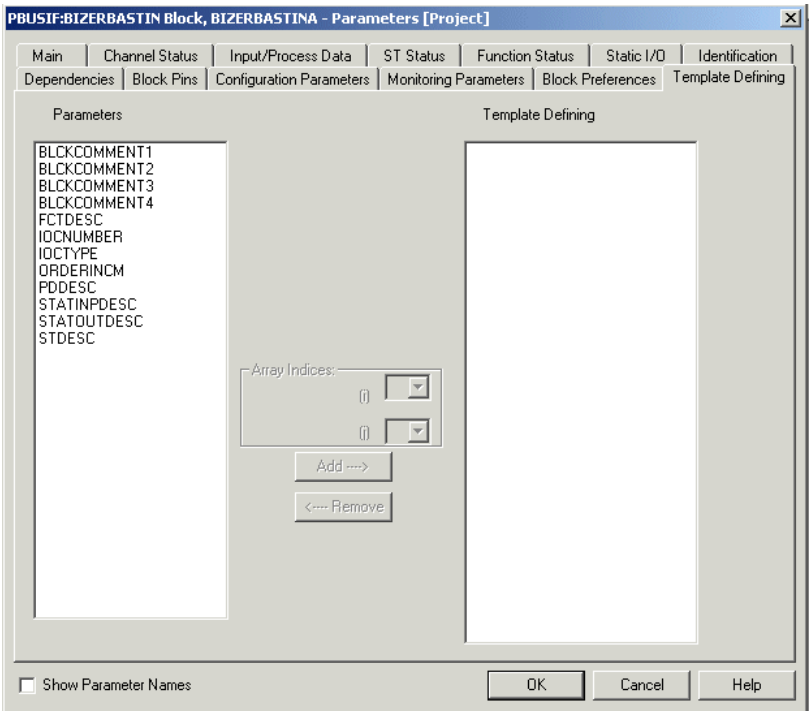
If Configuration Form Tab Is . . .	Then, use it to . . .
Input/Process Data 	Configure 24-character descriptor for process data values. View 16-bit unsigned integer value for Function Acknowledge/Error Code in Monitor mode.
ST Status 	Configure 24-character descriptor for ST Status Bit values. View 16-bit unsigned integer value for BIZERBA ST status word.

If Configuration Form Tab Is . . .	Then, use it to . . .
Function Status  	Configure 24-character descriptor for function status bit. View 16-bit unsigned integer value for BIZERBA ST function status word.
Static I/O  	Configure 24-character descriptor for to identify use of corresponding static input bit or static output bit. View 8-bit unsigned integer raw values for BIZERBA ST static inputs and outputs.

If Configuration Form Tab Is . . .	Then, use it to . . .
Identification 	View and configure data associated with the block template. You must have a template license to use this form.
Dependencies 	View template hierarch derivation. . You must have a template license to use this form.

If Configuration Form Tab Is . . .	Then, use it to . . .
Block Pins 	Configure block pins to be exposed on the block.
Configuration Parameters 	Configure parameters to be exposed on block face in Project mode.

If Configuration Form Tab Is . . .	Then, use it to . . .
Monitoring Parameters 	Configure parameters to be exposed on block face in Monitor mode.
Block Preferences 	Configure desired block display preferences.

If Configuration Form Tab Is . . .	Then, use it to . . .
Template Defining  	View/configure parameters to be designated for the template. You must have a template license to use this form.

11.4 BIZERBA ST Output Channel Block

Related topics

“About the BIZERBA ST Output Channel Block” on page 171
“Output data message format” on page 171
“Configuring BIZERBASTOUT block” on page 171

11.4.1 About the BIZERBA ST Output Channel Block

The BIZERBA ST Output Channel block (BIZERBASTOUT) provides the control interface to all output parameters to the BIZERBA ST device. You can only assign the BIZERBASTOUT block to channel 1 of the BIZERBA ST Device block.

Refer to the “Commonality of PROFIBUS I/O Channel Blocks” on page 71 section for attributes that are common to all PROFIBUS channel blocks.

11.4.2 Output data message format

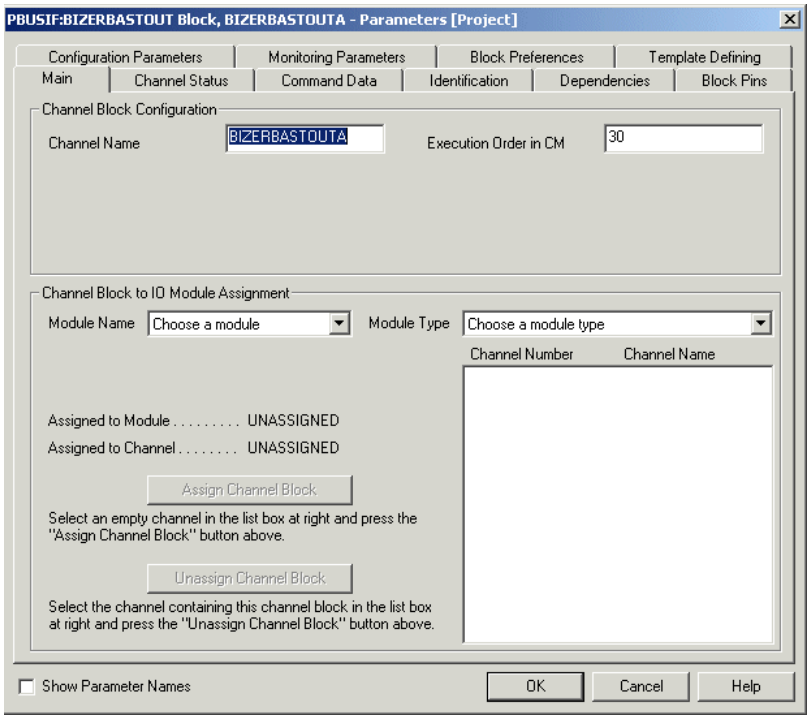
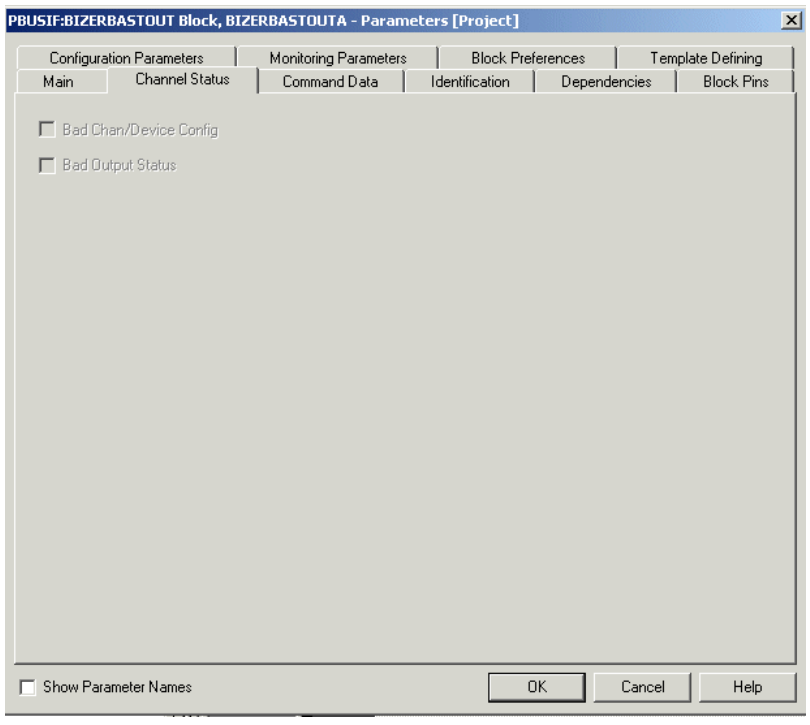
The output data message is 8-Bytes long and has the following fixed format.

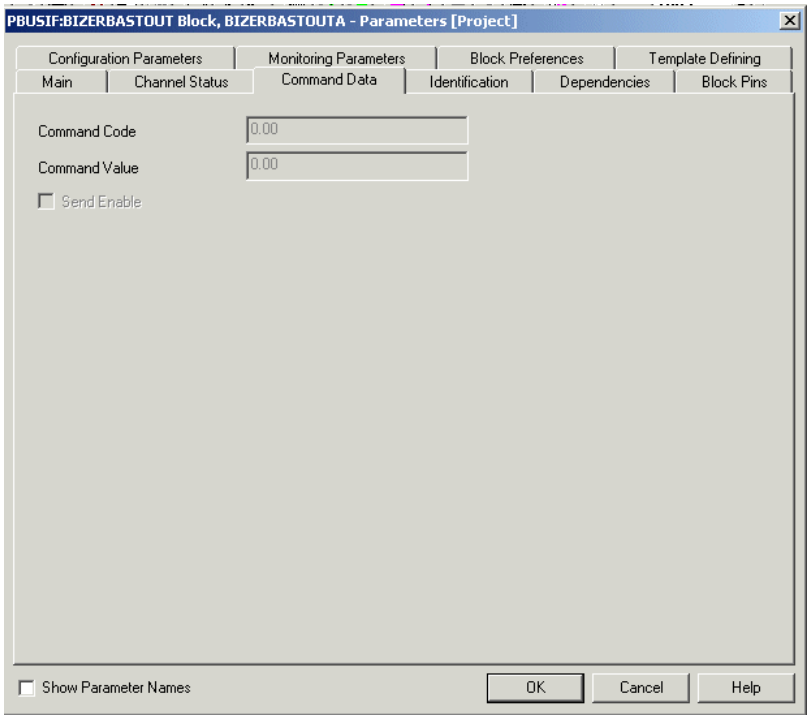
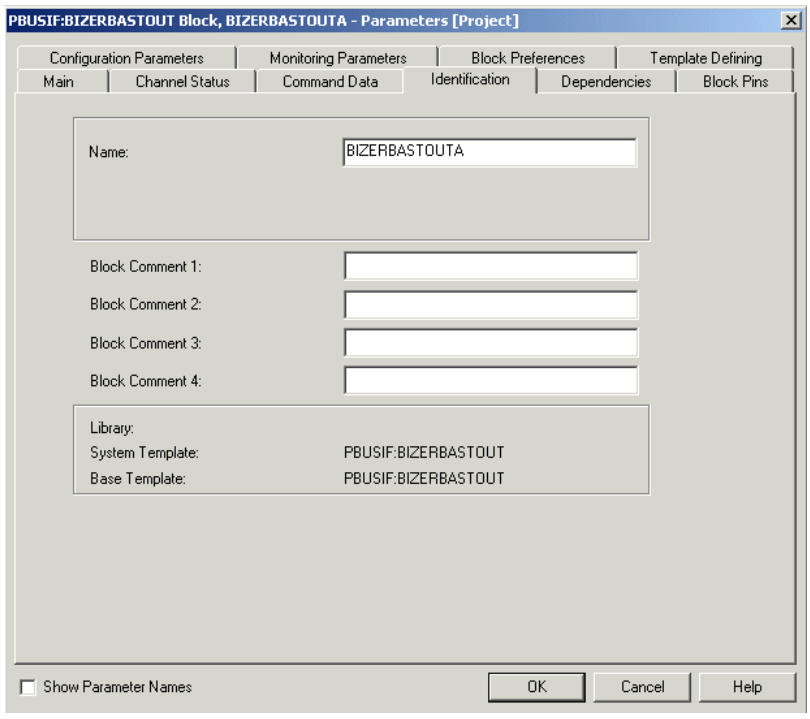
Parameter	Format	Length in Bytes
command code	IEEE real numeric value	4
command value	IEEE real numeric value	4

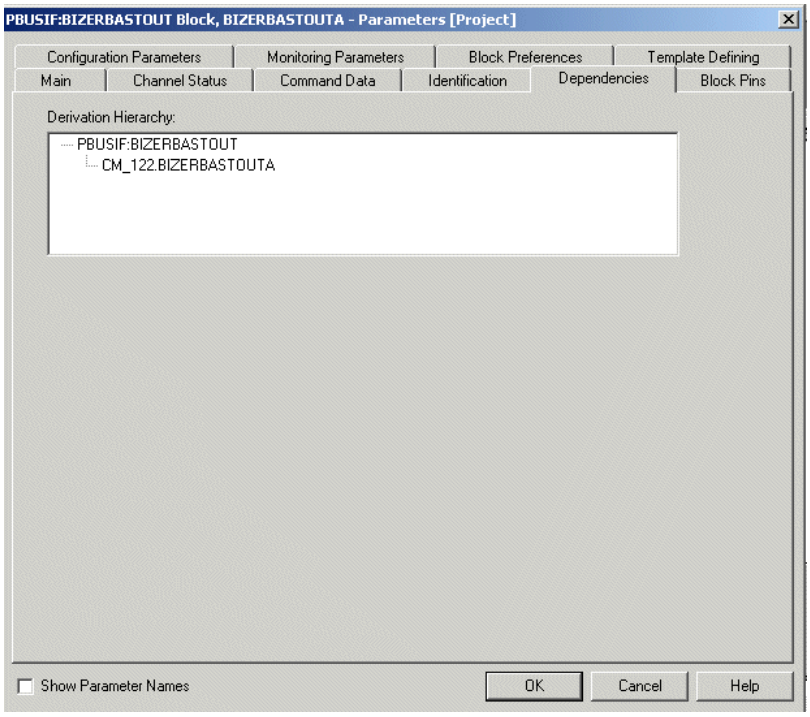
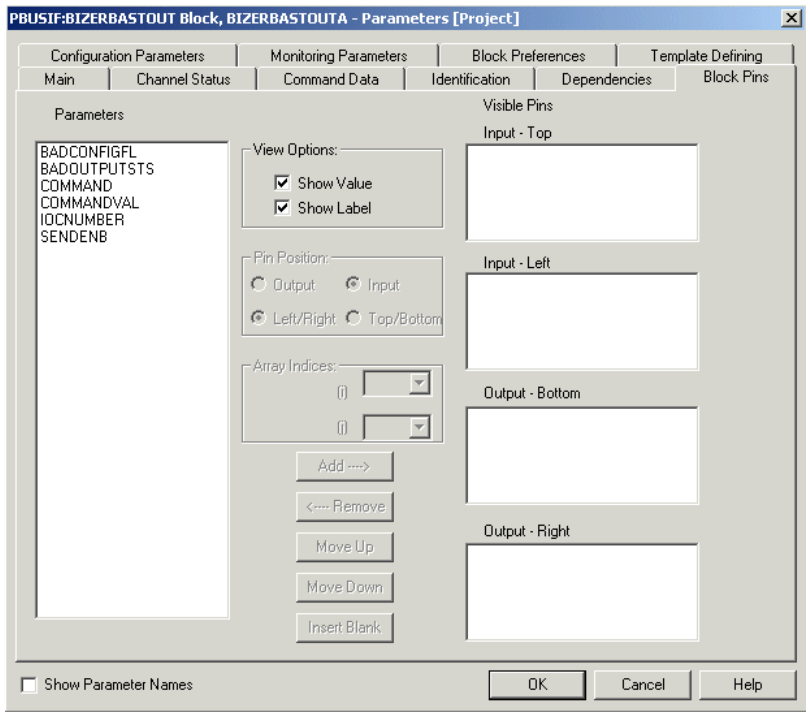
11.4.3 Configuring BIZERBASTOUT block

