TeSys Active

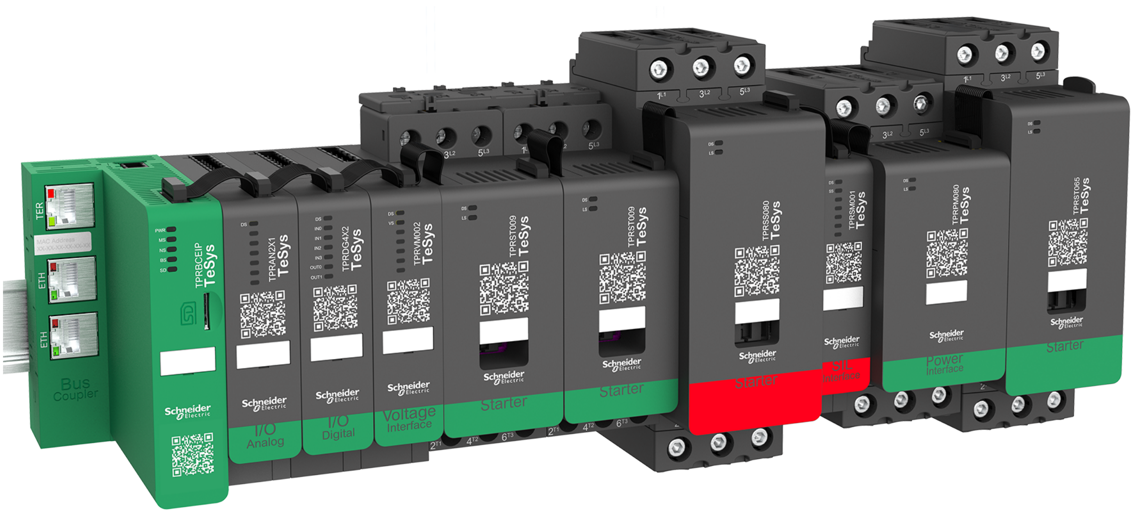
TeSys island – Digital Motor Management Solution

For PROFINET and PROFIBUS Applications

Quick Start and Function Block Library Guide

TeSys offers innovative and connected solutions for motor starters.

DOCA0272EN-00
08/2023



www.se.com

Legal Information

As standards, specifications, and designs change from time to time, information contained in this guide may be subject to change without notice.

To the extent permitted by applicable law, no responsibility or liability is assumed by Schneider Electric and its subsidiaries for any errors or omissions in the informational content of this material or consequences arising out of or resulting from the use of the information contained herein.

As part of a group of responsible, inclusive companies, we are updating our communications that contain non-inclusive terminology. Until we complete this process, however, our content may still contain standardized industry terms that may be deemed inappropriate by our customers.

Table of Contents

Safety Information

Important Information

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

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

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

⚠ DANGER

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

⚠ WARNING

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

⚠ CAUTION

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

NOTICE

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

Please Note

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

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

About the Book

Document Scope

This user guides provides:

- instructions for configuring a TeSys™ island device within the Siemens® Totally Integrated Automation (TIA) Portal environment.
- characteristics of the Siemens SIMATIC S7–1200 CPUs and PROFINET IO as bus systems. The basic principles remain the same when using PROFIBUS DP or S7–1500 PLCs as they are for the bus systems above. The process for handling other systems can vary. This guide outlines these differences.
- information of TeSys island IEC 61131-3 function block libraries for Siemens TIA Portal. These libraries mainly consist of function blocks that can be used to control, monitor, and diagnose the avatars and devices of a TeSys island from within a PLC program written in Siemens TIA Portal.
- details on the interfaces of said function blocks as well as an introduction to their usage to aid in creating such a PLC program. It also defines the requirements and prerequisites that need to be met to use the function blocks.

Validity Note

This guide is valid for all TeSys island configurations. The availability of some functions described in this guide depends on the communication protocol used and the physical modules installed on the TeSys island.

For product compliance with environmental directives such as RoHS, REACH, PEP, and EOLI, go to www.se.com/green-premium.

For technical characteristics of the physical modules described in this guide, go to www.se.com.

The technical characteristics presented in this guide should be the same as those that appear online. We may revise content over time to improve clarity and accuracy. If you see a difference between the information contained in this guide and online information, use the online information.

Related Documentation

Document title	Description	Document number
TeSys island – System, Installation, and Operation Guide	Describes main functions, mechanical installation, wiring, commissioning of the TeSys island, and how to operate and maintain TeSys island.	DOCA0270EN
TeSys island – EtherNet/IP™ – Quick Start and Function Block Library Guide	Describes how to integrate the TeSys island and the information of TeSys island library used in the Rockwell Software® Studio 5000® EtherNet/IP environment.	DOCA0271EN
TeSys island – Functional Safety Guide	Describes the Functional Safety features of TeSys island.	8536IB1904
TeSys island – Third Party Function Block Guide	Contains the information needed to create function blocks for third party hardware.	8536IB1905
TeSys island – DTM Online Help Guide	Describes how to install and use various functions of TeSys island configuration software and how to configure the parameters of TeSys island.	8536IB1907
TeSys island – Product Environmental Profile	Describes constituent materials, recyclability potential, and environmental impact information for the TeSys island.	ENVPEP1904009
TeSys island – Product End of Life Instructions	Contains end of life instructions for the TeSys island.	ENVEOLI1904009
TeSys island – Instruction Sheet, Bus Coupler, TPRBCEIP	Describes how to install the TeSys island Ethernet/IP bus coupler.	MFR44097
TeSys island – Instruction Sheet, Bus Coupler, TPRBCPFN	Describes how to install the TeSys island PROFINET bus coupler.	MFR44098
TeSys island – Instruction Sheet, Bus Coupler, TPRBCPFB	Describes how to install the TeSys island PROFIBUS DP bus coupler.	GDE55148
TeSys island – Instruction Sheet, Starters and Power Interface Modules, Size 1 and 2	Describes how to install size 1 and 2 TeSys island starters and power interface modules.	MFR77070
TeSys island – Instruction Sheet, Starters and Power Interface Modules, Size 3	Describes how to install size 3 TeSys island starters and power interface modules.	MFR77085
TeSys island – Instruction Sheet: Input/Output Modules	Describes how to install the TeSys island analog and digital I/O modules.	MFR44099
TeSys island – Instruction Sheet: SIL Interface and Voltage Interface Modules	Describes how to install the TeSys island voltage interface modules and SIL[1] interface modules.	MFR44100

[1] Safety Integrity Level according to standard IEC 61508.

Precautions

Read and understand the following precautions before performing any procedures in this guide.

⚠⚠ DANGER

HAZARD OF ELECTRIC SHOCK, EXPLOSION, OR ARC FLASH

- This equipment must only be installed and serviced by qualified electrical personnel.
- Turn off all power supplying this equipment before working on or inside this equipment.
- Use only the specified voltage when operating this equipment and any associated products.
- Always use a properly rated voltage sensing device to confirm power is off.
- Use appropriate interlocks where personnel and/or equipment hazards exist.
- Power line circuits must be wired and protected in compliance with local and national regulatory requirements.
- Apply appropriate personal protective equipment (PPE) and follow safe electrical work practices per NFPA 70E, NOM-029-STPS, or CSA Z462 or local equivalent.

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

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- For complete instructions about functional safety, refer to the TeSys™ island Functional Safety Guide, 8536IB1904.
- Do not disassemble, repair, or modify this equipment. There are no user serviceable parts.
- Install and operate this equipment in an enclosure appropriately rated for its intended application environment.
- Each implementation of this equipment must be individually and thoroughly tested for proper operation before being placed into service.

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

⚠ **WARNING:** This product can expose you to chemicals including Antimony oxide (Antimony trioxide), which is known to the State of California to cause cancer. For more information go to www.P65Warnings.ca.gov.

Qualified Personnel

Only appropriately trained personnel who are familiar with and understand the content of this guide and all other related product documentation are authorized to work on and with this product.

The qualified personnel must be able to detect possible hazards that may arise from modifying parameter values and generally from mechanical, electrical, or electronic equipment. The qualified personnel must be familiar with the standards, provisions, and regulations for the prevention of industrial accidents, which they must observe when designing and implementing the system.

The use and application of the information contained in this guide requires expertise in the design and programming of automated control systems. Only you, the user, machine builder, or integrator, can be aware of all the conditions and factors present during installation, setup, operation, and maintenance of the machine or process, and can therefore determine the automation and associated equipment and the related safeties and interlocks which can be effectively and properly used when selecting automation and control equipment, and any other related equipment or software, for a particular application. You must also consider applicable local, regional, or national standards and/or regulations.

Pay particular attention to conformance with any safety information, electrical requirements, and normative standards that apply to your machine or process in the use of this equipment.

Intended Use

The products described in this guide, together with software, accessories, and options, are starters for low-voltage electrical loads, intended for industrial use according to the instructions, directions, examples, and safety information contained in the present document and other supporting documentation.

The product may only be used in compliance with all applicable safety regulations and directives, the specified requirements, and the technical data.

Before using the product, you must perform a risk assessment of the planned application. Based on the results, appropriate safety-related measures must be implemented.

Since the product is used as a component of a machine or process, you must ensure the safety of persons by means of the overall system design.

Operate the product only with the specified cables and accessories. Use only genuine accessories and spare parts.

Any use other than the use explicitly permitted is prohibited and can result in unanticipated hazards.

Cybersecurity

Introduction

Cybersecurity is a branch of network administration that addresses attacks on or by PCs and through PC networks that can result in accidental or intentional disruptions. The objective of cybersecurity is to help provide increased levels of protection for information and physical assets from theft, corruption, misuse, or accidents while maintaining access for their intended users.

No single cybersecurity approach is adequate. Schneider Electric recommends a defense-in-depth approach. Conceived by the National Security Agency (NSA), this approach layers the network with security features, appliances, and processes. The basic components of this approach are:

- Risk assessment
- Security plan built on the results of the risk assessment
- Multi-phase training campaign
- Physical separation of the industrial networks from enterprise networks using a demilitarized zone (DMZ) and the use of firewalls and routing to establish other security zones
- System access control
- Device hardening
- Network monitoring and maintenance

This section defines elements that help you configure a system that is less susceptible to cyber attacks. For detailed information on the defense-in-depth approach, refer to the *Recommended Cybersecurity Best Practices* on the Schneider Electric website.

Schneider Electric's Approach on Cybersecurity

Schneider Electric adheres to industries best practice in the development and implementation of control systems. This includes a defense-in-depth approach to secure an industrial control system. This approach places the controllers behind one or more firewalls to restrict access to authorized personnel and protocols only.

⚠WARNING

UNAUTHENTICATED ACCESS AND SUBSEQUENT UNAUTHORIZED OPERATION

- Evaluate whether your equipment or complete environment are connected to your critical infrastructure and, if so, take appropriate steps in terms of prevention, based on defense-in-depth, before connecting the automation system to any network.
- Limit the number of devices connected to a network inside your company.
- Isolate your industrial network from other networks inside your company.
- Protect any network against unintended access by using firewalls, VPN, or other, proven security measures.
- Monitor activities within your systems.
- Prevent subject devices from direct access or direct link by unauthorized parties or unauthenticated actions.
- Prepare a recovery plan including backup of your system and process information.

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

Cyber Threats

Cyber threats are deliberate actions or accidents that can disrupt the normal operations of PCs and PC networks. These actions can be initiated from within the physical facility or from an external location. Security challenges for the control environment include:

- Diverse physical and logical boundaries
- Multiple sites and large geographic spans
- Adverse effects of security implementation on process availability
- Increased exposure to worms and viruses migrating from business systems to control systems as business-control communications become more open

- Increased exposure to malicious software from USB devices, vendor and service technician laptops, and the enterprise network
- Direct impact of control systems on physical and mechanical systems

Sources of Cyber Attacks

Implement a cybersecurity plan that accounts for various potential sources of cyber attacks and accidents, including:

Source	Description
Internal	• Inappropriate employee or contractor behavior • Disgruntled employee or contractor
External opportunistic (non-directed)	• Script kiddies[1] • Recreational hackers • Virus writers
External deliberate (directed)	• Criminal groups • Activists • Terrorists • Agencies of foreign states
Accidental	

[1] Slang term for hackers who use malicious scripts written by others without necessarily possessing a comprehensive understanding of how the script works or its potential impact on a system.

A deliberate cyber attack on a control system may be launched to achieve a number of malicious results, including:

- Disrupt the production process by blocking or delaying the flow of information.
- Damage, disable, or shut down equipment to negatively impact production or the environment.
- Modify or disable safety systems to cause intentional harm.

How Attackers Gain Access

A cyber attacker bypasses the perimeter defenses to gain access to the control system network. Common points of access include:

- Dial-up access to remote terminal unit (RTU) devices
- Supplier access points (such as technical support access points)
- IT-controlled network products
- Corporate virtual private network (VPN)
- Database links
- Poorly configured firewalls
- Peer utilities

Reporting and Management

To submit a cybersecurity question, report security issues, or to get the latest news from Schneider Electric, visit our Schneider Electric website.

Introduction to TeSys island

What’s in This Part

TeSys island Concept

What’s in This Chapter

TeSys island is a modular, multifunctional system providing integrated functions inside an automation architecture, primarily for the direct control and management of low-voltage loads. TeSys island can switch, help protect, and manage motors and other electrical loads up to 80 A (AC1) installed in an electrical control panel.

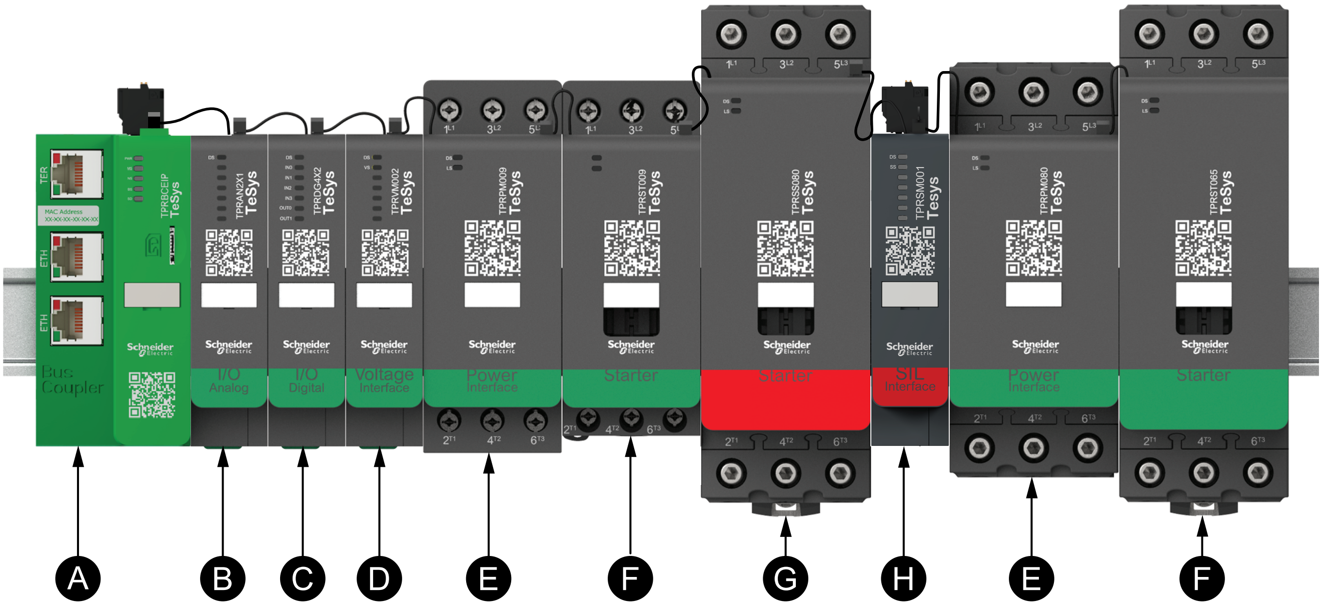
This system is designed around the concept of TeSys avatars. These avatars:

- Represent both the logical and physical aspects of the automation functions
- Determine the configuration of the TeSys island

The logical aspects of the TeSys island are managed with software tools, covering all phases of product and application lifecycle: design, engineering, commissioning, operation, and maintenance.

The physical TeSys island consists of a set of devices installed on a single DIN rail and connected together with flat cables providing the internal communication between modules. The external communication with the automation environment is made through a single bus coupler module, and the TeSys island is seen as a single node on the network. The other modules include starters, power interface modules, analog and digital I/O modules, voltage interface modules, and SIL (Safety Integrity Level according to standard IEC 61508) interface modules, covering a wide range of operational functions.

TeSys island Overview



A	Bus Coupler	E	Power Interface Module
B	Analog I/O Module	F	Standard Starter
C	Digital I/O Module	G	SIL Starter
D	Voltage Interface Module	H	SIL Interface Module

Master Range: TeSys

TeSys™ is an innovative motor control and management solution from the global market leader. TeSys offers connected, efficient products and solutions for switching and protection of motors and electrical loads in compliance with all major global electrical standards.

General Information

TeSys island add-on instructions provide function blocks to support developing applications and to control avatar modules. Avatar modules are digital function objects managed by the TeSys island system. The system interacts with power devices and accessories such as the analog I/O devices. The avatar modules are configured on the TeSys island, and the bus coupler (via the System avatar) manages fieldbus communication with the controller.

The configuration of the TeSys island modules is managed by the TeSys island Device Type Manager (DTM). For further information, refer to the TeSys island DTM Library Online Help.

Avatar Definition

TeSys avatars bring ready-to-use functions through their predefined logic and associated physical devices. The avatar logic is executed in the bus coupler. The bus coupler manages data exchanges internally within the TeSys island, and also externally with the PLC.

There are four types of TeSys avatars:

System avatar

Represents the whole island as a system. The System avatar allows setting the network configuration and computes TeSys island level data.

Device avatars

Represent functions performed by switches and I/O modules.

Load avatars

Represent functions related to specific loads, such as a forward-reverse motor. Load avatars include the appropriate modules and operating characteristics to serve the load type. For example, a Motor Two Directions avatar includes two starter modules, accessories, pre-programmed control logic, and a pre-configuration of the available protection functions.

Standard (non-SIL[2]) Load avatars provide the following:

- Local control

  NOTE: Local control is applicable for all Load avatars (except PIM avatar).

- Local trip reset (to allow an operator to use a local input to trigger the local trip reset on rising edge of the input. When the input changes from 0 to 1, then the trip reset of avatar is executed)

  NOTE: Local trip reset is applicable for all Load avatars (except PIM avatar).

- Bypass (to allow an operator to use a local command to temporarily bypass a trip condition and continue the operation of the avatar)
- Process variable monitoring

Application avatars

Represent functions related to specific user applications, such as a pump or conveyor. Application avatars provide the following:

- Local control
- Local trip reset (to allow an operator to use a local input to trigger the local trip reset on rising edge of the input. When the input changes from 0 to 1, then the trip reset of avatar is executed)
- Bypass (to allow an operator to use a local command to temporarily bypass a trip condition and continue the operation of the avatar)
- Manual mode override (to allow an operator to use a local input to override the configured control mode and control the avatar from a local command source)
- Process variable monitoring

For example, a Pump avatar includes the following:

- One starter module
- One or more digital I/O modules for local control, local trip, and process variable (PV) switches
- Configurable control logic
- Pre-configuration of the load and electrical functions

[2] Safety Integrity Level according to standard IEC 61508.

PV inputs receive analog values from sensors such as a pressure meter, a flow meter, or a vibration meter. PV switches receive discrete signals from switches such as a flow switch or a pressure switch.

Operational control (Run and Stop command) of the avatar in autonomous mode is configurable for up to two PV inputs or PV switches. It includes settings for the threshold and hysteresis for analog inputs, and positive or negative logic for both analog and digital inputs for the Pump avatar.

The avatars installed on the TeSys island are controlled by the TeSys island bus coupler. Each avatar includes predefined logic for managing its physical modules, while also providing easy data exchange with PLCs through function blocks. Avatarsinclude pre-configuration of the available protection functions.

Information accessible through the avatar includes the following:

- Control data
- Advanced diagnostics data
- Asset management data
- Energy data

List of TeSys Avatars

TeSys Avatars

Name	Icon	Description
System avatar		A required avatar that enables a single point of communication to the TeSys island.
	Device	
Switch		To make or break a power line in an electrical circuit
Switch - SIL Stop, W. Cat 1/2[3]		To make or break a power line in an electrical circuit with Stop Category 0 or Stop Category 1[4] function compliance for Wiring Category 1 and Category 2.
Switch - SIL Stop, W. Cat 3/4[5]		To make or break a power line in an electrical circuit with Stop Category 0 or Stop Category 1 function compliance for Wiring Category 3 and Category 4.
Digital I/O		To provide control of 2 digital outputs and status of 4 digital inputs

[3] Safety Integrity Level according to standard IEC 61508. Wiring Category 1 and Category 2 according to ISO 13849.
[4] Stop category according to EN/IEC 60204-1.
[5] Safety Integrity Level according to standard IEC 61508. Wiring Category 3 and Category 4 according to ISO 13849.