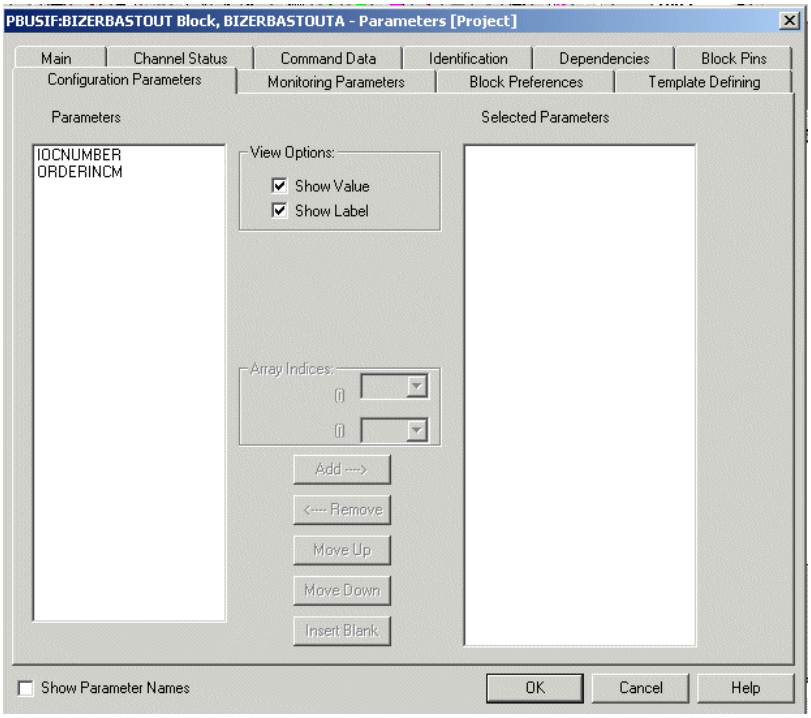
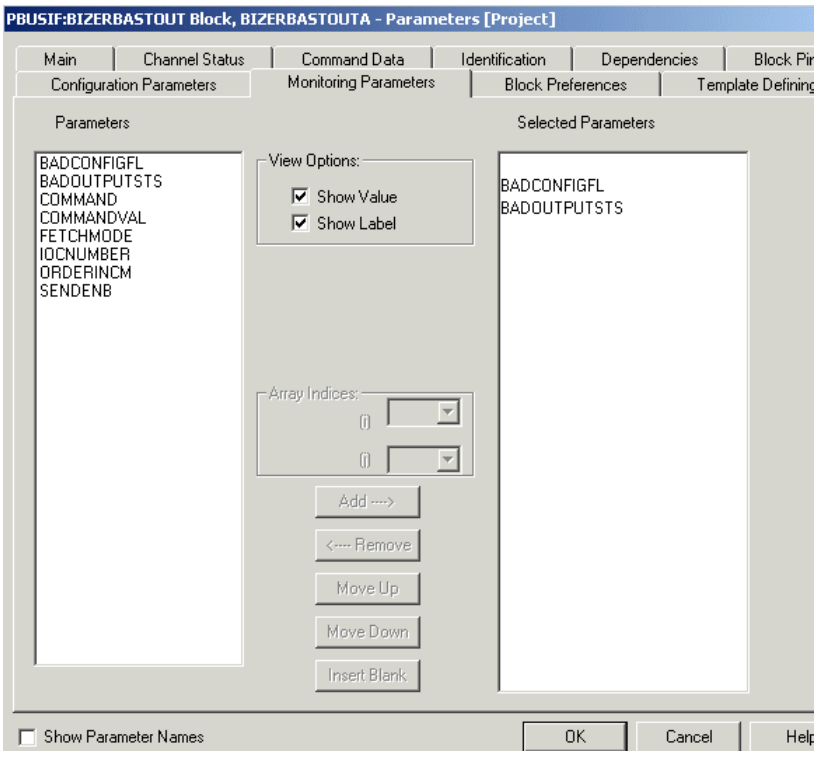
You configure a Bizerba ST output (BIZERBASTOUT) block the same way you configure any other block in Control Builder. You can click and drag the BIZERBASTOUT block icon from the Library tab to an open Control Module chart to add it to a control strategy in Project. The following table shows the configuration forms associated with a BIZERBASTOUT block that include specific parameters for viewing and configuring the device within the system. Use the online help to access a definition for a given parameter or search for a parameter definition within this document.

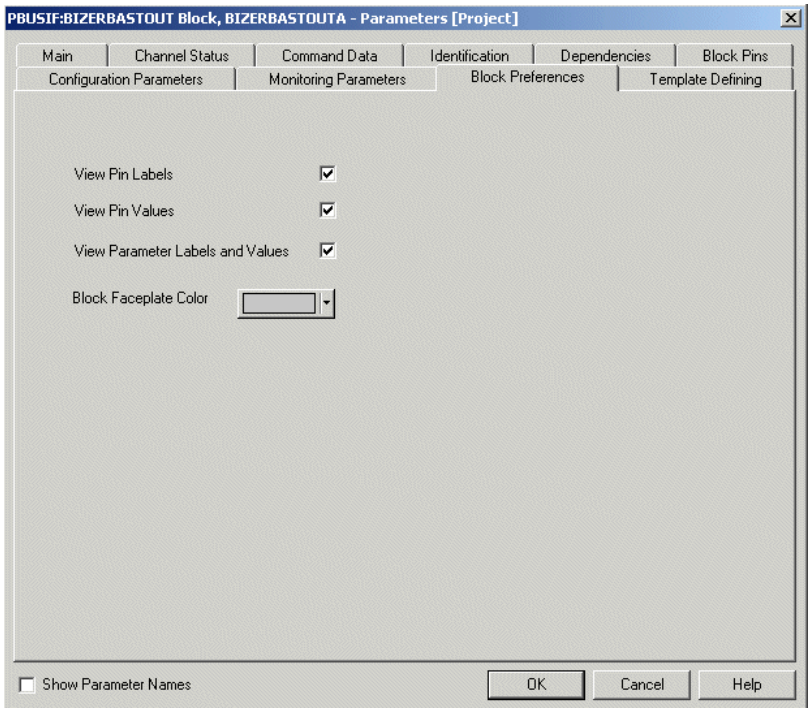
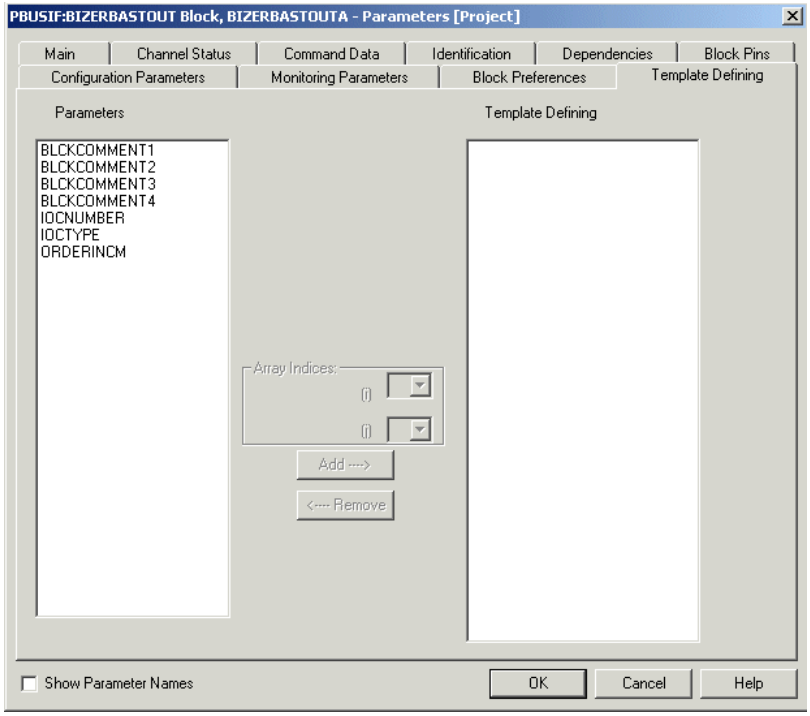
Tip
Besides the channel assignment, block execution order, and any desired change in the parameter or pin exposure on the block, the BIZERBA ST Output Channel block requires no configuration.

If Configuration Form Tab Is . . .	Then, use it to . . .
Main  	Enter a unique name for the device and desired block execution order. Assign or unassign channels for given module.
Channel Status  	View channel status in Monitor mode.

If Configuration Form Tab Is . . .	Then, use it to . . .
Command Data 	Send Commands/Function Requests to the Bizerba ST device. All Function Requests consist of the Command Code itself and, if required by the function, an additional Command Value. Command Code (COMMAND) and Command Value (COMMANDVAL) form a request entity, which must be complete and consistent in itself prior to sending a new Function Request to the BIZERBA ST device. Consistency is reached by using parameter SENDENB as an indicator for request completion. After having entered COMMAND and COMMANDVAL, setting of SENDENB will initiate the transmission of a new Function Request. SENDENB is reset automatically after the first successful transmission of the new Function Request.
Identification 	View and configure data associated with the block template. You must have a template license to use this form.

If Configuration Form Tab Is . . .	Then, use it to . . .
Dependencies  	View template hierarch derivation. . You must have a template license to use this form.
Block Pins  	Configure block pins to be exposed on the block.

If Configuration Form Tab Is . . .	Then, use it to . . .
Configuration Parameters 	Configure parameters to be exposed on block face in Project mode.
Monitoring Parameters 	Configure parameters to be exposed on block face in Monitor mode.

If Configuration Form Tab Is . . .	Then, use it to . . .
Block Preferences 	Configure desired block display preferences.
Template Defining 	View/configure parameters to be designated for the template. You must have a template license to use this form.

11.5 BIZERBA ST Parameter Reference

Related topics

“PDDESC[0..5]” on page 177
“STDESC[0..15]” on page 177
“FCTDESC[0..15]” on page 178
“STATINPDESC[0..7]” on page 178
“STATOUTPDESC[0..7]” on page 178
“ACKERR” on page 179
“PD[0..5]” on page 179
“STSTATUS” on page 179
“STBIT[0..15]” on page 180
“FCTSTATUS” on page 180
“FCTBIT[0..15]” on page 180
“STATICINP” on page 181
“STATINPBIT[0..7]” on page 181
“STATICOUT” on page 181
“STATOUTBIT” on page 182
“COMMAND” on page 182
“COMMANDVAL” on page 182
“SENDENB” on page 183

11.5.1 PDDESC[0..5]

Specific to Block(s)	BIZERBASTIN
Description	Parameter Data Description
Data Type	String
Range	24 Characters maximum
Default	
Config Load	Yes
Access Lock	Application Developer Only
Residence	SR
Related Parameters	“PD[0..5]” on page 179
Remarks	Because the PD[x] parameters cannot be renamed, this text descriptor is useful in documenting the use of each PD[x].

11.5.2 STDESC[0..15]

Specific to Block(s)	BIZERBASTIN
Description	ST Description - Describes the ST Bit.
Data Type	String
Range	24 Characters maximum
Default	

Config Load	Yes
Access Lock	Application Developer Only
Residence	SR
Related Parameters	“STBIT[0..15]” on page 180
Remarks	Because the STBIT[x] parameters cannot be renamed, this text descriptor is useful in documenting the use of each STBIT[x].

11.5.3 FCTDESC[0..15]

Specific to Block(s)	BIZERBASTIN
Description	Function Description - Describes Function Bit.
Data Type	String
Range	24 Characters Maximum
Default	
Config Load	Yes
Access Lock	Application Developer Only
Residence	SR
Related Parameters	“FCTBIT[0..15]” on page 180
Remarks	Because the FCTBIT[x] parameters cannot be renamed, this text descriptor is useful in documenting the use of each FCTBIT[x].

11.5.4 STATINPDESC[0..7]

Specific to Block(s)	BIZERBASTIN
Description	Static Input Description - Describes the static input bit.
Data Type	String
Range	24 Characters maximum
Default	
Config Load	Yes
Access Lock	Application Developer Only
Residence	SR
Related Parameters	“STATINPBIT[0..7]” on page 181
Remarks	Because the STATINPBIT[x] parameters cannot be renamed, this text descriptor is useful in documenting the use of each STATINPBIT[x].