TeSys Avatars (Continued)

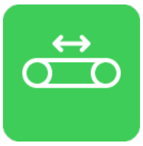
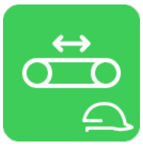
Name	Icon	Description
Analog I/O		To provide control of 1 analog output and status of 2 analog inputs
Load		
Power Interface without I/O (measure)		To monitor current supplied to an external device, such as a solid-state relay, soft starter, or variable speed drive
Power Interface with I/O (control)		To monitor current supplied to and to control an external device, such as a solid-state relay, soft starter, or variable speed drive
Motor One Direction		To manage[6] a motor in one direction
Motor One Direction - SIL Stop, W. Cat 1/2		To manage a motor in one direction, with Stop Category 0 or Stop Category 1 function compliance for Wiring Category 1 and Category 2.
Motor One Direction - SIL Stop, W. Cat 3/4		To manage a motor in one direction, with Stop Category 0 or Stop Category 1 function compliance for Wiring Category 3 and Category 4.
Motor Two Directions		To manage a motor in two directions (forward and reverse)
Motor Two Directions - SIL Stop, W. Cat 1/2		To manage a motor in two directions (forward and reverse), with Stop Category 0 or Stop Category 1 function compliance for Wiring Category 1 and Category 2
Motor Two Directions - SIL Stop, W. Cat 3/4		To manage a motor in two directions (forward and reverse), with Stop Category 0 or Stop Category 1 function compliance for Wiring Category 3 and Category 4

[6] “Manage” in this context encompasses energizing, controlling, monitoring, diagnosing, and protecting the load.

TeSys Avatars (Continued)

Name	Icon	Description
Motor Y/D One Direction	M Y/Δ	To manage a wye-delta (star-delta) motor in one direction
Motor Y/D Two Directions	M Y/Δ	To manage a wye-delta (star-delta) motor in two directions (forward and reverse)
Motor Two Speeds	M	To manage a two-speed motor and two-speed motor with Dahlander option
Motor Two Speeds - SIL Stop, W. Cat 1/2	M	To manage a two-speed motor, with Stop Category 0 or Stop Category 1 function compliance for Wiring Category 1 and Category 2
Motor Two Speeds - SIL Stop, W. Cat 3/4	M	To manage a two-speed motor, with Stop Category 0 or Stop Category 1 function compliance for Wiring Category 3 and Category 4
Motor Two Speeds Two Directions	M	To manage a two-speed motor in two directions (forward and reverse)
Motor Two Speeds Two Directions - SIL Stop, W. Cat 1/2	M	To manage a two-speed motor in two directions (forward and reverse), with Stop Category 0 or Stop Category 1 function compliance for Wiring Category 1 and Category 2
Motor Two Speeds Two Directions - SIL Stop, W. Cat 3/4	M	To manage a two-speed motor in two directions (forward and reverse), with Stop Category 0 or Stop Category 1 function compliance for Wiring Category 3 and Category 4
Resistor		To manage a resistive load
Power Supply		To manage a power supply

TeSys Avatars (Continued)

Name	Icon	Description
Transformer		To manage a transformer
Application		
Pump		To manage a pump
Conveyor One Direction		To manage a conveyor in one direction
Conveyor One Direction - SIL Stop, W. Cat 1/2		To manage a conveyor in one direction, with Stop Category 0 or Stop Category 1 function compliance for Wiring Category 1 and Category 2
Conveyor Two Directions		To manage a conveyor in two directions (forward and reverse)
Conveyor Two Directions - SIL Stop, W. Cat 1/2		To manage a conveyor in two directions (forward and reverse), with Stop Category 0 or Stop Category 1 function compliance for Wiring Category 1 and Category 2

NOTE: For two speed two direction avatar disable current phase reversal trip.

Methodology

The integration of theTeSys™ island into Siemens TIA Portal uses the following components:

- Device description files for TeSys island bus couplers PROFINET I/O and PROFIBUS DP
- IEC 61131–3 function block library to access the data of the TeSys island via the bus systems from within TIA Portal
- The CAx data import feature of Siemens TIA Portal to import the island composition from TeSys island Device Type Manager (DTM)
- A basic TCI integration of the TeSys island DTM into Siemens TIA Portal
- Import of a global data block containing Predictive Alarm Messages, provided by the TeSys island DTM

The prerequisites for using and installing the components are described in the following chapters.

SoMove™ software provides Automation Markup Language (AutomationML; AML) files containing the TeSys™ island topology and parts of the configuration data. These files can be imported into Siemens TIA Portal using the CAx data import feature. Using the CAx data import feature helps you avoid manually recreating the island topology in Siemens TIA Portal. You must import the appropriate device description files into Siemens TIA Portal before you import the AutomationML files.

NOTE:

- There are interdependencies between the avatars in the TeSys island (that is, the devices' topology and their representation to the bus system). Therefore, it is recommended that you maintain and change the island topology within SoMove™ only.
- By using the CAx data import feature, data can be transferred to Siemens TIA Portal and used in an Automation project.
- The Update feature of SoMove™ enables you to transfer changes to existing island configurations within the Siemens TIA Portal (see Updating an Existing TeSys™ island Configuration in the TIA Portal, page 44).

  To avoid problems during integration, read the product release notes to make sure that the individual components in use (such as TeSys island function block library for Siemens TIA Portal, TeSys island firmware, TeSys island device description files for PROFIBUS DP and PROFINET IO, SoMove software, and TeSys island DTM library) are compatible with each other.

Software Installation

What’s in This Part

Installing the Siemens TIA Portal

The software package for Siemens TIA Portal Openness must be installed along with Siemens TIA Portal before the CAx data import feature of the Siemens TIA Portal is available for use. This package is available for free. Siemens TIA Portal V15.1 package is automatically installed by default.

For Siemens TIA Portal and TIA Portal Openness system requirements and detailed instructions about how to install them, refer to .

Installing the Device Description Files

You can download and install the General Station Description Markup Language (GSDML) or General Station Description (GSD) files for the TeSys™ island PROFINET IO and PROFIBUS DP bus couplers from the Schneider Electric website.

Downnload the TeSys™ island GSDML file for the PROFINET bus coupler at https://www.se.com/ww/en/product-range-download/65746-tesys-island/.

Download the TeSys™ island GSD file for the PROFIBUS DP bus coupler at https://www.se.com/ww/en/product-range-download/65746-tesys-island/.

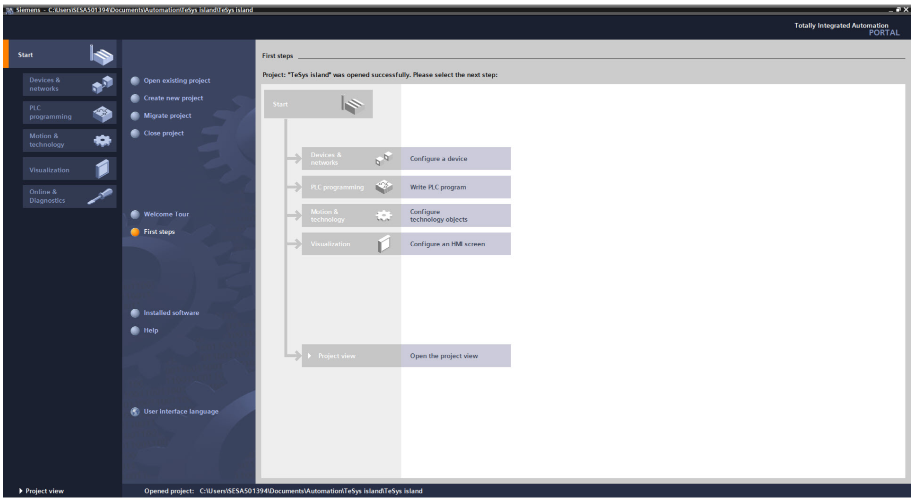
NOTE: Make sure you are using compatible versions of Device Type Manager (DTM) and GSD or GSDML.

To install the device description files in the Siemens® TIA Portal, follow these steps:

1. Launch Siemens TIA Portal and click on **Create new project**. Enter a project name, then click **Create**.

   NOTE: Click on an existing project if one has already been created.

   Start Window

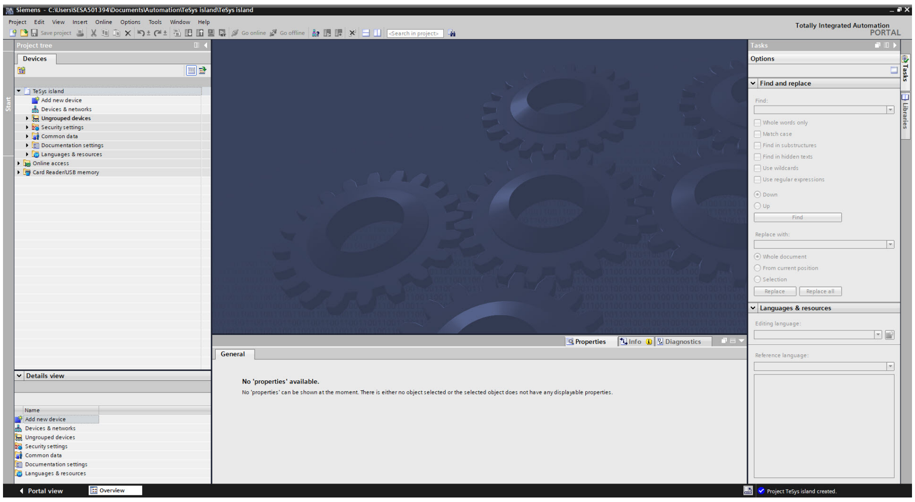


Project: "TeSys island" was opened successfully. Please select the next step:

Opened project: C:\Users\SESA501394\Documents\Automation\TeSys island\TeSys island

2. Click on **Project view** to open the project view window.

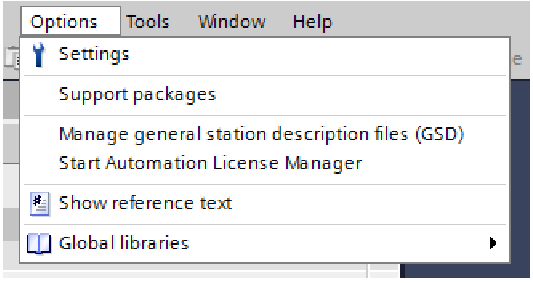
Project View



No 'properties' available.

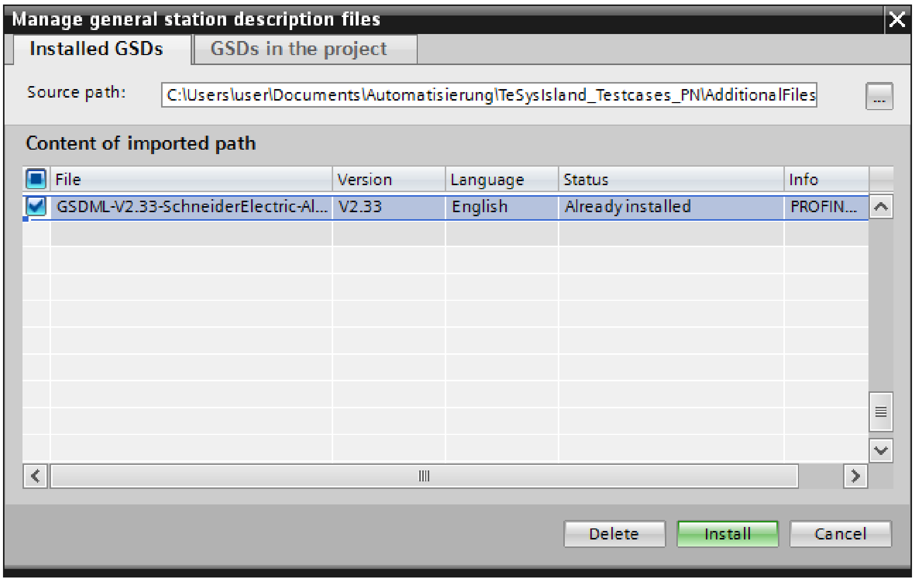
3. Select **Manage general station description files (GSD)** in the Options menu, then click **OK**.

Options



4. In the **Manage general station description files** window, select the path in the **Source path** field where the device description files will be stored.

Manage GSD Files



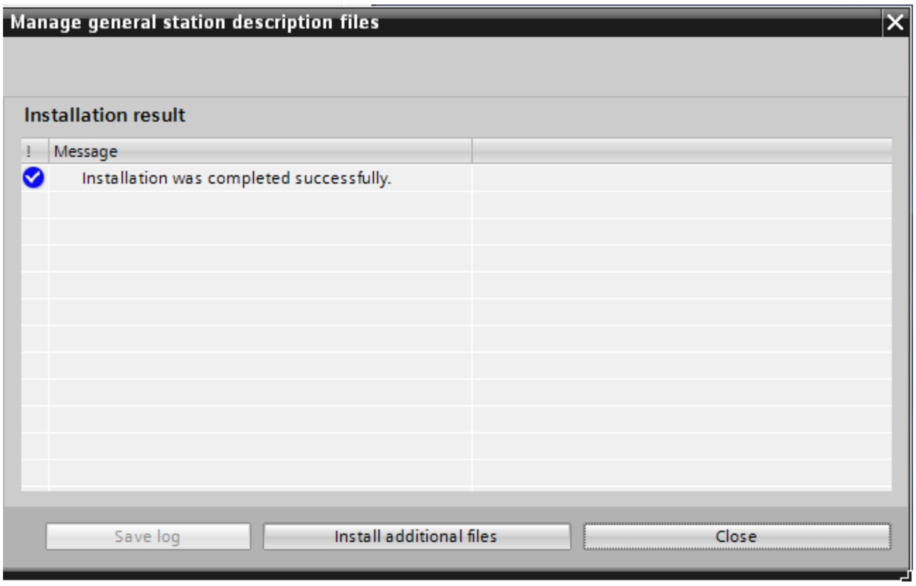
Source path: C:\Users\user\Documents\Automatisierung\TeSysIsland_Testcases_PN\AdditionalFiles

Content of imported path

■ File	Version	Language	Status	Info
☑ GSDML-V2.33-SchneiderElectric-Al...	V2.33	English	Already installed	PROFIN...

5. Select the files you want to install from the list of GSD and GSDML files, then click **Install**. The installation process begins and you will get a prompt of the results.

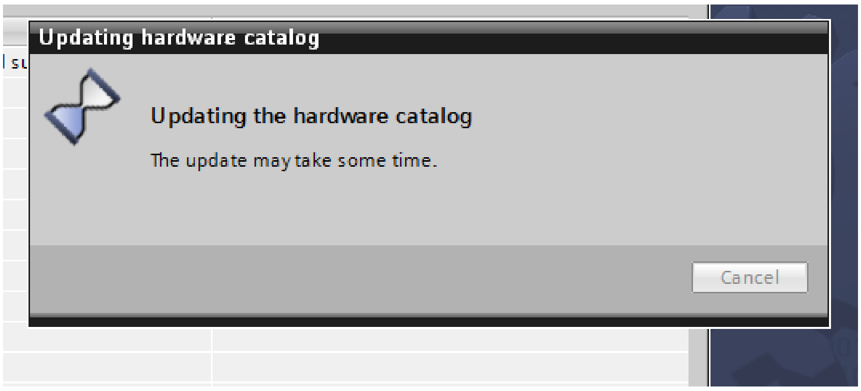
Installation Results



✓ Installation was completed successfully.

6. When the files have been imported successfully, click **Close**. Siemens TIA Portal automatically updates the hardware catalog with the imported device description files.

Updating Hardware Catalog



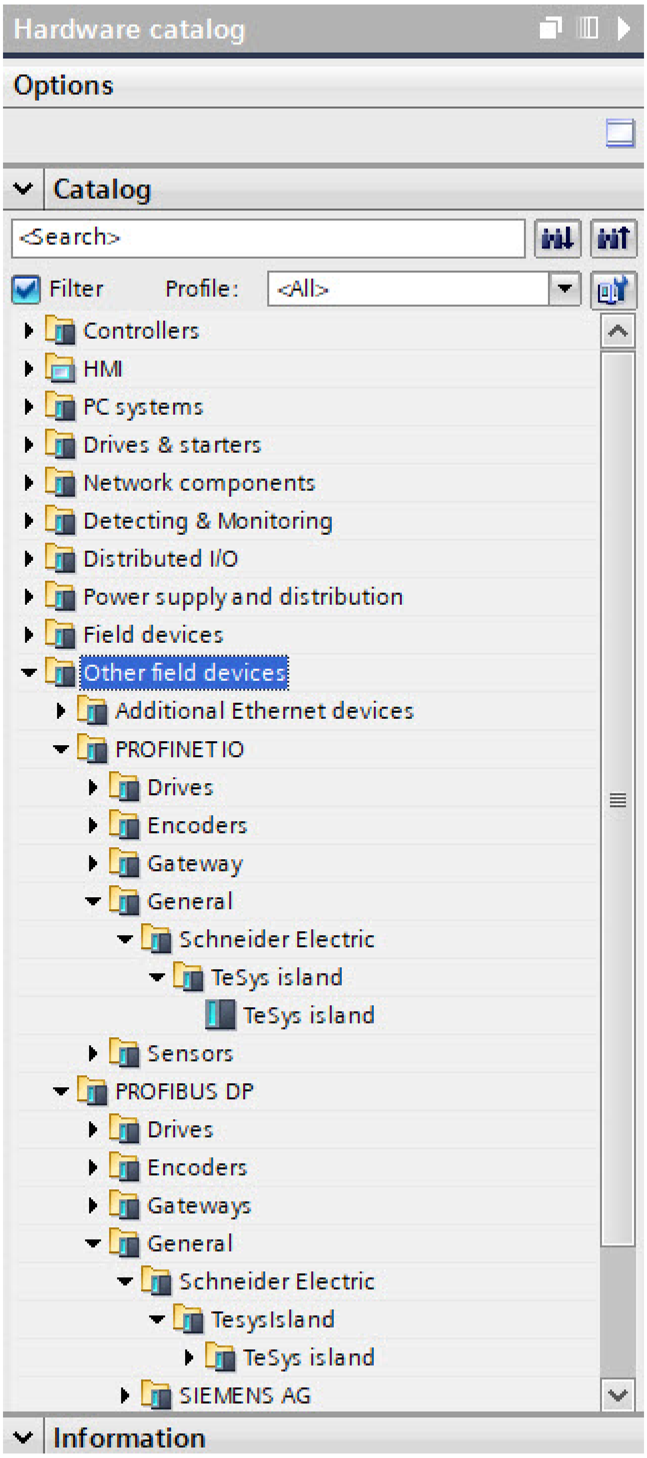
Updating the hardware catalog

The update may take some time.

7. Under Project tree, click on **Devices & networks**. The Hardware catalog task view window opens on the right of the screen.

   The installed TeSys™ island bus coupler device is now located in the hardware catalog under **Other field devices → PROFINET IO/PROFIBUS DP → General → Schneider Electric → TeSys island**.

Hardware Catalog



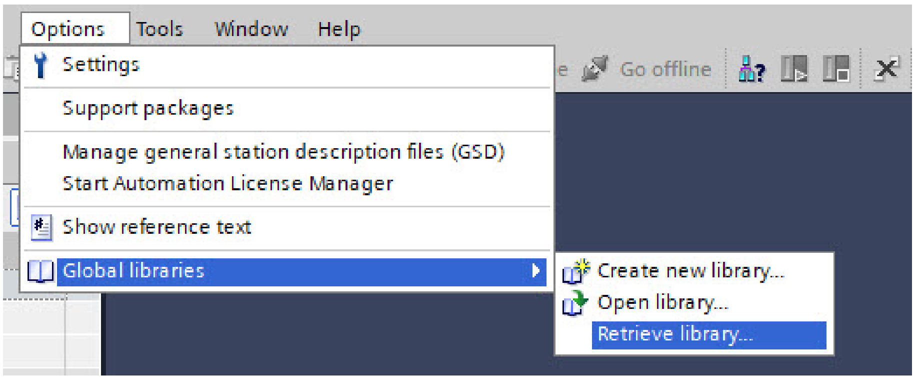
Integrating the Function Block Library

To integrate the IEC 61131-3 function block library file for the TeSys island PROFINET IO and PROFIBUS DP bus couplers, download the file from https://www.se.com/ww/en/product-range-download/65746-tesys-island/.

Use the following steps to install the file in Siemens TIA Portal after you download the file:

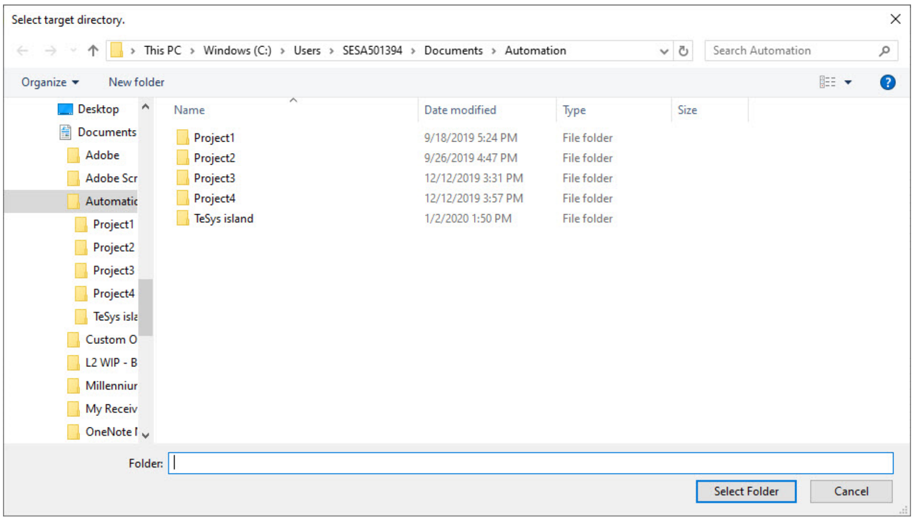
1. Launch Siemens TIA Portal, then click **Project View**.
2. Select **Global libraries** → **Retrieve library** in the Options menu, then click **Ok**.

Retrieve Library



3. Select the archive file of the function block library you downloaded.
4. Click **Open**, then select the **Automation** directory as the target directory.
5. Click **OK**.