11.5.5 STATOUTPDESC[0..7]

Specific to Block(s)	BIZERBASTIN
Description	Static Output Description - Describes the static output bit.
Data Type	String
Range	24 Characters maximum
Default	
Config Load	Yes

Access Lock	Application Developer Only
Residence	SR
Related Parameters	“STATOUTBIT” on page 182
Remarks	Because the STATOUTBIT[x] parameters cannot be renamed, this text descriptor is useful in documenting the use of each STATOUTBIT[x].

11.5.6 ACKERR

Specific to Block(s)	BIZERBASTIN
Description	Function Acknowledge Code or Error Code (see Bizerba manual)
Data Type	Integer 32
Range	0 to 65535
Default	0
Config Load	No
Access Lock	View Only
Residence	CEE
Related Parameters	
Remarks	Parameter ACKERR is a 16bit unsigned integer value received from the device.

11.5.7 PD[0..5]

Specific to Block(s)	BIZERBASTIN
Description	Process Data - Identifies process input weight or other value depending on configuration.
Data Type	Float 64
Range	
Default	0
Config Load	No
Access Lock	View Only
Residence	CEE
Related Parameters	“PDDESC[0..5]” on page 177
Remarks	Parameters PD[x] are 32-bit IEEE values received from the device.

11.5.8 STSTATUS

Specific to Block(s)	BIZERBASTIN
Description	ST Status (See Bizerba manual.)
Data Type	Integer 32
Range	0 to 65535
Default	0
Config Load	No
Access Lock	View Only
Residence	CEE

Related Parameters	“STBIT[0..15]” on page 180
Remarks	Parameter STSTATUS is a 16bit unsigned integer value received from the device.

11.5.9 STBIT[0..15]

Specific to Block(s)	BIZERBASTIN
Description	ST Status Bit - Boolean parameter STBIT[x] corresponds to bit position 'x' of STSTATUS.
Data Type	Boolean
Range	Off (0) On (1)
Default	Off (0)
Config Load	No
Access Lock	View Only
Residence	CEE
Related Parameters	“STSTATUS” on page 179
Remarks	Boolean parameter STBIT[x] corresponds to bit position 'x' of STSTATUS.

11.5.10 FCTSTATUS

Specific to Block(s)	BIZERASTIN
Description	Function Status (See Bizerba manual.)
Data Type	Integer 32
Range	0 to 65535
Default	0
Config Load	No
Access Lock	View Only
Residence	CEE
Related Parameters	“FCTBIT[0..15]” on page 180
Remarks	Parameter FCTSTATUS is a 16bit unsigned integer value received from the device.

11.5.11 FCTBIT[0..15]

Specific to Block(s)	BIZERBASTIN
Description	Function Status Bit - Boolean parameter FCTBIT[x] corresponds to bit position 'x' of FCTSTATUS.
Data Type	Boolean
Range	Off (0) On (1)
Default	Off (0)
Config Load	No
Access Lock	View Onlyk
Residence	CEE

Related Parameters	“FCTSTATUS” on page 180
Remarks	

11.5.12 STATICINP

Specific to Block(s)	BIZERBASTIN
Description	Static Input Status (See Bizerba manual.)
Data Type	Integer 32
Range	0 to 255
Default	0
Config Load	No
Access Lock	View Only
Residence	CEE
Related Parameters	“STATINPBIT[0..7]” on page 181
Remarks	Parameter STATICINP is an 8bit unsigned integer value received from the device.

11.5.13 STATINPBIT[0..7]

Specific to Block(s)	BIZERBASTIN
Description	Static Input Bit
Data Type	Boolean
Range	Off (0) On (1)
Default	Off (0)
Config Load	No
Access Lock	View Only
Residence	CEE
Related Parameters	“STATICINP” on page 181
Remarks	Boolean parameter STATINPBIT[x] corresponds to bit position 'x' of STATICINP.

11.5.14 STATICOUT

Specific to Block(s)	BIZERBASTIN
Description	Static Output (See Bizerba manual.)
Data Type	Integer 32
Range	0 to 255
Default	0
Config Load	No
Access Lock	View Only
Residence	CEE
Related Parameters	“STATOUTBIT” on page 182

Remarks	Parameter STATICOUT is an 8bit unsigned integer value received from the device.

11.5.15 STATOUTBIT

Specific to Block(s)	BIZERBASTIN
Description	Static Output Status Bit
Data Type	Boolean
Range	Off (0) On (1)
Default	Off (0)
Config Load	No
Access Lock	View Only
Residence	CEE
Related Parameters	“STATICOUT” on page 181
Remarks	Boolean parameter STATOUTBIT[x] corresponds to bit position 'x' of STATICOUT.

11.5.16 COMMAND

Specific to Block(s)	BIZERBASTOUT
Description	Command - Function request code.
Data Type	Float 64
Range	
Default	0
Config Load	No
Access Lock	Engineer
Residence	CEE
Related Parameters	“COMMANDVAL” on page 182
Remarks	COMMAND and COMMANDVAL form a consistent request entity.

11.5.17 COMMANDVAL

Specific to Block(s)	BIZERBASTOUT
Description	Command Value - Function Request Value (not required for some requests).
Data Type	Float 64
Range	
Default	0
Config Load	No
Access Lock	Engineer
Residence	CEE
Related Parameters	“COMMAND” on page 182
Remarks	COMMAND and COMMANDVAL form a consistent request entity.

11.5.18 SENDENB

Specific to Block(s)	BIZERBAOUT
Description	Send Enable - Self-resetting send enable flag for a new function request.
Data Type	Boolean
Range	Off (0) On (1)
Default	Off (0)
Config Load	No
Access Lock	Engineer
Residence	CEE
Related Parameters	“COMMAND” on page 182, “COMMANDVAL” on page 182
Remarks	Setting of SENDENB will initiate the transmission of a new Function Request. SENDENB is reset automatically after the first successful transmission of this new Function Request.

12 Simocode 3UF5x Function Blocks

The SIMOCODE-DP (**Si**emens **Mo**tor Protection and **Co**ntrol **De**vice) is a communications-capable motor protection and control device for connecting to the PROFIBUS-DP. With its integrated motor protection functions, it protects three-phase loads (1.25A to 820A) against thermal overload. All protection functions can be parameterized to provide a warning or to switch off the motor.

The blocks described in this section have been developed based upon the 'SIMOCODE-DP System Manual' and can be used with PROFIBUS-DP devices that adhere to it.

The simocode function block set consists of the following blocks:

- Simocode Device Block
- Simocode Input Channel Block
- Simocode Output Channel Block.

Related topics

“Simocode Device Block” on page 186
“Simocode Input Channel Block” on page 190
“Simocode Output Channel Block” on page 206

12.1 Simocode Device Block

Related topics

“Simocode Device Block description” on page 186
“Simocode Device Block Channel Assignment” on page 186
“Simocode Device Block Configuration” on page 186
“Simocode Device Block Operation” on page 189

12.1.1 Simocode Device Block description

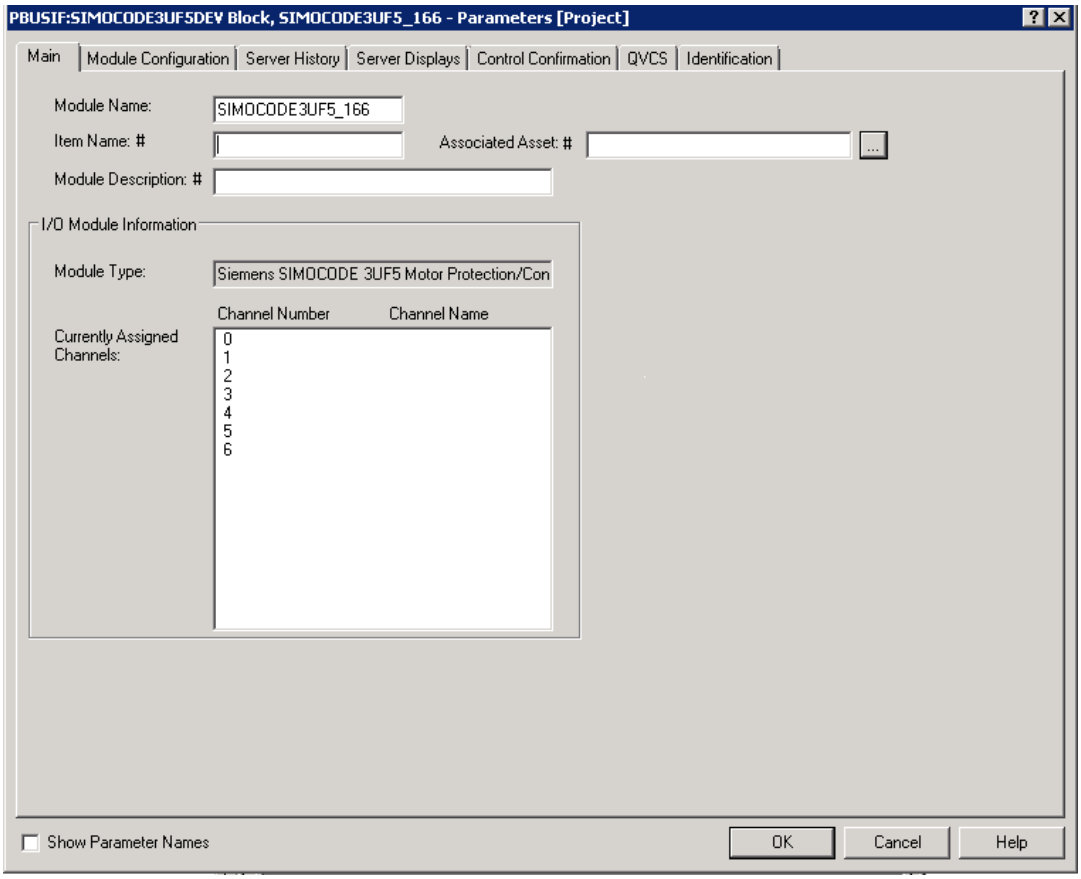
The block template name of the Simocode Device Block is: SIMOCODE3UF5DEV. This block represents the simocode device itself. Process data is made available through input and output channel blocks that are associated with the Simocode Device Block.

12.1.2 Simocode Device Block Channel Assignment

The Simocode Device Block is capable of hosting only two channel blocks. Channel 0 is reserved for a Simocode Input Channel Block, and channel 1 is reserved for assignment by a Simocode Output Channel Block.

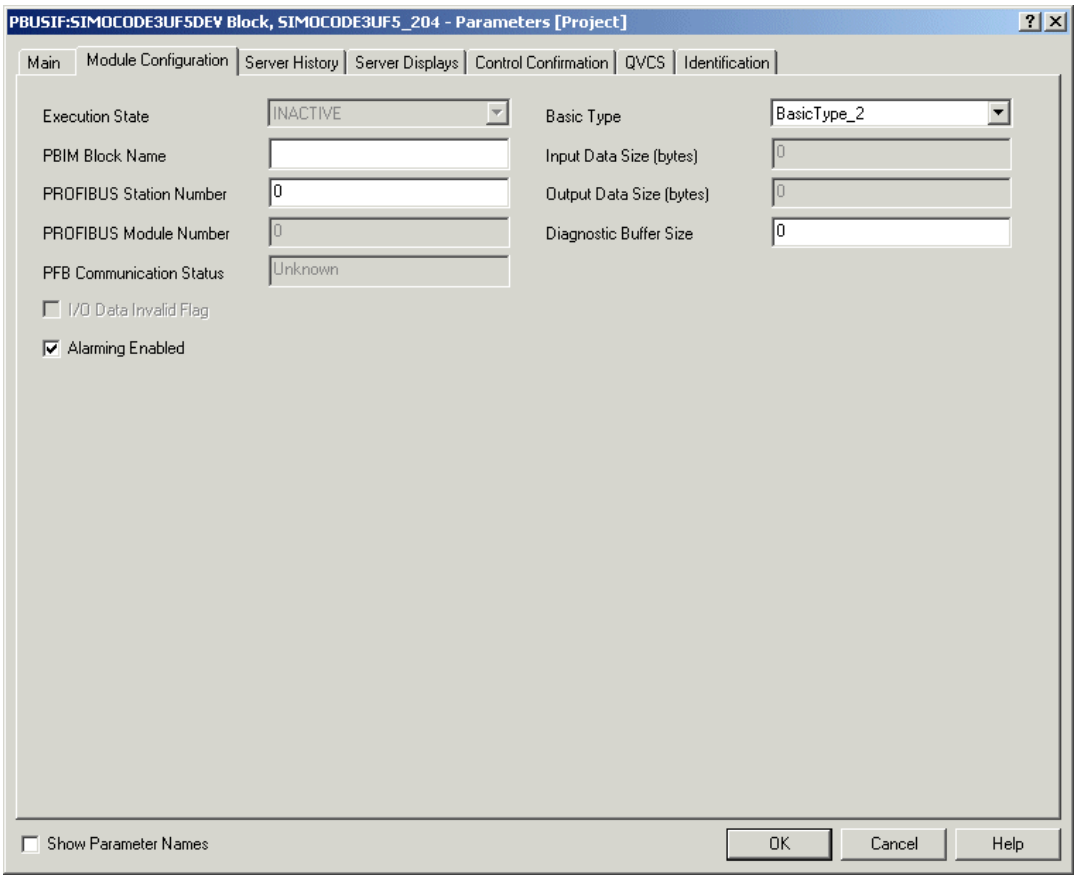
12.1.3 Simocode Device Block Configuration

The Main tab of the Simocode Device block configuration form is as follows.