Automation Directory

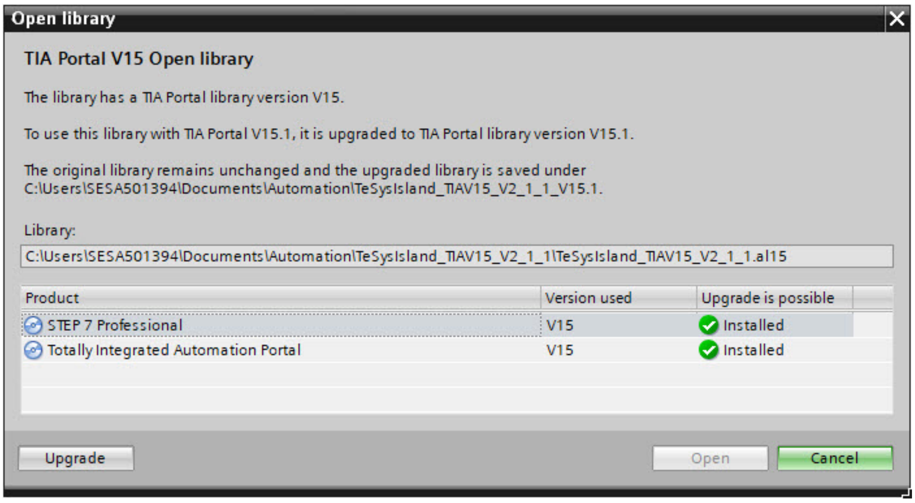

Select target directory.

This PC › Windows (C:) › Users › SESA501394 › Documents › Automation

Name	Date modified	Type	Size
Project1	9/18/2019 5:24 PM	File folder	
Project2	9/26/2019 4:47 PM	File folder	
Project3	12/12/2019 3:31 PM	File folder	
Project4	12/12/2019 3:57 PM	File folder	
TeSys island	1/2/2020 1:50 PM	File folder	

6. The following Open library window might display prompting you that the library version will be upgraded to version 15.1, depending on the version of TIA Portal that you are using.

Open Library



Open library

TIA Portal V15 Open library

The library has a TIA Portal library version V15.

To use this library with TIA Portal V15.1, it is upgraded to TIA Portal library version V15.1.

The original library remains unchanged and the upgraded library is saved under C:\Users\SESA501394\Documents\Automation\TeSysIsland_TIAV15_V2_1_V15.1.

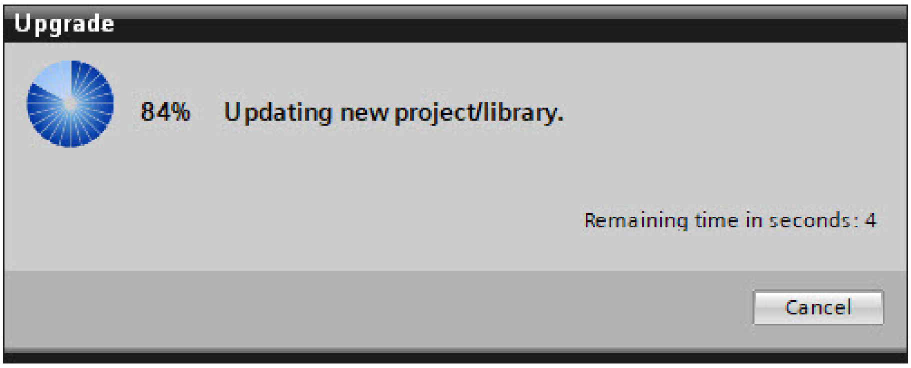
Library:

C:\Users\SESA501394\Documents\Automation\TeSysIsland_TIAV15_V2_1_1\TeSysIsland_TIAV15_V2_1_1.al15

Product	Version used	Upgrade is possible
STEP 7 Professional	V15	✓ Installed
Totally Integrated Automation Portal	V15	✓ Installed

7. Click **Upgrade** in the Open library window and the Library Upgrade progress window displays.

Library Upgrade

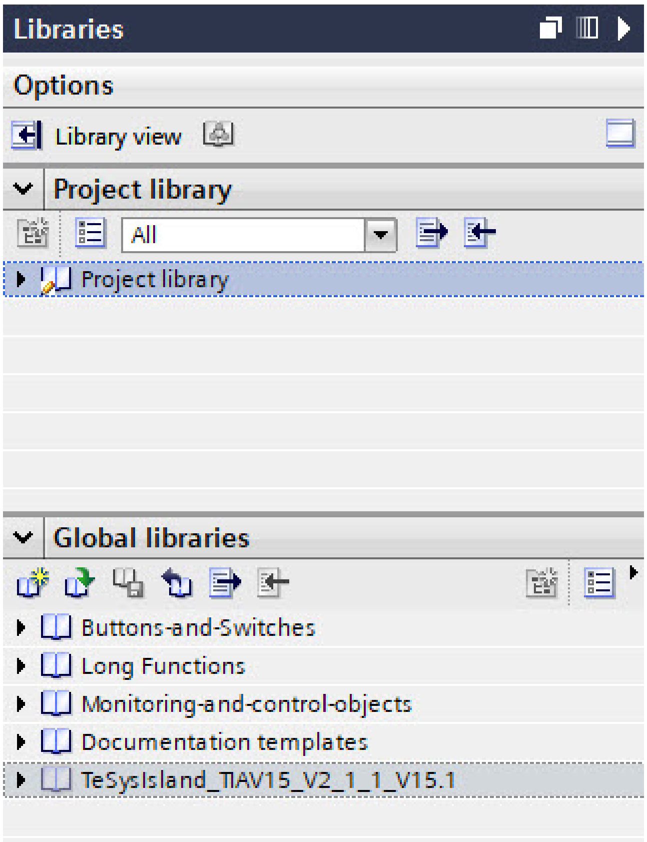

Upgrade

84% Updating new project/library.

Remaining time in seconds: 4

The function block library opens and is located in the Global Libraries pane of the Siemens TIA Portal.

Global Libraries



For additional information, refer to the Siemens TIA Portal information and .

Library Compatibility: The function block library for accessing the TeSys™ island via PROFIBUS DP and PROFINET IO bus couplers are compatible with Siemens TIA Portal V15. All function blocks and data blocks included in the library are compiled for non-optimized access. For execution, the function blocks in the library require the availability of certain System Function Blocks on the SIMATIC PLCs. For a detailed description of these dependencies, refer section Function Block Library, page 49.

Installing SoMove™ Software and TeSys™ island DTM

To generate the AML configuration files for import into the Siemens® TIA Portal, you must install SoMove™ software and the TeSys™ island Device Type Manager (DTM). These tools provide the necessary features to configure the TeSys™ island and download the configuration to the bus coupler.

To download the SoMove software, go to https://www.se.com/us/en/product-range-download/2714-somove/#/software-firmware-tab.

To download the TeSys island DTM library, go to https://www.se.com/us/en/product-range-download/65746-tesys-island/#/software-firmware-tab.

For detailed instructions on the installation and usage of these tools, refer to the *TeSys™ island DTM Online Help Guide*, document number 8536IB1907.

PROFIBUS DP and PROFINET IO Differences

The instructions in this document apply to both PROFIBUS DP and PROFINET IO. Siemens® TIA Portal and SoMove™ software use the same mechanisms for PROFIBUS DP and PROFINET IO bus coupler systems. However, there are minor differences in how devices are addressed between the two bus systems.

In general, most of the differences are addressed by SoMove™ software. SoMove™ software will create the appropriate structures in the exported AML file for each of the bus systems. You might still need to manually set some parameters for the function blocks in the Siemens TIA Portal.

PROFIBUS DP uses a Numeric node ID for addressing. PROFINET IO uses an IP address or a PROFINET name for addressing. If the addresses are assigned in SoMove software, the addresses are automatically transferred in the AML file. Otherwise, the Siemens TIA Portal automatically assigns default addresses.

Integration Procedure

What's in This Part

Prerequisites

The integration procedure assumes that you have done the following:

- Set up a TeSys island device using SoMove software according to the *TeSys island – System, Installation, and Operation Guide*, including the relevant fieldbus parameters

  NOTE:

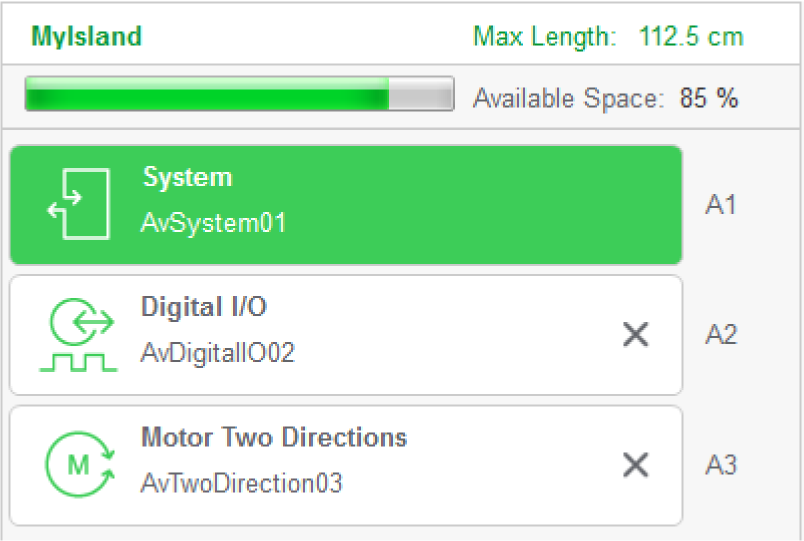
  - For PROFINET, establish an IPv4 address for the device (which applies to both the terminal and fieldbus ports), then configure the island.
  - For PROFIBUS, establish an IPv4 address for the terminal port for configuration, then set a PROFIBUS Node ID in the DTM to establish an address on the PROFIBUS network.
- Installed the appropriate version of Siemens TIA Portal software and TIA Portal Openness as described in Installing the Siemens TIA Portal, page 23
- Set up a project in the Siemens TIA Portal with a PLC as bus controller for the designated bus system (not covered in this document)
- Imported the device description files for the TeSys island and the appropriate function block library into TIA Portal as described in Installing the Device Description Files, page 24 and Integrating the Function Block Library, page 28

Example Items

The TeSys island configuration used in this example includes the following items, as shown in the figure below.

- Digital I/O avatar
- Motor Two Directions avatar

Avatar View



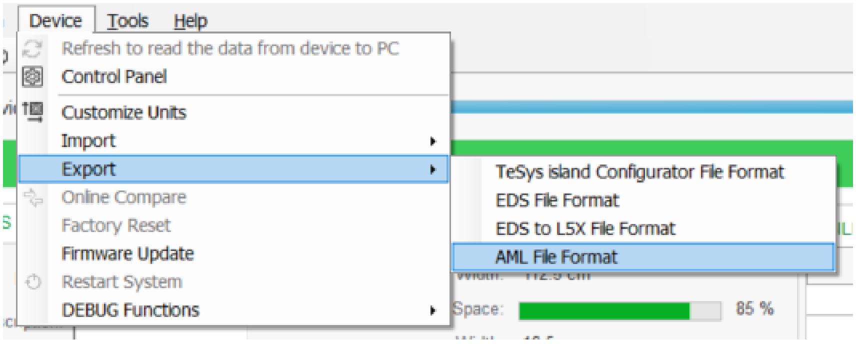
NOTE: The procedures described here refer to a project with a Siemens SIMATIC S7-1200 CPU and PROFINET IO as the bus system. The basic steps remain the same when using a SIMATIC S7-1500 CPU or PROFIBUS DP as a bus system. The notable differences in the handling of PROFIBUS DP are outlined in PROFIBUS DP and PROFINET IO Differences, page 32.

Import the TeSys island Configuration to Siemens TIA Portal

To import the TeSys™ island configuration setup using SoMove™ software, follow these steps:

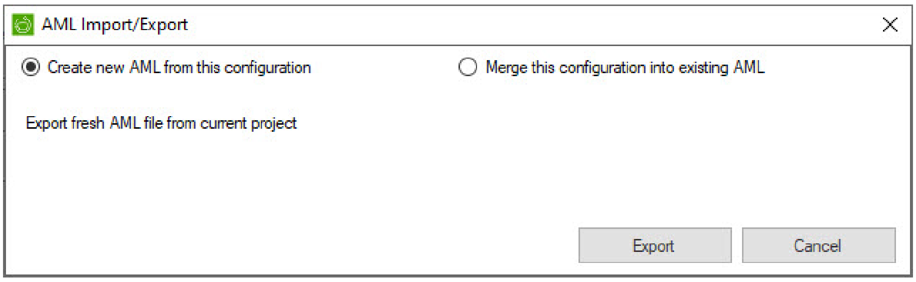
1. Launch SoMove software and open the project you are working on.
2. Launch Siemens TIA Portal and open the project.
3. Select **Export – AML File Format** within the SoMove software Device menu to export the TeSys™ island configuration.

Export AML File



4. Select **Create new AML from this configuration** in the AML Import/Export window and click **Export**.

AML Export

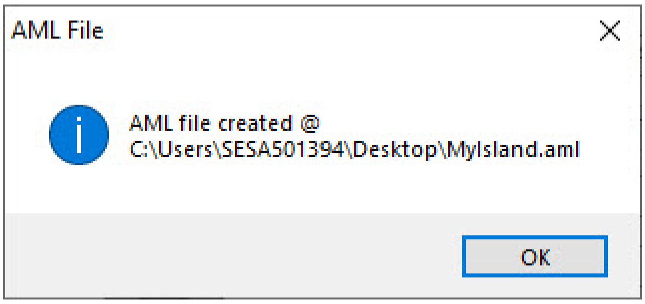

AML Import/Export

● Create new AML from this configuration ○ Merge this configuration into existing AML

Export fresh AML file from current project

5. Select the path where you want to store the file and assign a unique file name for the file.
6. Click **Save**.The AML file will be created at the designated location.
7. Click **OK** in the AML File Export window.

AML File Created

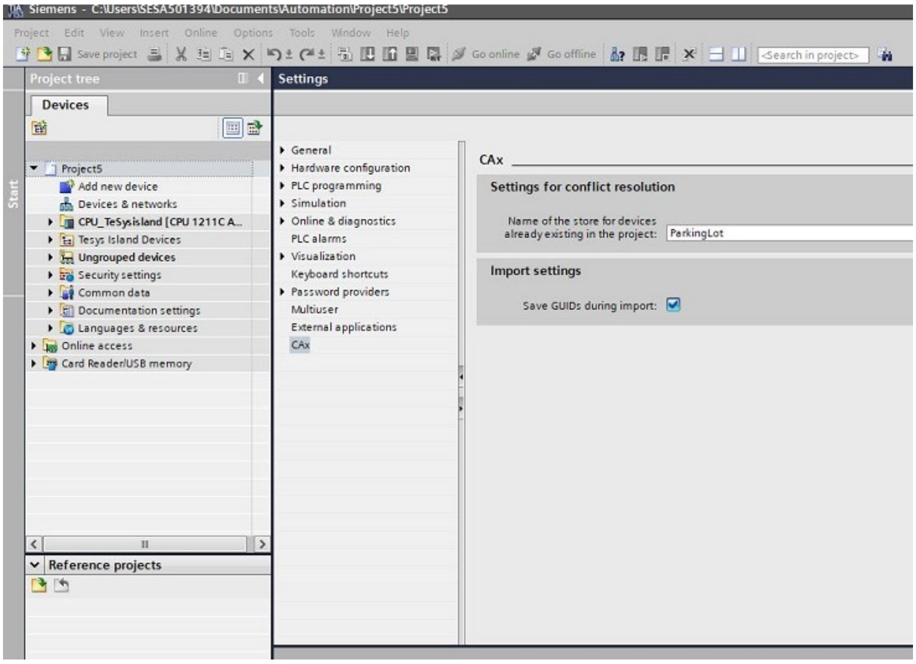

AML File

AML file created @
C:\Users\SESA501394\Desktop\MyIsland.aml

8. Switch to the Siemens TIA Portal software.
9. Select **Setting** in the Options menu within the Siemens TIA Portal project to modify the CAx settings.

10. Select **CAx** from the menu in the Settings window (on the left).

CAx Settings



CAx

Settings for conflict resolution

Name of the store for devices already existing in the project: ParkingLot

Import settings

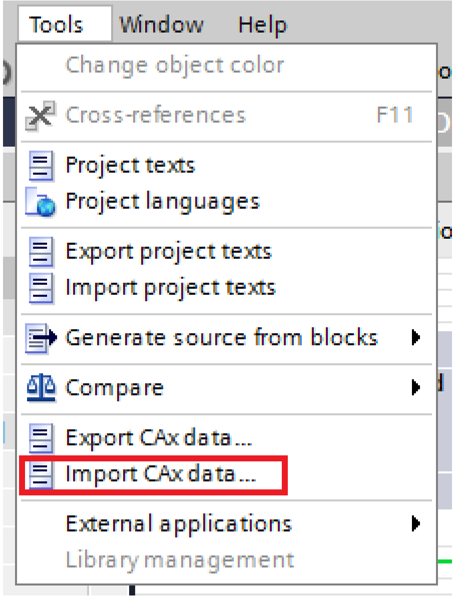
Save GUIDs during import: ☑

11. Check the check-box **Save GUIDs during import:** under Import settings.

12. Select **Import CAx data…** in the Tools menu within the Siemens TIA Portal project to import the TeSys™ island configuration.

    NOTE: If you receive the message prompt “The CAx operation cannot be started. Check if you are logged on as a member of the Siemens TIA Openness group and restart the process.”, the software package for Siemens TIA Portal Openness must be installed. You might also need to perform the procedure in How to Add a User to Siemens® TIA Openness, page 321.

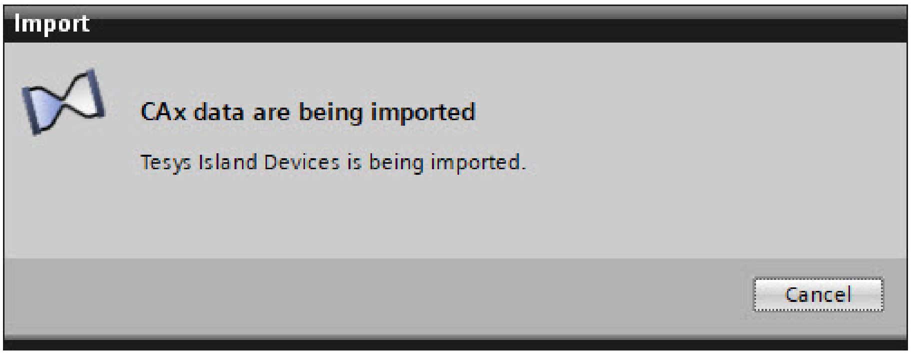
Import CAx Data



13. Select the source path and the AML file you previously created.

14. Click **Open**. The Siemens TIA Portal imports the TeSys island configuration data from the AML file.

CAx File Importing



CAx data are being imported

Tesys Island Devices is being imported.

The TeSys™ island configuration is listed within the Project tree under TeSys Island Devices and in the Networks & Devices view of Siemens TIA Portal.

NOTE:

- If you have already imported a TeSys Island configuration to the Siemens TIA Portal and you need to make changes to it, using the procedure above might lead to loss of the configuration done in Siemens TIA Portal after the first import. The result might be that the device is no longer associated with the PLC, that the I and Q addresses or network addresses of the modules are reset to their default values, or that the device is moved to the “TeSys Island devices” folder again. To avoid these kinds of reversions, use the procedure described in Updating an Existing TeSys™ island Configuration in the TIA Portal, page 44.

Configure TeSys™ island as a Bus Device

What's in This Chapter

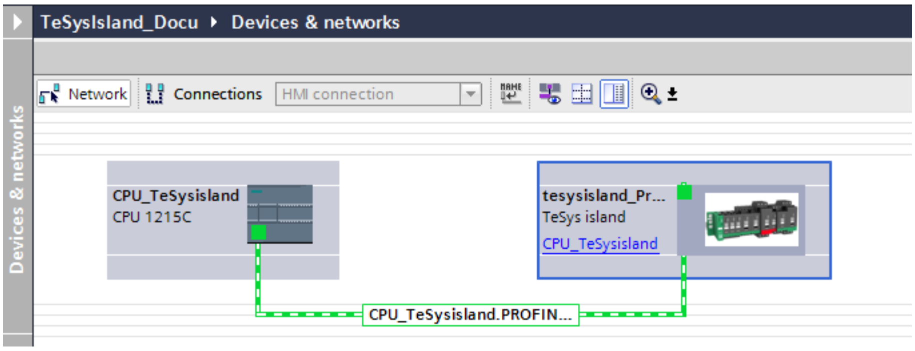
The TeSys™ island bus coupler must be configured before it can be used as a bus device in combination with a Siemens® CPU. The following procedures describe the steps necessary to configure the TeSys island as a bus device.

Assign the TeSys™ island Bus Coupler to a CPU

1. Open the Network view using the Devices & networks entry in the project navigation pane.
2. Left click on the Ethernet interface of the TeSys island bus coupler, then drag it to the Ethernet interface of the CPU.

   When you release the left mouse button, the device is assigned to the PLC. Alternatively, you can click on Not assigned on the TeSys island device and select the appropriate PLC and interface in the dialog box that opens.

   TeSysIsland Devices & networks



3. While assigning the device to the PLC, Siemens® TIA Portal automatically assigns I and Q addresses to the TeSys™ island modules. If the I and Q addresses need to be edited, change to the device view of the bus coupler and enter the desired addresses.

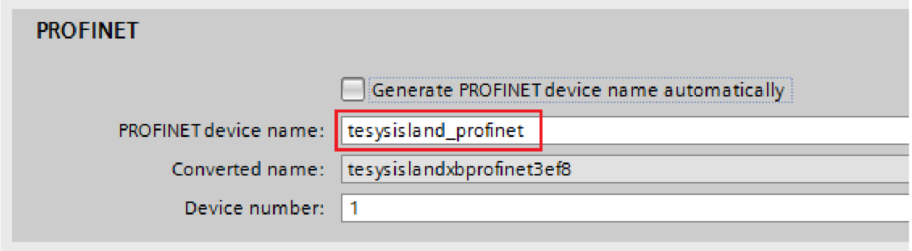
Assign the PROFINET Device Name

The PROFINET IO Controller (in this case, the PLC) requires the PROFINET name of the device to ensure that the PROFINET IO device can be located on the network. To verify whether the name in the Siemens® TIA Portal matches the setting in SoMove™ software, use the following procedure.