The Associated Asset field represents an asset selected from those configured in the Enterprise Model Database to set the Scope of Responsibility (SOR) for the point.

The Module Configuration tab for the Simocode Device Block is pictured below.



Notes on the example configuration tab:

- The 'PBIM Block Name' and 'PROFIBUS Station Number' fields represent the SRCBLOCK and STATION parameters respectively. These parameters, as well as the Diagnostic Buffer Size (DIAGBUFSIZE) parameter, are common to all PROFIBUS device/module blocks and are described in the *Commonality of PROFIBUS I/O Device/Module Blocks* Section.
- Because the simocode station supports only a single module, the module number is not configurable, and is set to 0 by default.
- The input and output data sizes are not configurable directly. Instead, these parameters are calculated after block load based upon the selection for 'Basic Type', which is described below in the parameter definition for BASICTYPE.

The only device specific configuration parameter for the Simocode Device Block is BASICTYPE.

Parameter Name:	**BASICTYPE**		
Specific to Block:	SIMOCODE3UF5DEV		
Description:	Selection of Basic Type		
Data Type:	Enumeration of BASETYPE		
Range:	#	*Enum text*	*Description*
	1	BasicType_1	Basic Type 1 selected
	2	BasicType_2	Basic Type 2 selected
	3	BasicType_3	Basic Type 3 selected
Default:	Basic_Type_2		
Config Load:	Yes		

Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	
Remarks:	See the 'SIMOCODE-DP System Manual' for data size and format of the basic types.

12.1.4 Simocode Device Block Operation

Refer to *Operation* in the *Commonality of PROFIBUS I/O Device/Module Blocks* Section, which describes the operational characteristics common to all PROFIBUS device/module blocks.

12.2 Simocode Input Channel Block

Refer to the *Commonality of PROFIBUS I/O Channel Blocks* Section for attributes that are common to all PROFIBUS channel blocks.

Related topics

“Simocode Input Channel Block description” on page 190
“Simocode Input Channel Block Configuration” on page 190
“Motor Current Scaling” on page 190
“Configuration Parameters for Motor Current Scaling” on page 191
“User Specific Descriptions” on page 192
“Configuration Parameters for the Signal Data 0/1 Detail tab” on page 192
“Simocode Input Channel Block Operation” on page 194
“Process Data” on page 194
“Operational Parameters for the Process Data tab” on page 194
“Signal Data Byte 0 and 1 Detail” on page 200
“Operational Parameters for the Signal Data 0/1 Detail tab” on page 201
“Signal Data Byte 2 and 3 Detail” on page 202
“Operational Paramaters for the Signal Data 2/3 Detail tab” on page 203

12.2.1 Simocode Input Channel Block description

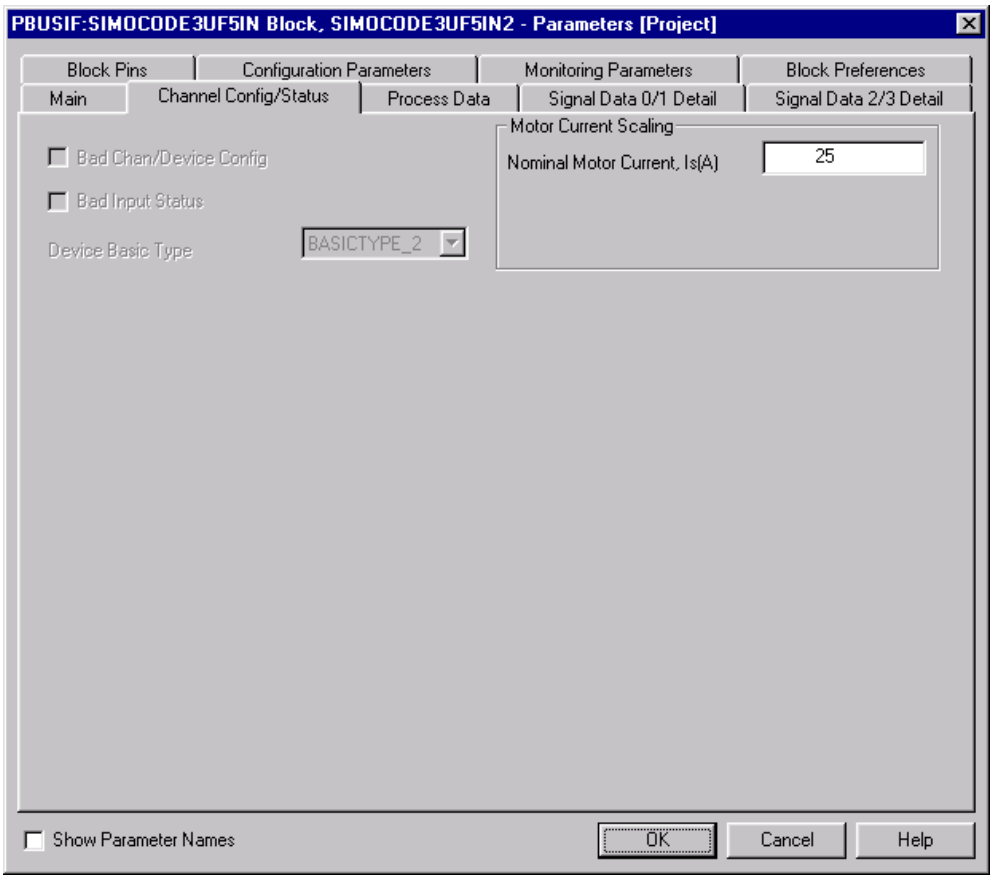
The Simocode Input Channel Block template name is: SIMOCODE3UF5IN. This input channel block is responsible for providing the control interface to all input parameters from the simocode device. The Simocode Input Channel Block can be assigned only to channel 0 of the Simocode Device Block.

12.2.2 Simocode Input Channel Block Configuration

Besides the channel assignment, block execution order, and any desired change in the parameter or pin exposure on the block, the Simocode Input Channel Block requires the device specific configuration parameter MOTORCURNOM and the user specific descriptions SD0DESC and SD1DESC - and by using the base type 3: SD2DESC and SD3DESC - of the configured signaling data.

12.2.3 Motor Current Scaling

The Basic Type / Scaling tab for the Simocode Input Channel Block is pictured below.



Notes on the example configuration tab:

- The BADCONFIGFL and BASICTYPE are not configurable. Instead, these parameters are calculated after block load based upon the selection for 'Basic Type' of the assigned Simocode Device Block.
- MOTORCURNOM is the value (in ampere) corresponding to 100% motor current. The valid range is: 0 A, 1.25 A <= value <= 820 A.

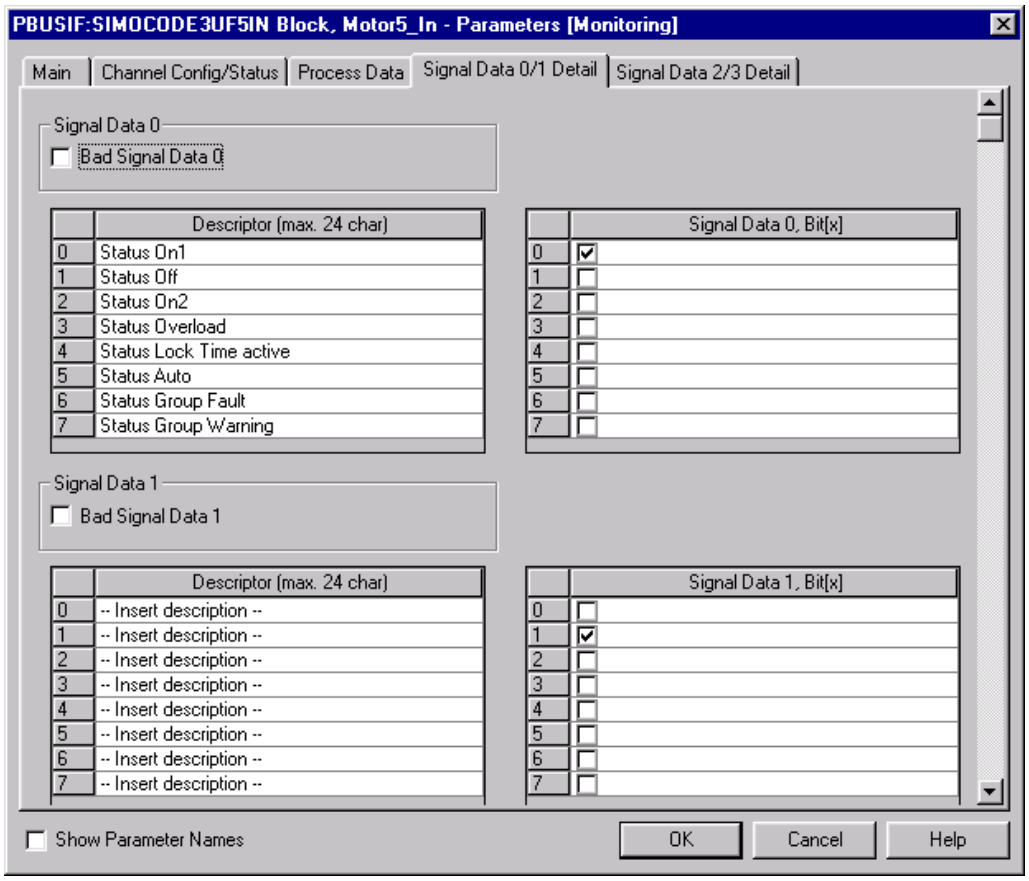
12.2.4 Configuration Parameters for Motor Current Scaling

The only configuration parameter for motor current scaling is MOTORCURNOM:

Parameter Name:	**MOTORCURNOM**
Specific to Block:	SIMOCODE3UF5IN
Description:	Nominal Motor Current
Data Type:	FLOAT64
Range:	0, 1.25 .. 820
Default:	0
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	MOTORCURRENT
Remarks:	Values between 0 and 1.25 are not allowed!

12.2.5 User Specific Descriptions

The Signal Data 0/1 Detail tab for the Simocode Input Channel Block is pictured below.



Notes on the example configuration tab:

- The Bad PV Signal Data 0 and 1 as well as SD0BITS(x) and SD1BITS(x) are operational parameters and are not configurable.
- The meaning/interpretation of the signal data of the simocode device is freely configurable. The text descriptors SD0DESC and SD1DESC may help to document the use of each bit.
- The Signal Data 2/3 Detail tab (only used with basic type 3) works in the same way as the configuration tab above.

12.2.6 Configuration Parameters for the Signal Data 0/1 Detail tab

The following parameters are the configuration parameters for user specific descriptions:

Parameter Name:	**SD0DESC[0..7]**
Specific to Block:	SIMOCODE3UF5IN
Description:	Descriptor of SD0BITS[0..7]
Data Type:	24 Character Text String
Range:	
Default:	'-- Insert description --'

Config Load:	Yes
Access Lock:	Application Developer
Residence:	SR
Related Parameters:	SD0BITS[..]
Remarks:	Because the SD0BITS[x] parameter cannot be renamed, this text descriptor is useful in documenting the use of each SD0BITS[x].

Parameter Name:	**SD1DESC[0..7]**
Specific to Block:	SIMOCODE3UF5IN
Description:	Descriptor of SD1BITS[0..7]
Data Type:	24 Character Text String
Range:	
Default:	'-- Insert description --'
Config Load:	Yes
Access Lock:	Application Developer
Residence:	SR
Related Parameters:	SD1BITS[..]
Remarks:	Because the SD1BITS[x] parameter cannot be renamed, this text descriptor is useful in documenting the use of each SD1BITS[x].

Parameter Name:	**SD2DESC[0..7]**
Specific to Block:	SIMOCODE3UF5IN
Description:	Descriptor of SD2BITS[0..7]
Data Type:	24 Character Text String
Range:	
Default:	'-- Insert description --'
Config Load:	Yes
Access Lock:	Application Developer
Residence:	SR
Related Parameters:	SD2BITS[..]
Remarks:	Because the SD2BITS[x] parameter cannot be renamed, this text descriptor is useful in documenting the use of each SD2BITS[x].

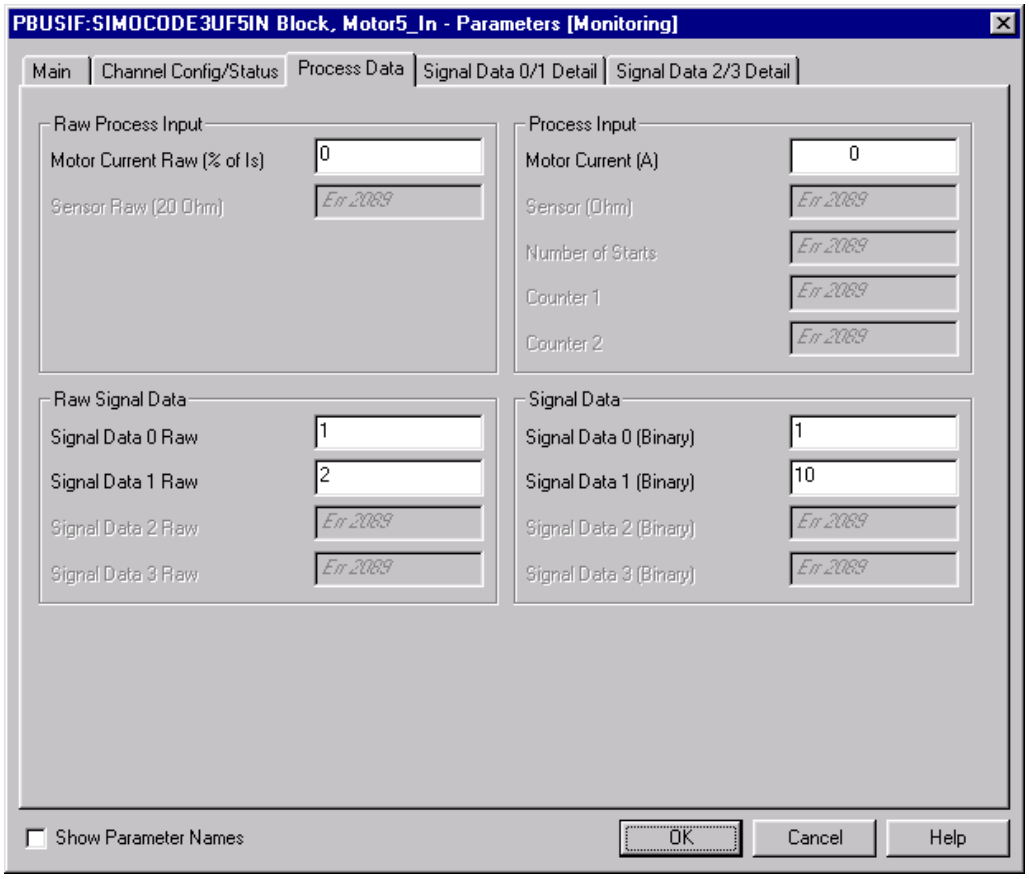
Parameter Name:	**SD3DESC[0..7]**
Specific to Block:	SIMOCODE3UF5IN
Description:	Descriptor of SD3BITS[0..7]
Data Type:	24 Character Text String
Range:	
Default:	'-- Insert description --'
Config Load:	Yes
Access Lock:	Application Developer
Residence:	SR
Related Parameters:	SD3BITS[..]

Remarks:	Because the SD3BITS[x] parameter cannot be renamed, this text descriptor is useful in documenting the use of each SD3BITS[x] .

12.2.7 Simocode Input Channel Block Operation

12.2.8 Process Data

The Process Data tab of the Simocode Input Channel Block is pictured below.



Notes on the example tab:

- This tab shows the simocode input parameters according to basic type 2. Parameters marked with *Err 2089*are not supported by the current basic type!
- The left hand side group boxes contain the raw field data, whereas the calculated values are shown in the right site.
- Please note the binary indication of signal data. This kind of presentation makes it easier to detect the separate signal data bits. Unfortunately, it is not possible to show leading '0's!. Thus, it is recommended that the user reads the binary data from right to left. The first digit from right represents bit 0, the second digit from right represents bit 1, and so on up to bit 7. Note: In the Signal Data x/y Detail tabs, the separate signal data bits with their user specific descriptions are displayed.

12.2.9 Operational Parameters for the Process Data tab

The following parameters describe the operational parameters of the Simocode Input Channel Block (Process Data tab).

Parameter Name:	**MOTORCURRAW**
Specific to Block:	SIMOCODE3UF5IN
Description:	Motor current [% of MOTORCURNOM], raw value This value is the raw field value that is transported via PROFIBUS. It is scaled by the block to calculate MOTORCURRENT.
Data Type:	INT32
Range:	
Default:	0
Config Load:	No
Access Lock:	View only
Residence:	CEE
Related Parameters:	MOTORCURRENT
Remarks:	This is the maximum motor current in steps of 1%. It is only available in basic type 1 and 2. In basic type 3, it is marked with *Err 2089*. See the 'SIMOCODE-DP System Manual' for details.

Parameter Name:	**MOTORCURRENT**
Specific to Block:	SIMOCODE3UF5IN
Description:	Motor current [A] This parameter represents the maximum motor current in ampere.
Data Type:	FLOAT64
Range:	
Default:	0.0
Config Load:	No
Access Lock:	View only
Residence:	CEE
Related Parameters:	MOTORCURRAW, MOTORCURNOM
Remarks:	This is MOTORCURRAW % of MOTORCURNOM. It is only available in basic type 1 and 2. In basic type 3, it is marked with *Err 2089*. See the 'SIMOCODE-DP System Manual' for details.

Parameter Name:	**SENSORRAW**
Specific to Block:	SIMOCODE3UF5IN
Description:	Sensor value [20 ohm] This value is the raw field value that is transported through PROFIBUS. It is scaled by the block to calculate SENSOR.
Data Type:	INT32
Range:	0..255
Default:	0
Config Load:	No
Access Lock:	View only
Residence:	CEE
Related Parameters:	SENSOR

Remarks:	This is the sensor value in steps of 20 ohm. It is only available in basic type 1. In basic type 2 and 3 it is marked with *Err 2089*. See the 'SIMOCODE-DP System Manual' for details.

Parameter Name:	**SENSOR**
Specific to Block:	SIMOCODE3UF5IN
Description:	Sensor value [ohm] This parameter represents the sensor value in ohm.
Data Type:	FLOAT64
Range:	0..5110
Default:	0
Config Load:	No
Access Lock:	View only
Residence:	CEE
Related Parameters:	SENSORRAW
Remarks:	This is SENSORRAW * 20 ohm. It is only available in basic type 1. In basic type 2 and 3, it is marked with *Err 2089*. See the 'SIMOCODE-DP System Manual' for details.

Parameter Name:	**NUMSTARTS**
Specific to Block:	SIMOCODE3UF5IN
Description:	Number of starts.
Data Type:	INT32
Range:	0..16 777 215
Default:	0
Config Load:	No
Access Lock:	View only
Residence:	CEE
Related Parameters:	
Remarks:	This parameter is only available in basic type 1. In basic type 2 and 3, it is marked with *Err 2089*. See the 'SIMOCODE-DP System Manual' for details.

Parameter Name:	**COUNTER1**
Specific to Block:	SIMOCODE3UF5IN
Description:	Value of counter1.
Data Type:	FLOAT64
Range:	0..65 535
Default:	0
Config Load:	No
Access Lock:	View only
Residence:	CEE
Related Parameters:	

Remarks:	This parameter is only available in basic type 1. In basic type 2 and 3, it is marked with *Err 2089.* See the 'SIMOCODE-DP System Manual' for details.

Parameter Name:	**COUNTER2**
Specific to Block:	SIMOCODE3UF5IN
Description:	Value of counter2.
Data Type:	FLOAT64
Range:	0..65 535
Default:	0
Config Load:	No
Access Lock:	View only
Residence:	CEE
Related Parameters:	
Remarks:	This parameter is only available in basic type 1. In basic type 2 and 3, it is marked with *Err 2089.* See the 'SIMOCODE-DP System Manual' for details.

Parameter Name:	**SD0RAW**
Specific to Block:	SIMOCODE3UF5IN
Description:	Signal Data Byte 0 This value is the raw field value that is transported via PROFIBUS. It is converted by the block to SD0BIN and SD0BITS[x].
Data Type:	INT32
Range:	0..255
Default:	0
Config Load:	No
Access Lock:	View only
Residence:	CEE
Related Parameters:	SD0BIN, SD0BITS[x]
Remarks:	This parameter is available in all 3 basic types. See the 'SIMOCODE-DP System Manual' for details.

Parameter Name:	**SD0BIN**
Specific to Block:	SIMOCODE3UF5IN
Description:	Signal Data Byte 0 in binary indication.
Data Type:	INT32
Range:	
Default:	0
Config Load:	No
Access Lock:	View only
Residence:	CEE
Related Parameters:	SD0RAW, SD0BITS[x]

Remarks:	This parameter represents the signal data byte 0 in specifically binary format (without leading '0's). The first digit from right represents bit 0.0, the second digit from right represents bit 0.1, and so on up to bit 0.7. Digit '0' represents state 'OFF', digit '1' state 'ON'. This parameter is available in all 3 basic types.

Parameter Name:	**SD1RAW**
Specific to Block:	SIMOCODE3UF5IN
Description:	Signal Data Byte 1 This value is the raw field value that is transported via PROFIBUS. It is converted by the block to SD1BIN and SD1BITS[x].
Data Type:	INT32
Range:	0..255
Default:	0
Config Load:	No
Access Lock:	View only
Residence:	CEE
Related Parameters:	SD1BIN, SD1BITS[x]
Remarks:	This parameter is available in all 3 basic types. See the 'SIMOCODE-DP System Manual' for details.

Parameter Name:	**SD1BIN**
Specific to Block:	SIMOCODE3UF5IN
Description:	Signal Data Byte 1 in binary indication.
Data Type:	INT32
Range:	
Default:	0
Config Load:	No
Access Lock:	View only
Residence:	CEE
Related Parameters:	SD1RAW, SD1BITS[x]
Remarks:	This parameter represents the signal data byte 1 in specifically binary format (without leading '0's). The first digit from right represents bit 1.0, the second digit from right represents bit 1.1, and so on up to bit 1.7. Digit '0' represents state 'OFF', digit '1' state 'ON'. This parameter is available in all 3 basic types.

Parameter Name:	**SD2RAW**
Specific to Block:	SIMOCODE3UF5IN
Description:	Signal Data Byte 2 This value is the raw field value that is transported via PROFIBUS. It is converted by the block to SD2BIN and SD2BITS[x].
Data Type:	INT32
Range:	0..255

Default:	0
Config Load:	No
Access Lock:	View only
Residence:	CEE
Related Parameters:	SD2BIN, SD2BITS[x]
Remarks:	This parameter is only available in basic type 3. In basic type 1 and 2 it is marked with *Err 2089.* See the 'SIMOCODE-DP System Manual' for details.

Parameter Name:	**SD2BIN**
Specific to Block:	SIMOCODE3UF5IN
Description:	Signal Data Byte 2 in binary indication.
Data Type:	INT32
Range:	
Default:	0
Config Load:	No
Access Lock:	View only
Residence:	CEE
Related Parameters:	SD2RAW, SD2BITS[x]
Remarks:	This parameter represents the signal data byte 2 in specifically binary format (without leading '0's). The first digit from right represents bit 2.0, the second digit from right represents bit 2.1, and so on up to bit 2.7. Digit '0' represents state 'OFF', digit '1' state 'ON'. This parameter is only available in basic type 3. In basic type 1 and 2 it is marked with *Err 2089.*

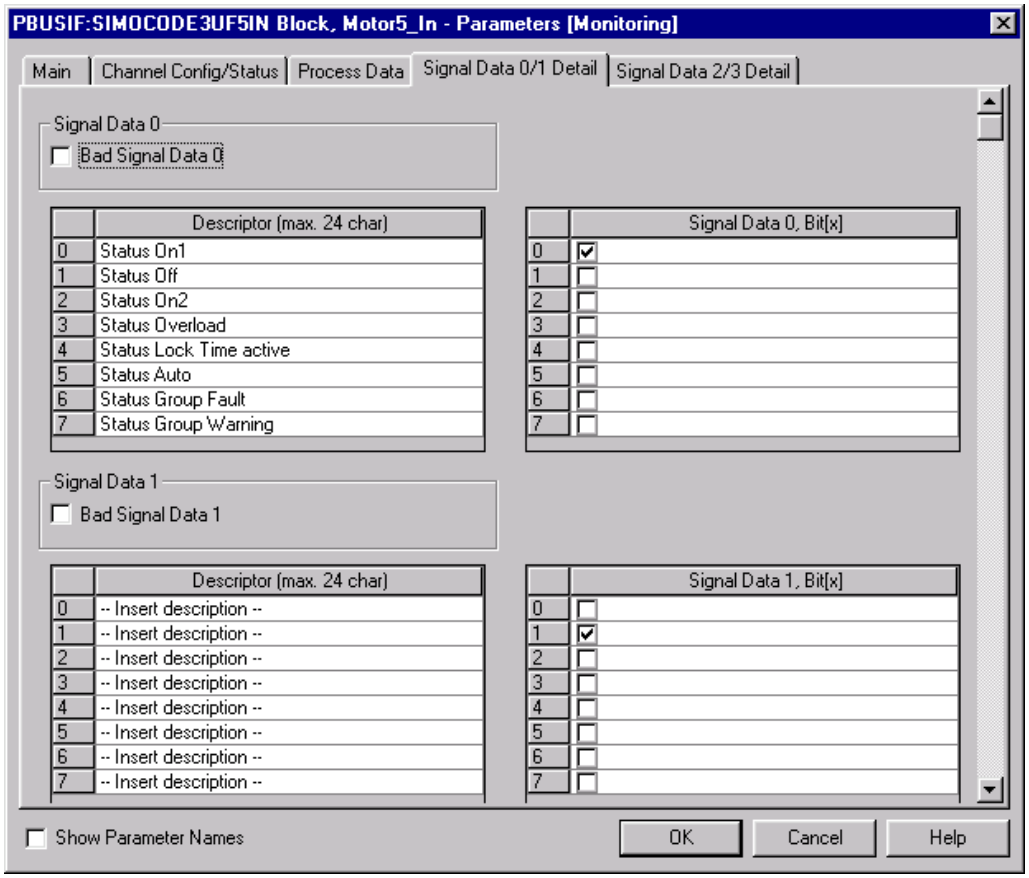
Parameter Name:	**SD3RAW**
Specific to Block:	SIMOCODE3UF5IN
Description:	Signal Data Byte 3 This value is the raw field value that is transported via PROFIBUS. It is converted by the block to SD3BIN and SD3BITS[x].
Data Type:	INT32
Range:	0..255
Default:	0
Config Load:	No
Access Lock:	View only
Residence:	CEE
Related Parameters:	SD3BIN, SD3BITS[x]
Remarks:	This parameter is only available in basic type 3. In basic type 1 and 2 it is marked with *Err 2089.* See the 'SIMOCODE-DP System Manual' for details.