1. In SoMove™ software, look for the network name for the bus coupler in the System Avatar settings under **FIELDBUS** → **START-UP** → **Network Name**. This is the device's default PROFINET name.
2. In the Siemens® TIA Portal, look for the PROFINET name for the bus coupler in the PROFINET interface settings of the TeSys™ island bus coupler.

3. If the names are not the same, the PROFINET name must be changed by doing one of the following:
   a. In the Siemens TIA Portal, uncheck the check-box **Generate PROFINET device name automatically** and replace the PROFINET device name with the PROFINET name in use on the TeSys™ island bus coupler.
   b. Assign the PROFINET name used in the Siemens TIA Portal to the TeSys™ island either by using SoMove™ software or the Operation and Maintenance Tool of the TeSys™ island.
   c. Assign the PROFINET name to the TeSys island via PROFINET Discovery and Configuration Protocol (DCP) commands using the built-in Assign PROFINET name functionality of the Siemens TIA Portal (see Siemens TIA Portal information system for more details). The configuration of the TeSys™ island should be read in SoMove™ software once the name has been assigned to ensure it is not overwritten if the SoMove™ project is deployed to theTeSys island again.

PROFINET Device Name



PROFINET

□ Generate PROFINET device name automatically

PROFINET device name: tesysisland_profinet

Converted name: tesysislandxbprofinet3ef8

Device number: 1

For information on how to access the System Avatar settings, see the *TeSys™ island DTM Online Help Guide*, document number 8536IB1907 . For Information on how to access the PROFINET device properties, see the Siemens TIA Portal information system.

NOTE: The network address configured in the TeSys™ island configuration of SoMove™ is transferred with the AML export feature and automatically imported into the Siemens TIA Portal with the AML import feature. However, if no valid network address has been assigned in SoMove (because Dynamic Host Configuration Protocol (DHCP) is used), the TIA Portal automatically assigns a default network address. In this case, ensure that the IP address in Siemens TIA Portal matches the one actually used by the device. For information about where you can check and edit the IP address in the TIA Portal, see the Siemens TIA Portal information system.

Create Function Block Instances for the Avatars

What's in This Chapter

The TeSys island library for Siemens TIA Portal provides several proxy function blocks as well as user defined datatypes (UDTs) to control the avatars with the Siemens CPU and to view status information. The function blocks need to be imported into the project from the TeSys island library. Once the function blocks are in the PLC program, the function block parameters must be created to access the TeSys™ island data.

A dedicated function block controls each type of avatar from within the PLC project. Additionally, there are function blocks that access supplemental information (energy or diagnostic data) with a common interface for all avatars. A third group of function blocks provides access to the System Avatar functions.

For detailed information about the library function block parameters and their usage, refer section Function Block Library, page 49.

> **NOTE:** Starting with version 2.2.0, the function block library also includes master copies of a global data block that defines common constants that the function blocks use (for example, common error and status codes). For the function blocks to work, you must copy this data block to the project. For more information, refer section Function Block Library, page 49.

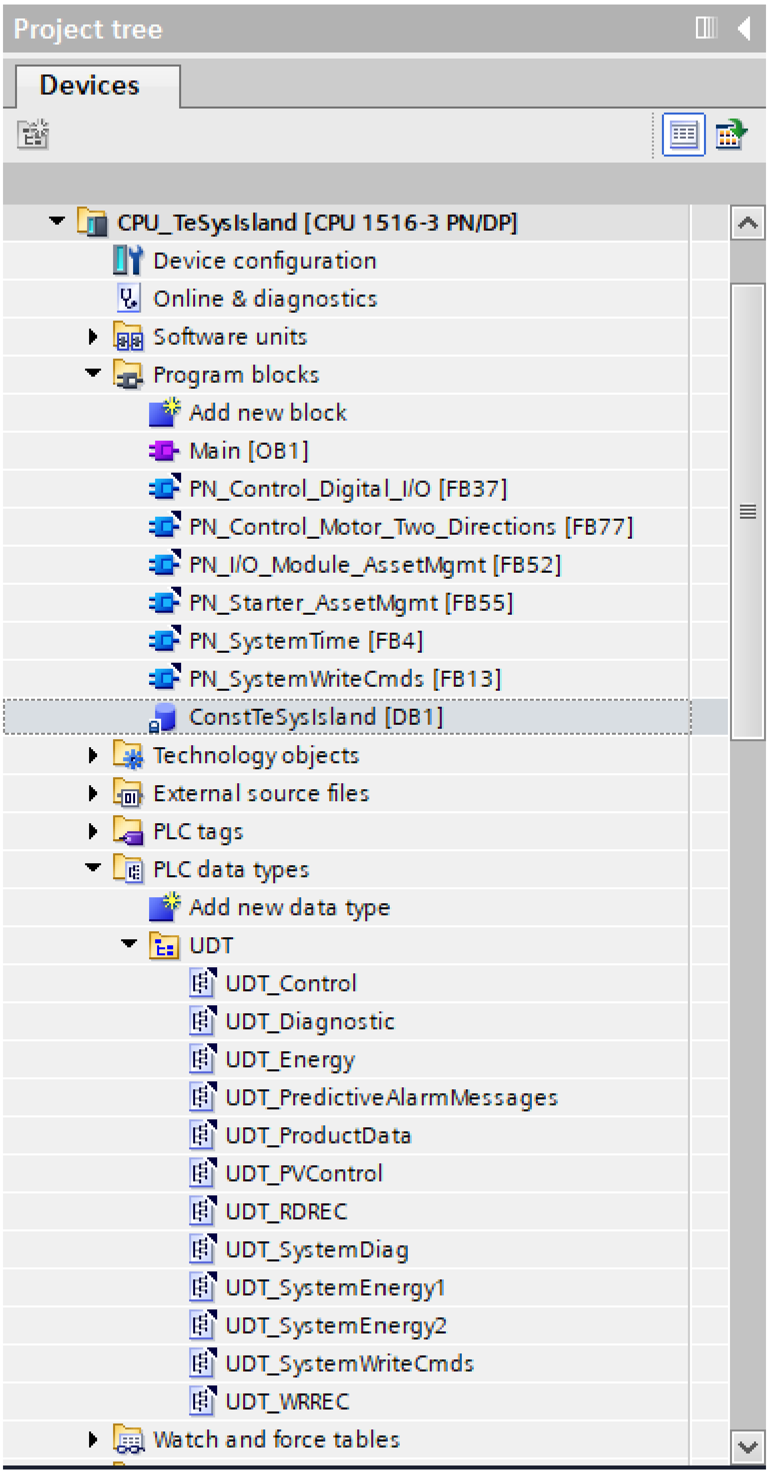
Import a Library Function Block into a TIA Portal Project

To import a TeSys™ island library function block, follow these steps:

- Open the Siemens® TIA Portal project where you want to import the function blocks.
- In the Library pane of Siemens TIA Portal, locate the TeSys™ island function block library and click **Open**.
- From **Master Copies**, select the subfolder appropriate for the PLC type you are using (“DBs S7-1200” for Siemens SIMATIC S7-1200 PLCs, “DBs S7-1500” for Siemens SIMATIC S7-1500 PLCs) and drag the data block “**ConstTeSysIsland**” contained in it to the **Program Blocks** folder in the project tree. The global data block (DB) with constants relevant to the TeSys™ island function blocks is created in the project.
- From **Types**, select the **UDT** folder and drag the contents to the **PLC data types** folder in the Project tree . The UDTs are imported into the project.
- Select any function block types required for the avatars and the functionality you want for your project in the **Types** folder. Drag and drop the function blocks to the **Program blocks** folder in the Project tree. You can also import complete folders of function blocks.

- The imported items should now be listed in the Project tree as shown in the illustration below.

Project tree



Project tree

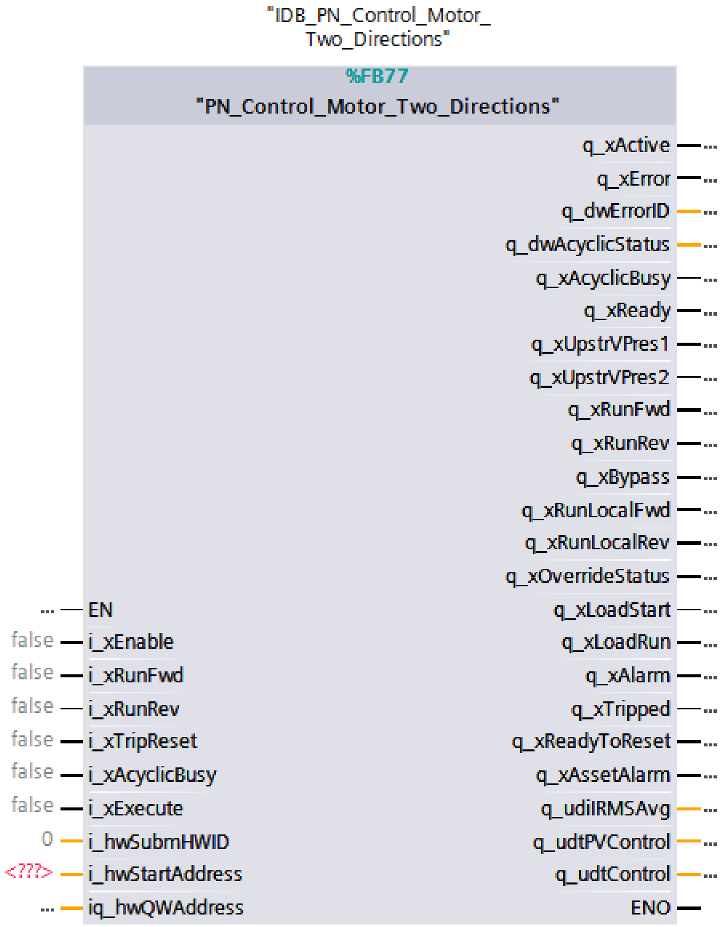
Devices

- CPU_TeSysIsland [CPU 1516-3 PN/DP]
  - Device configuration
  - Online & diagnostics
  - Software units
  - Program blocks
    - Add new block
    - Main [OB1]
    - PN_Control_Digital_I/O [FB37]
    - PN_Control_Motor_Two_Directions [FB77]
    - PN_I/O_Module_AssetMgmt [FB52]
    - PN_Starter_AssetMgmt [FB55]
    - PN_SystemTime [FB4]
    - PN_SystemWriteCmds [FB13]
    - ConstTeSysIsland [DB1]
  - Technology objects
  - External source files
  - PLC tags
  - PLC data types
    - Add new data type
    - UDT
      - UDT_Control
      - UDT_Diagnostic
      - UDT_Energy
      - UDT_PredictiveAlarmMessages
      - UDT_ProductData
      - UDT_PVControl
      - UDT_RDREC
      - UDT_SystemDiag
      - UDT_SystemEnergy1
      - UDT_SystemEnergy2
      - UDT_SystemWriteCmds
      - UDT_WRREC
  - Watch and force tables

Assign the Function Block Parameters

1. Create a call for the function block (within the main object block (OB1)).
2. Assign a name for the function block's instance data block (DB).
3. The function block call is created.