Parameter Name:	**SD3BIN**
Specific to Block:	SIMOCODE3UF5IN

Description:	Signal Data Byte 3 in binary indication.
Data Type:	INT32
Range:	
Default:	0
Config Load:	No
Access Lock:	View only
Residence:	CEE
Related Parameters:	SD3RAW, SD3BITS[x]
Remarks:	This parameter represents the signal data byte 3 in specifically binary format (without leading '0's). The first digit from right represents bit 3.0, the second digit from right represents bit 3.1, and so on up to bit 3.7. Digit '0' represents state 'OFF', digit '1' state 'ON'. This parameter is only available in basic type 3. In basic type 1 and 2 it is marked with *Err 2089*.

12.2.10 Signal Data Byte 0 and 1 Detail

The Signal. Data 0/1 Detail tab of the Simocode Input Channel Block is pictured below.



Notes on the example tab:

- This tab shows the Bad PV flag, the user specific descriptions and the status bits of signal data byte 0 and 1.
- Bit 0.0 and bit 1.1 are set to 'ON', the others are set to 'OFF'. The Bad PV flags are set to 'OFF', that means, the input data is valid.

12.2.11 Operational Parameters for the Signal Data 0/1 Detail tab

The following parameters describe the operational parameters of the Simocode Input Channel Block (Signal Data 0/1 Detail tab).

Parameter Name:	**SD0BAD**	
Specific to Block:	SIMOCODE3UF5IN	
Description:	Bad Communication Status Indicator for SD0RAW, SD0BIN and SD0BITS[x]	
Data Type:	BOOLEAN	
Range:	OFF(0)	Good status
	ON(1)	Bad status
Default:	ON(1)	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	SD0RAW, SD0BIN, SD0BITS[x]	
Remarks:	This bad status indicator will set to 'ON' if the channel block is not actively receiving valid input data from the input device for any reason, specifically if either 1) there is no connection to PBIM module or 2) simocode device is not online. This parameter is available in all 3 basic types.	

Parameter Name:	**SD0BITS[0..7]**
Specific to Block:	SIMOCODE3UF5IN
Description:	Signal Data bit 0.0 to bit 0.7.
Data Type:	BOOLEAN
Range:	OFF(0) ON(1)
Default:	OFF
Config Load:	No
Access Lock:	View only
Residence:	CEE
Related Parameters:	SD0RAW, SD0BIN
Remarks:	This parameter represents the Boolean (flag) indication of bit 0.0 to bit 0.7. It is available in all 3 basic types.

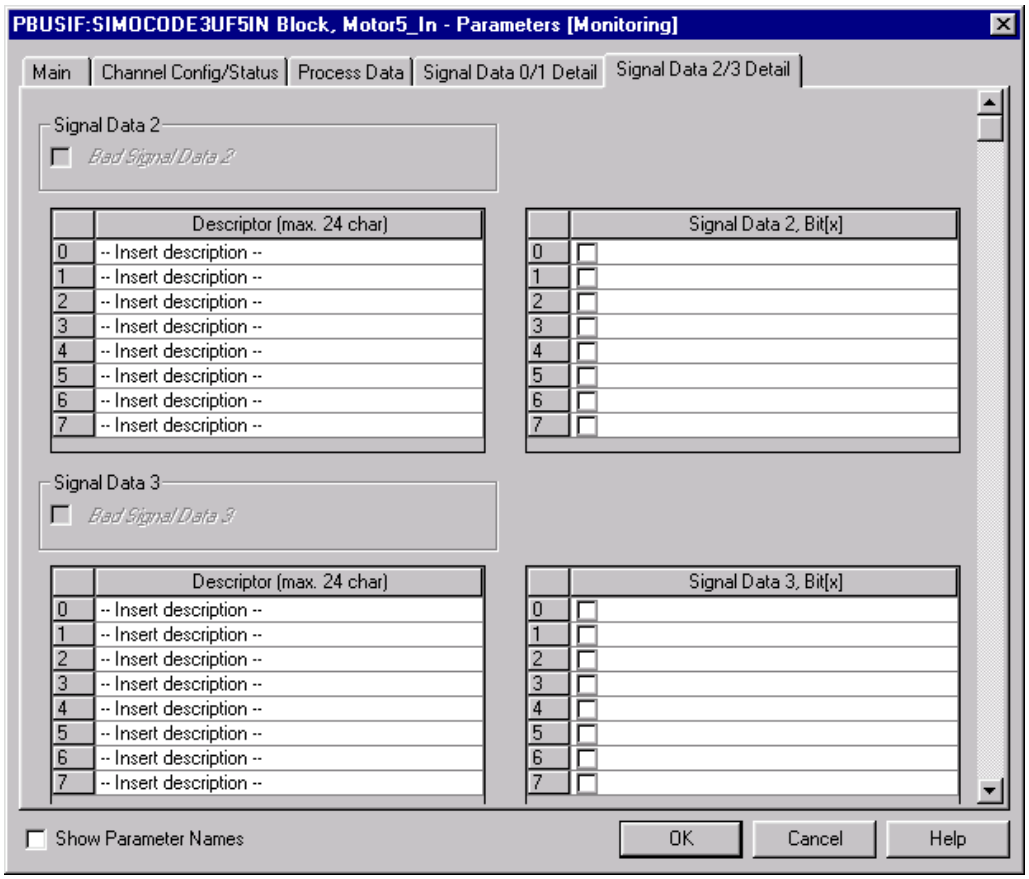
Parameter Name:	**SD1BAD**	
Specific to Block:	SIMOCODE3UF5IN	
Description:	Bad Communication Status Indicator for SD1RAW, SD1BIN and SD1BITS[x]	
Data Type:	BOOLEAN	
Range:	OFF(0)	Good status
	ON(1)	Bad status
Default:	ON(1)	
Config Load:	No	

Access Lock:	View Only
Residence:	CEE
Related Parameters:	SD0RAW, SD0BIN, SD0BITS[x]
Remarks:	This bad status indicator will set to 'ON' if the channel block is not actively receiving valid input data from the input device for any reason, specifically if either 1) there is no connection to PBIM module or 2) simocode device is not online. This parameter is available in all 3 basic types.

Parameter Name:	**SD1BITS[0..7]**
Specific to Block:	SIMOCODE3UF5IN
Description:	Signal Data bit 1.0 to bit 1.7.
Data Type:	BOOLEAN
Range:	OFF(0) ON(1)
Default:	OFF
Config Load:	No
Access Lock:	View only
Residence:	CEE
Related Parameters:	SD0RAW, SD0BIN
Remarks:	This parameter represents the Boolean (flag) indication of bit 1.0 to bit 1.7. It is available in all 3 basic types.

12.2.12 Signal Data Byte 2 and 3 Detail

The Signal. Data 2/3 Detail tab of the Simocode Input Channel Block is pictured below.



Notes on the example tab:

- This tab shows the Bad PV flag, the user specific descriptions and the status bits of signal data byte 2 and 3. These Parameters are only available in basic type 3. The screenshot above is made with a present basic type 2. That is why the unavailable Bad PV flag descriptions are indicated in *gray italic*.

12.2.13 Operational Paramaters for the Signal Data 2/3 Detail tab

The following parameters describe the operational parameters of the Simocode Input Channel Block (Signal. Data 2/3 Detail tab).

Parameter Name:	**SD2BAD**	
Specific to Block:	SIMOCODE3UF5IN	
Description:	Bad Communication Status Indicator for SD2RAW, SD2BIN and SD2BITS[x]	
Data Type:	BOOLEAN	
Range:	OFF(0)	Good status
	ON(1)	Bad status
Default:	ON(1)	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	SD2RAW, SD2BIN, SD2BITS[x]	

Remarks:	This bad status indicator will set to 'ON' if the channel block is not actively receiving valid input data from the input device for any reason, specifically if either 1) there is no connection to PBIM module or 2) simocode device is not online. This parameter is only available in basic type 3. In basic type 1 and 2 its description is marked in gray italic.

Parameter Name:	**SD2BITS[0..7]**
Specific to Block:	SIMOCODE3UF5IN
Description:	Signal Data bit 2.0 to bit 2.7.
Data Type:	BOOLEAN
Range:	OFF(0) ON(1)
Default:	OFF
Config Load:	No
Access Lock:	View only
Residence:	CEE
Related Parameters:	SD2RAW, SD2BIN
Remarks:	This parameter represents the Boolean (flag) indication of bit 2.0 to bit 2.7. It is only available in basic type 3.

Parameter Name:	**SD3BAD**	
Specific to Block:	SIMOCODE3UF5IN	
Description:	Bad Communication Status Indicator for SD3RAW, SD3BIN and SD3BITS[x]	
Data Type:	BOOLEAN	
Range:	OFF(0)	Good status
	ON(1)	Bad status
Default:	ON(1)	
Config Load:	No	
Access Lock:	View Only	
Residence:	CEE	
Related Parameters:	SD3RAW, SD3BIN, SD3BITS[x]	
Remarks:	This bad status indicator will set to 'ON' if the channel block is not actively receiving valid input data from the input device for any reason, specifically if either 1) there is no connection to PBIM module or 2) simocode device is not online. This parameter is only available in basic type 3. In basic type 1 and 2, its description is marked in gray italic.	

Parameter Name:	**SD3BITS[0..7]**
Specific to Block:	SIMOCODE3UF5IN
Description:	Signal Data bit 3.0 to bit 3.7.
Data Type:	BOOLEAN
Range:	OFF(0) ON(1)
Default:	OFF

Config Load:	No
Access Lock:	View only
Residence:	CEE
Related Parameters:	SD3RAW, SD3BIN
Remarks:	This parameter represents the Boolean (flag) indication of bit 3.0 to bit 3.7. It is only available in basic type 3.

12.3 Simocode Output Channel Block

Refer to the *Commonality of PROFIBUS I/O Channel Blocks* Section for attributes, which are common to all PROFIBUS channel blocks.

Related topics

“Simocode Output Channel Block description” on page 206
“Simocode Output Channel Block Configuration” on page 206
“Configuration Parameters for User Specific Descriptions” on page 207
“Initialization Parameters” on page 209
“Simocode Output Channel Block Operation” on page 212
“Control Data” on page 212
“Operational Parameters for the Control Data tab” on page 213
“Control Data Byte 0 and 1 Detail” on page 216
“Operational Parameters for the Control Data 0/1 Detail tab” on page 216
“Control Data Byte 2 and 3 Detail” on page 217

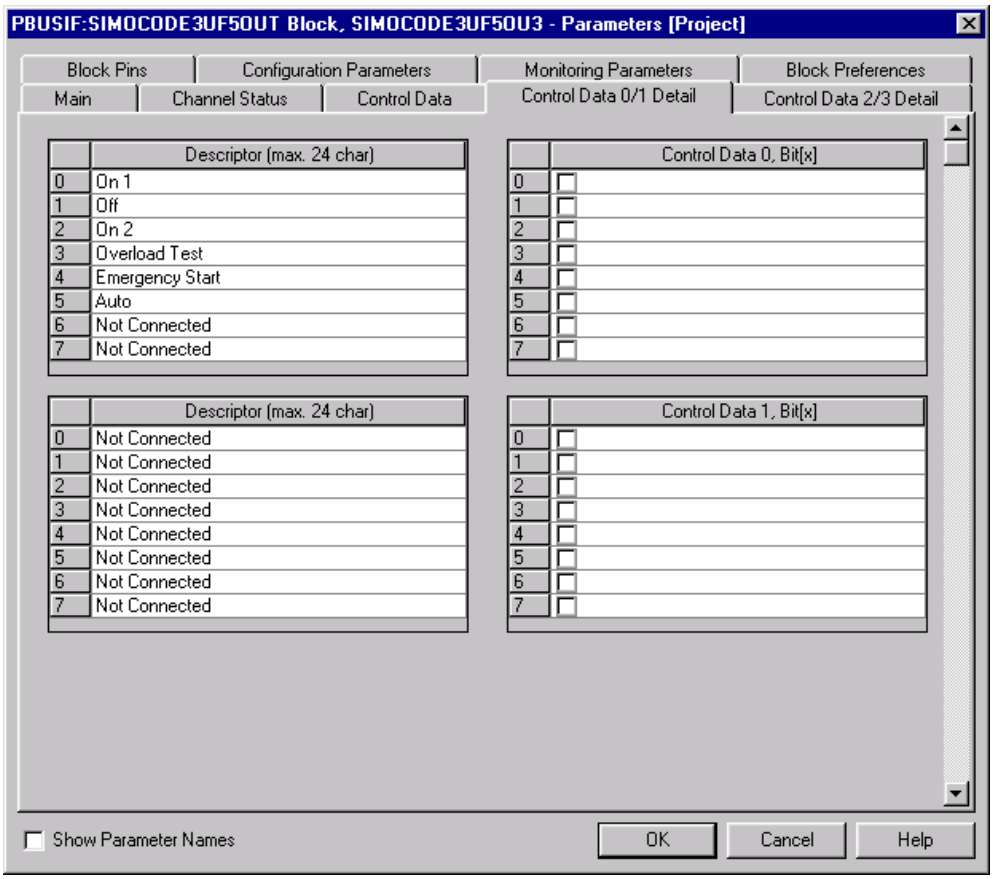
12.3.1 Simocode Output Channel Block description

The Simocode Output Channel Block template name is: SIMOCODE3UF5OUT. This output channel block is responsible for providing the control interface to all output parameters to the simocode device. The Simocode Output Channel Block can be assigned only to channel 1 of the Simocode Device Block.

12.3.2 Simocode Output Channel Block Configuration

Besides the channel assignment, block execution order, and any desired change in the parameter or pin exposure on the block, the Simocode Output Channel Block requires the user specific descriptions CD0DESC, CD1DESC, CD2DESC and SD3DESC of the configured control data.

The Control Data 0/1 Detail tab of the Simocode Output Channel Block is pictured below.



Notes on the example tab:

- Parameters CD0BITS(x) and CD1BITS(X) are operational parameters and are not configurable.
- The meaning/interpretation of the control data of the simocode device is freely configurable. Text descriptors CD0DESC and CD1DESC may help to document the use of each bit.
- The Control Data 2/3 Detail tab works in the same way as the configuration tab above.

12.3.3 Configuration Parameters for User Specific Descriptions

The following parameters allow configuration of user specific descriptors for the Simocode Output Channel Block (Control Data 0/1 Detail tab and Control Data 2/3 Detail tab).

Parameter Name:	**CD0DESC[0..7]**
Specific to Block:	SIMOCODE3UF5OUT
Description:	Descriptor of CD0BITS[0..7]
Data Type:	24 Character Text String
Range:	
Default:	'-- Insert description --'
Config Load:	Yes
Access Lock:	Application Developer
Residence:	SR
Related Parameters:	CD0BITS[..]

Remarks:	Because the CD0BITS[x] parameter cannot be renamed, this text descriptor is useful in documenting the use of each CD0BITS[x]. Available in all 3 basic types.

Parameter Name:	**CD1DESC[0..7]**
Specific to Block:	SIMOCODE3UF5OUT
Description:	Descriptor of CD1BITS[0..7]
Data Type:	24 Character Text String
Range:	
Default:	'-- Insert description --'
Config Load:	Yes
Access Lock:	Application Developer
Residence:	SR
Related Parameters:	CD1BITS[..]
Remarks:	Because the CD1BITS[x] parameter cannot be renamed, this text descriptor is useful in documenting the use of each CD1BITS[x]. Available in all 3 basic types.

Parameter Name:	**CD2DESC[0..7]**
Specific to Block:	SIMOCODE3UF5OUT
Description:	Descriptor of CD2BITS[0..7]
Data Type:	24 Character Text String
Range:	
Default:	'-- Insert description --'
Config Load:	Yes
Access Lock:	Application Developer
Residence:	SR
Related Parameters:	CD2BITS[..]
Remarks:	Because the CD2BITS[x] parameter cannot be renamed, this text descriptor is useful in documenting the use of each CD2BITS[x]. Available in all 3 basic types.

Parameter Name:	**CD3DESC[0..7]**
Specific to Block:	SIMOCODE3UF5OUT
Description:	Descriptor of CD3BITS[0..7]
Data Type:	24 Character Text String
Range:	
Default:	'-- Insert description --'
Config Load:	Yes
Access Lock:	Application Developer
Residence:	SR
Related Parameters:	CD3BITS[..]

Remarks:	Because the CD3BITS[x] parameter cannot be renamed, this text descriptor is useful in documenting the use of each CD3BITS[x]. Available in all 3 basic types.

12.3.4 Initialization Parameters

The following parameters are used to set the behavior of the Simocode Output Channel Block in case of losing the communication between controller and device (Control Data 0/1 Detail tab and Control Data 2/3 Detail tab).

Parameter Name:	**CD0INITMODE[0..7]**	
Specific to Block:	SIMOCODE3UF5OUT	
Description:	Latch to init state	
Data Type:	BOOLEAN	
Range:	OFF(0)	Disabled
	ON(1)	Enabled
Default:	OFF(0)	
Config Load:	Yes	
Access Lock:	Application Developer	
Residence:	CEE	
Related Parameters:	CD0BITS[0..7], CD0INITVALUE[0..7]	
Remarks:	If enabled: In case of losing the communication between controller and device the corresponding output bit CD0BITS[0..7] will be set to the related CD0INITVALUE[0..7]. Available in all 3 basic types.	

Parameter Name:	**CD0INITVALUE[0..7]**
Specific to Block:	SIMOCODE3UF5OUT
Description:	Init state
Data Type:	BOOLEAN
Range:	OFF(0)
	ON(1)
Default:	OFF(0)
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	CD0BITS[0..7], CD0INITMODE[0..7]
Remarks:	If the corresponding CD0INITMODE[0..7] is enabled: In case of losing the communication between controller and device the corresponding output bit CD0BITS[0..7] will be set to the related CD0INITVALUE[0..7]. Available in all 3 basic types.

Parameter Name:	**CD1INITMODE[0..7]**
Specific to Block:	SIMOCODE3UF5OUT
Description:	Latch to init state
Data Type:	BOOLEAN

Range:	OFF(0)	Disabled
	ON(1)	Enabled
Default:	OFF(0)	
Config Load:	Yes	
Access Lock:	Application Developer	
Residence:	CEE	
Related Parameters:	CD1BITS[0..7], CD1INITVALUE[0..7]	
Remarks:	If enabled: In case of losing the communication between controller and device the corresponding output bit CD1BITS[0..7] will be set to the related CD1INITVALUE[0..7]. Available in all 3 basic types.	

Parameter Name:	**CD1INITVALUE[0..7]**
Specific to Block:	SIMOCODE3UF5OUT
Description:	Init state
Data Type:	BOOLEAN
Range:	OFF(0)
	ON(1)
Default:	OFF(0)
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	CD1BITS[0..7], CD1INITMODE[0..7]
Remarks:	If the corresponding CD1INITMODE[0..7] is enabled: In case of losing the communication between controller and device the corresponding output bit CD1BITS[0..7] will be set to the related CD1INITVALUE[0..7]. Available in all 3 basic types.

Parameter Name:	**CD2INITMODE[0..7]**	
Specific to Block:	SIMOCODE3UF5OUT	
Description:	Latch to init state	
Data Type:	BOOLEAN	
Range:	OFF(0)	Disabled
	ON(1)	Enabled
Default:	OFF(0)	
Config Load:	Yes	
Access Lock:	Application Developer	
Residence:	CEE	
Related Parameters:	CD2BITS[0..7], CD2INITVALUE[0..7]	
Remarks:	If enabled: In case of losing the communication between controller and device the corresponding output bit CD2BITS[0..7] will be set to the related CD2INITVALUE[0..7]. Available in all 3 basic types.	

Parameter Name:	**CD2INITVALUE[0..7]**

Specific to Block:	SIMOCODE3UF5OUT
Description:	Init state
Data Type:	BOOLEAN
Range:	OFF(0)
	ON(1)
Default:	OFF(0)
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	CD2BITS[0..7], CD2INITMODE[0..7]
Remarks:	If the corresponding CD2INITMODE[0..7] is enabled: In case of losing the communication between controller and device the corresponding output bit CD2BITS[0..7] will be set to the related CD2INITVALUE[0..7]. Available in all 3 basic types.

Parameter Name:	**CD3INITMODE[0..7]**	
Specific to Block:	SIMOCODE3UF5IN	
Description:	Latch to init state	
Data Type:	BOOLEAN	
Range:	OFF(0)	Disabled
	ON(1)	Enabled
Default:	OFF(0)	
Config Load:	Yes	
Access Lock:	Application Developer	
Residence:	CEE	
Related Parameters:	CD3BITS[0..7], CD3INITVALUE[0..7]	
Remarks:	If enabled: In case of losing the communication between controller and device the corresponding output bit CD3BITS[0..7] will be set to the related CD3INITVALUE[0..7]. Available in all 3 basic types.	

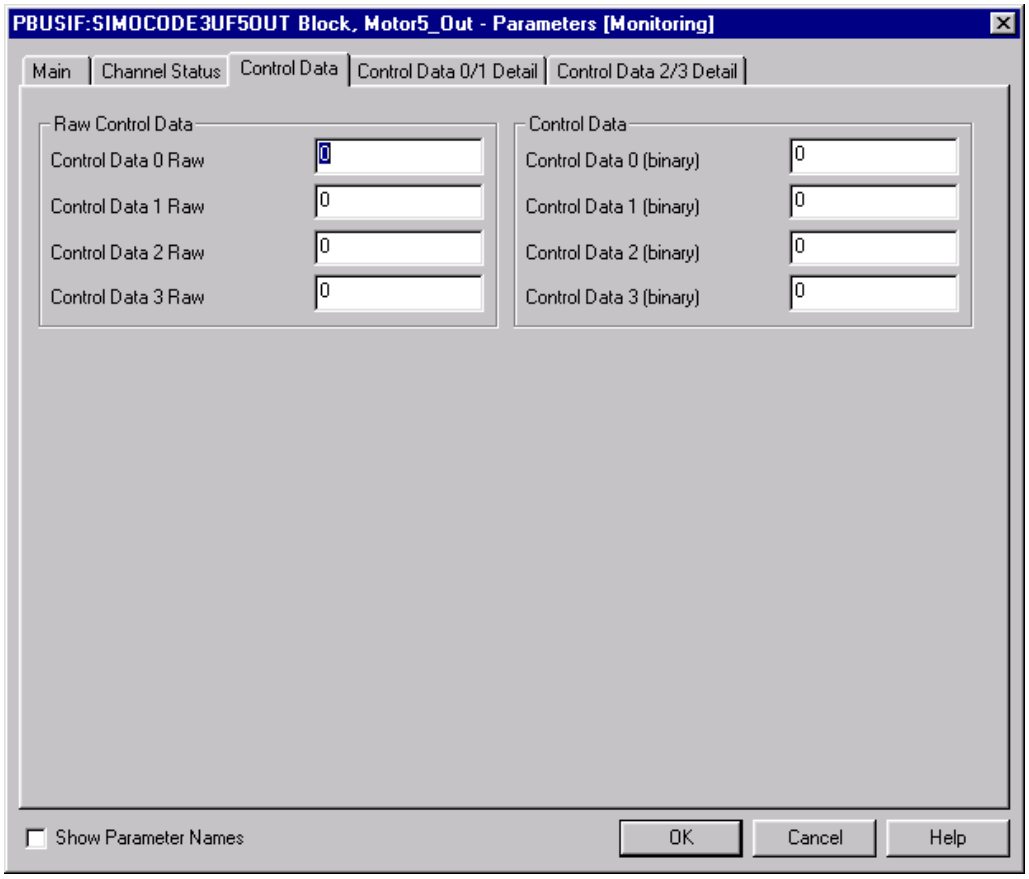
Parameter Name:	**CD3INITVALUE[0..7]**
Specific to Block:	SIMOCODE3UF5OUT
Description:	Init state
Data Type:	BOOLEAN
Range:	OFF(0)
	ON(1)
Default:	OFF(0)
Config Load:	Yes
Access Lock:	Application Developer
Residence:	CEE
Related Parameters:	CD3BITS[0..7], CD3INITMODE[0..7]

Remarks:	If the corresponding CD3INITMODE[0..7] is enabled: In case of losing the communication between controller and device the corresponding output bit CD3BITS[0..7] will be set to the related CD3INITVALUE[0..7]. Available in all 3 basic types.

12.3.5 Simocode Output Channel Block Operation

12.3.6 Control Data

The Control Data tab of the Simocode Output Channel Block is pictured below.



Notes on the example tab:

- This tab shows the simocode output parameters. The left hand side group box contains the raw data, whereas the binary indication of control data is shown in the right site. This kind of presentation makes it easier to detect the separate control data bits. Unfortunately, it is not possible to show leading '0's!. Thus, it is recommended that the user reads the binary data from right to left. The first digit from right represents bit 0, the second digit from right represents bit 1, and so on up to bit 7. Note: In the Control Data x/y Detail tabs, the separate control data bits with their user specific descriptions are displayed.

12.3.7 Operational Parameters for the Control Data tab

The following parameters describe the operational parameters of the Simocode Output Channel Block (Control Data tab).

Parameter Name:	**CD0RAW**
Specific to Block:	SIMOCODE3UF5OUT
Description:	Control Data Byte 0 This value is the output byte that is transported via PROFIBUS. It is converted by the block from CD0BITS[x] and to CD0BIN.
Data Type:	INT32
Range:	0..255
Default:	0
Config Load:	No
Access Lock:	Operator
Residence:	CEE
Related Parameters:	CD0BIN, CD0BITS[x]
Remarks:	This parameter is available in all 3 basic types. See the 'SIMOCODE-DP System Manual' for details.

Parameter Name:	**CD0BIN**
Specific to Block:	SIMOCODE3UF5OUT
Description:	Control Data Byte 0 in binary indication.
Data Type:	INT32
Range:	
Default:	0
Config Load:	No
Access Lock:	View only
Residence:	CEE
Related Parameters:	CD0RAW, CD0BITS[x]
Remarks:	This parameter represents the control data byte 0 in specifically binary format (without leading '0's). The first digit from right represents bit 0.0, the second digit from right represents bit 0.1, and so on up to bit 0.7. Digit '0' represents state 'OFF', digit '1' state 'ON'. This parameter is available in all 3 basic types.

Parameter Name:	**CD1RAW**
Specific to Block:	SIMOCODE3UF5OUT
Description:	Control Data Byte 1 This value is the output byte that is transported via PROFIBUS. It is converted by the block from CD1BITS[x] and to CD1BIN.
Data Type:	INT32
Range:	0..255

Default:	0
Config Load:	No
Access Lock:	Operator
Residence:	CEE
Related Parameters:	CD1BIN, CD1BITS[x]
Remarks:	This parameter is available in all 3 basic types. See the 'SIMOCODE-DP System Manual' for details.

Parameter Name:	**CD1BIN**
Specific to Block:	SIMOCODE3UF5OUT
Description:	Control Data Byte 1 in binary indication.
Data Type:	INT32
Range:	
Default:	0
Config Load:	No
Access Lock:	View only
Residence:	CEE
Related Parameters:	CD1RAW, CD1BITS[x]
Remarks:	This parameter represents the control data byte 1 in specifically binary format (without leading '0's). The first digit from right represents bit 1.0, the second digit from right represents bit 1.1, and so on up to bit 1.7. Digit '0' represents state 'OFF', digit '1' state 'ON'. This parameter is available in all 3 basic types.

Parameter Name:	**CD2RAW**
Specific to Block:	SIMOCODE3UF5OUT
Description:	Control Data Byte 2 This value is the output byte that is transported via PROFIBUS. It is converted by the block from CD2BITS[x] and to CD2BIN.
Data Type:	INT32
Range:	0..255
Default:	0
Config Load:	No
Access Lock:	Operator
Residence:	CEE
Related Parameters:	CD2BIN, CD2BITS[x]
Remarks:	This parameter is available in all 3 basic types. See the 'SIMOCODE-DP System Manual' for details.

Parameter Name:	**CD2BIN**
Specific to Block:	SIMOCODE3UF5OUT
Description:	Control Data Byte 2 in binary indication.
Data Type:	INT32

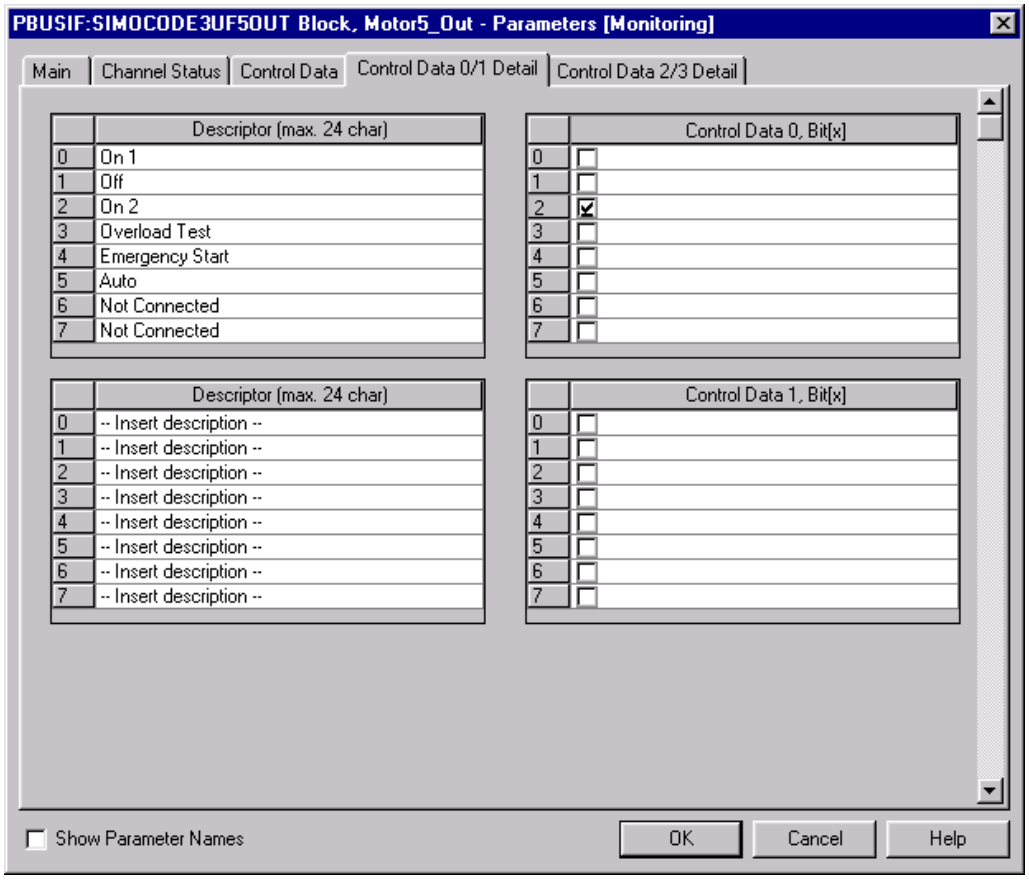
Range:	
Default:	0
Config Load:	No
Access Lock:	View only
Residence:	CEE
Related Parameters:	CD2RAW, CD2BITS[x]
Remarks:	This parameter represents the control data byte 1 in specifically binary format (without leading '0's). The first digit from right represents bit 2.0, the second digit from right represents bit 2.1, and so on up to bit 2.7. Digit '0' represents state 'OFF', digit '1' state 'ON'. This parameter is available in all 3 basic types.

Parameter Name:	**CD3RAW**
Specific to Block:	SIMOCODE3UF5OUT
Description:	Control Data Byte 3 This value is the output byte that is transported via PROFIBUS. It is converted by the block from CD3BITS[x] and to CD3BIN.
Data Type:	INT32
Range:	0..255
Default:	0
Config Load:	No
Access Lock:	Operator
Residence:	CEE
Related Parameters:	CD3BIN, CD3BITS[x]
Remarks:	This parameter is available in all 3 basic types. See the 'SIMOCODE-DP System Manual' for details.

Parameter Name:	**CD3BIN**
Specific to Block:	SIMOCODE3UF5OUT
Description:	Control Data Byte 3 in binary indication.
Data Type:	INT32
Range:	
Default:	0
Config Load:	No
Access Lock:	View only
Residence:	CEE
Related Parameters:	CD3RAW, CD3BITS[x]
Remarks:	This parameter represents the control data byte 3 in specifically binary format (without leading '0's). The first digit from right represents bit 3.0, the second digit from right represents bit 3.1, and so on up to bit 3.7. Digit '0' represents state 'OFF', digit '1' state 'ON'. This parameter is available in all 3 basic types.

12.3.8 Control Data Byte 0 and 1 Detail

The Control Data 0/1 Detail tab of the Simocode Output Channel Block is pictured below.



Notes on the example tab:

- Operational parameters on this tab are only CD0BITS(x) and CD1BITS(X).

12.3.9 Operational Parameters for the Control Data 0/1 Detail tab

The following parameters describe the operational parameters of the Simocode Output Channel Block (Control Data 0/1 Detail tab).

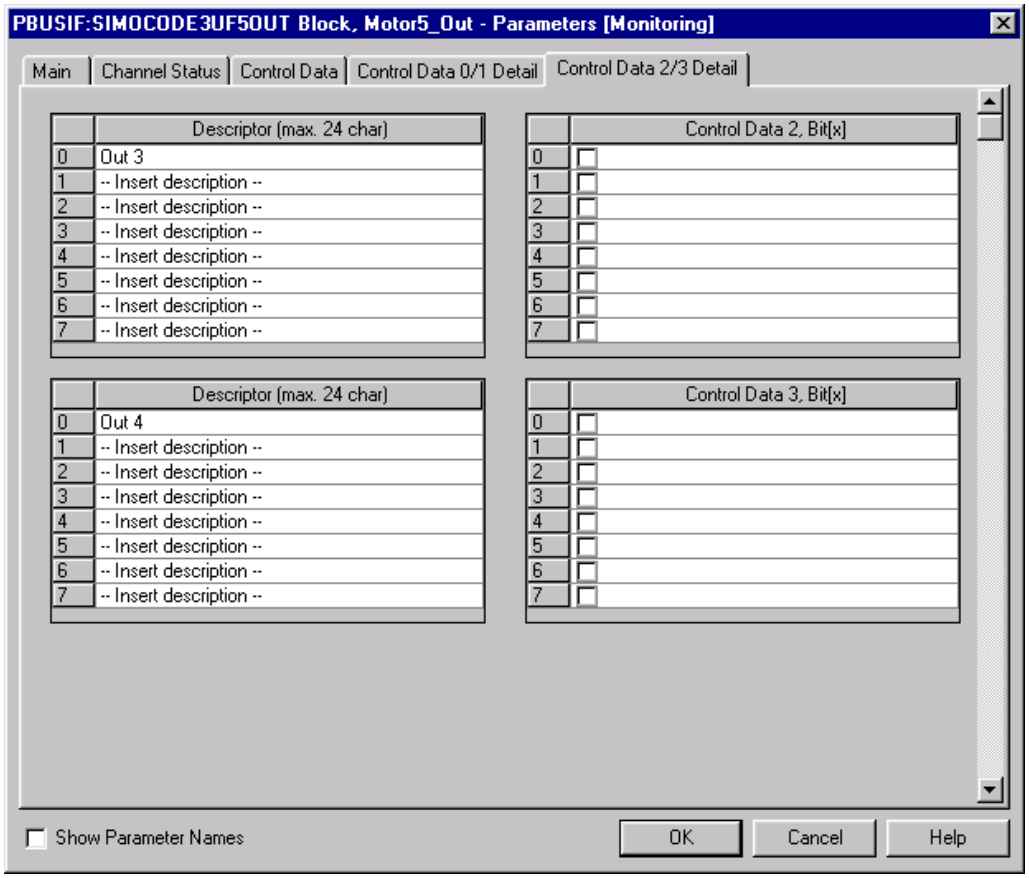
Parameter Name:	**CD2BITS[0..7]**
Specific to Block:	SIMOCODE3UF5OUT
Description:	Control Data bit 2.0 to bit 2.7. This parameter represents the Boolean (flag) indication of control data bit 2.0 to bit 2.7. It is converted by the block from CD0RAW.
Data Type:	BOOLEAN
Range:	OFF(0) ON(1)

Default:	OFF
Config Load:	No
Access Lock:	Operator
Residence:	CEE
Related Parameters:	CD2RAW, CD2BIN
Remarks:	This parameter is available in all 3 basic types.

Parameter Name:	**CD3BITS[0..7]**
Specific to Block:	SIMOCODE3UF5OUT
Description:	Control Data bit 3.0 to bit 3.7. This parameter represents the Boolean (flag) indication of control data bit 3.0 to bit 3.7. It is converted by the block from CD3RAW.
Data Type:	BOOLEAN
Range:	OFF(0) ON(1)
Default:	OFF
Config Load:	No
Access Lock:	Operator
Residence:	CEE
Related Parameters:	CD3RAW, CD3BIN
Remarks:	This parameter is available in all 3 basic types.

12.3.10 Control Data Byte 2 and 3 Detail

The Control Data 2/3 Detail tab of the Simocode Output Channel Block is pictured below.



Notes on the example tab:

- Operational parameters on this tab are only CD2BITS(x) and CD3BITS(x).

12.3.11 Operational Parameters for the Control Data 0/1 Detail tab

The following parameters describe the operational parameters of the Simocode Output Channel Block (Control Data 0/1 Detail tab).

Parameter Name:	**CD2BITS[0..7]**
Specific to Block:	SIMOCODE3UF5OUT
Description:	Control Data bit 2.0 to bit 2.7. This parameter represents the Boolean (flag) indication of control data bit 2.0 to bit 2.7. It is converted by the block from CD0RAW.
Data Type:	BOOLEAN
Range:	OFF(0) ON(1)
Default:	OFF
Config Load:	No
Access Lock:	Operator
Residence:	CEE

Related Parameters:	CD2RAW, CD2BIN
Remarks:	This parameter is available in all 3 basic types.

Parameter Name:	**CD3BITS[0..7]**
Specific to Block:	SIMOCODE3UF5OUT
Description:	Control Data bit 3.0 to bit 3.7. This parameter represents the Boolean (flag) indication of control data bit 3.0 to bit 3.7. It is converted by the block from CD3RAW.
Data Type:	BOOLEAN
Range:	OFF(0) ON(1)
Default:	OFF
Config Load:	No
Access Lock:	Operator
Residence:	CEE
Related Parameters:	CD3RAW, CD3BIN
Remarks:	This parameter is available in all 3 basic types.

13 Notices

Trademarks

Experion®, PlantScape®, SafeBrowse®, TotalPlant®, and TDC 3000® are registered trademarks of Honeywell International, Inc.

OneWireless™ is a trademark of Honeywell International, Inc.

Other trademarks

Microsoft and SQL Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Trademarks that appear in this document are used only to the benefit of the trademark owner, with no intention of trademark infringement.

Third-party licenses

This product may contain or be derived from materials, including software, of third parties. The third party materials may be subject to licenses, notices, restrictions and obligations imposed by the licensor. The licenses, notices, restrictions and obligations, if any, may be found in the materials accompanying the product, in the documents or files accompanying such third party materials, in a file named third_party_licenses on the media containing the product, or at http://www.honeywell.com/ps/thirdpartylicenses.

13.1 Documentation feedback

You can find the most up-to-date documents on the Honeywell Process Solutions support website at:

http://www.honeywellprocess.com/support

If you have comments about Honeywell Process Solutions documentation, send your feedback to:

hpsdocs@honeywell.com

Use this email address to provide feedback, or to report errors and omissions in the documentation. For immediate help with a technical problem, contact your local Honeywell Process Solutions Customer Contact Center (CCC) or Honeywell Technical Assistance Center (TAC) listed in the “Support and other contacts” section of this document.

13.2 How to report a security vulnerability

For the purpose of submission, a security vulnerability is defined as a software defect or weakness that can be exploited to reduce the operational or security capabilities of the software.

Honeywell investigates all reports of security vulnerabilities affecting Honeywell products and services.

To report a potential security vulnerability against any Honeywell product, please follow the instructions at:

https://honeywell.com/pages/vulnerabilityreporting.aspx

Submit the requested information to Honeywell using one of the following methods:

- Send an email to security@honeywell.com.

  or

- Contact your local Honeywell Process Solutions Customer Contact Center (CCC) or Honeywell Technical Assistance Center (TAC) listed in the “Support and other contacts” section of this document.

13.3 Support

For support, contact your local Honeywell Process Solutions Customer Contact Center (CCC). To find your local CCC visit the website, https://www.honeywellprocess.com/en-US/contact-us/customer-support-contacts/Pages/default.aspx.

13.4 Training classes

Honeywell holds technical training classes on Experion PKS. These classes are taught by experts in the field of process control systems. For more information about these classes, contact your Honeywell representative, or see http://www.automationcollege.com.