Function Block Instance DB



"IDB_PN_Control_Motor_
Two_Directions"

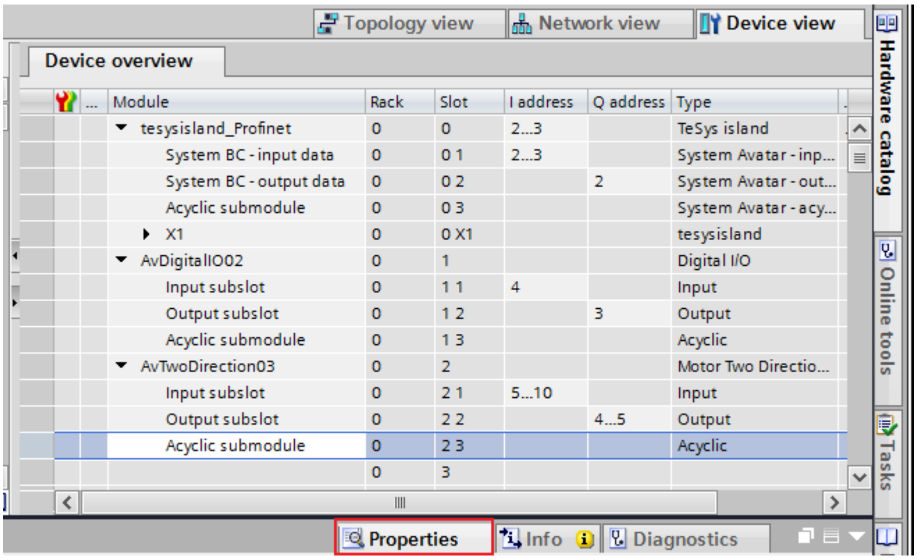
%FB77

"PN_Control_Motor_Two_Directions"

q_xActive — ...
q_xError — ...
q_dwErrorID — ...
q_dwAcyclicStatus — ...
q_xAcyclicBusy — ...
q_xReady — ...
q_xUpstrVPres1 — ...
q_xUpstrVPres2 — ...
q_xRunFwd — ...
q_xRunRev — ...
q_xBypass — ...
q_xRunLocalFwd — ...
q_xRunLocalRev — ...
q_xOverrideStatus — ...
... — EN
false — i_xEnable
false — i_xRunFwd
false — i_xRunRev
false — i_xTripReset
false — i_xAcyclicBusy
false — i_xExecute
0 — i_hwSubmHWID
<???> — i_hwStartAddress
... — iq_hwQWAddress
q_xLoadStart — ...
q_xLoadRun — ...
q_xAlarm — ...
q_xTripped — ...
q_xReadyToReset — ...
q_xAssetAlarm — ...
q_udiIRMSAvg — ...
q_udtPVControl — ...
q_udtControl — ...
ENO —

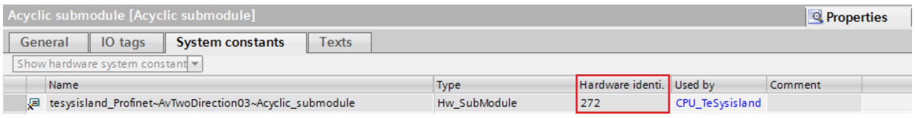
4. For the acyclic data call, the parameter i_hwSubmHWID is required. This parameter refers to the Acyclic sub-module of an avatar or a device. The hardware ID for the acyclic sub-module is located under Properties in the System constants pane in the Device overview of the sub-module.

Acyclic Sub-module



Module	Rack	Slot	I address	Q address	Type
▼ tesysisland_Profinet	0	0	2...3		TeSys island
System BC - input data	0	0 1	2...3		System Avatar - inp...
System BC - output data	0	0 2		2	System Avatar - out...
Acyclic submodule	0	0 3			System Avatar - acy...
▸ X1	0	0 X1			tesysisland
▼ AvDigitalIO02	0	1			Digital I/O
Input subslot	0	1 1	4		Input
Output subslot	0	1 2		3	Output
Acyclic submodule	0	1 3			Acyclic
▼ AvTwoDirection03	0	2			Motor Two Directio...
Input subslot	0	2 1	5...10		Input
Output subslot	0	2 2		4...5	Output
Acyclic submodule	0	2 3			Acyclic
	0	3			

Hardware Identifier

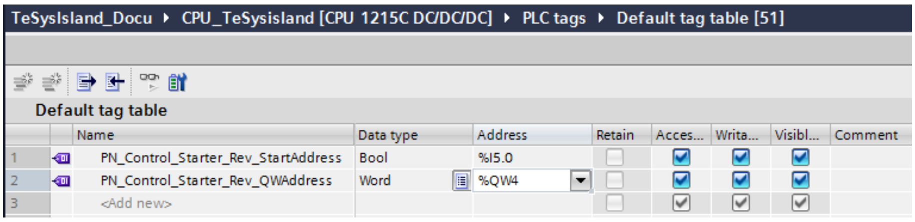

Acyclic submodule [Acyclic submodule]
Properties
General
IO tags
System constants
Texts
Show hardware system constant

Name	Type	Hardware identi.	Used by	Comment
tesysisland_Profinet~AvTwoDirection03~Acyclic_submodule	Hw_SubModule	272	CPU_TeSysisland	

The Siemens® TIA Portal automatically generates names for the hardware IDs of each sub-module. Type the **bus coupler name → module name →submodule name** in the interface of the function block (in this example, tesysisland_Profinet~AvTwoDirection03~Acyclic_submodule). Alternatively, the Hardware Identification (HWID) can be assigned to a user defined variable and provided to the function block.

5. For the cyclic data call, the parameters i_hwStartAddress and iq_hwQWAddress are required. The StartAddress parameter is needed for the function block to find the input data associated with the avatar. To create the i_hwStartAddress parameter, configure a PLC tag (a Boolean data type) with an address at the beginning of the I address range of the cyclic input sub-module and assign it to the interface of the function block.

Assign Function Block Addresses

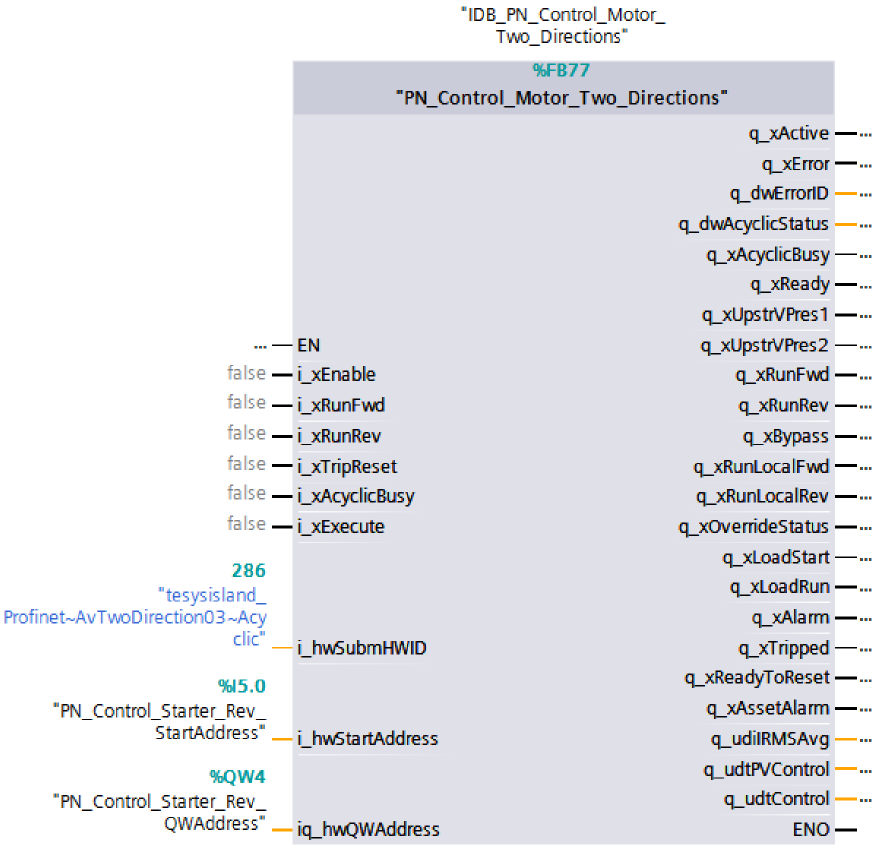

TeSysIsland_Docu ▸ CPU_TeSysisland [CPU 1215C DC/DC/DC] ▸ PLC tags ▸ Default tag table [51]
Default tag table

	Name	Data type	Address	Retain	Acces...	Writa...	Visibl...	Comment
1	PN_Control_Starter_Rev_StartAddress	Bool	%I5.0	□	☑	☑	☑	
2	PN_Control_Starter_Rev_QWAddress	Word	%QW4	□	☑	☑	☑	
3	<Add new>			□	☑	☑	☑	

6. The QWAddress is associated with the Q address of the cyclic output sub-module. To create the “i_hwQWAddress parameter, configure a word-sized PLC tag with the address of the cyclic output sub-module and assign it to the interface of the function block.

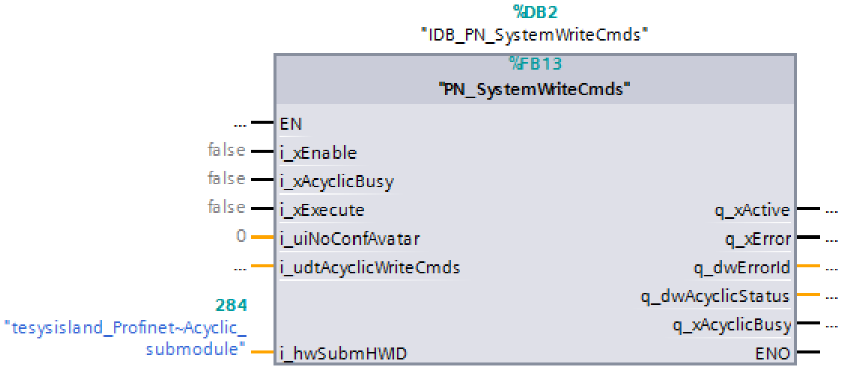
7. All parameters for data transfer are configured. The other parameters can be assigned as needed to the control logic of your PLC program.

Function Block Control Logic



8. Function blocks with the functionality to acyclically write data have an input parameter to specify the data to be written. For example, the System Write commands function block has the input i_udtAcyclicWriteCmds parameter that, in conjunction with i_uiNoConfAvatar parameter, can be used to specify the data to be transferred to the bus coupler.

SystemWrite Function Block



Updating an Existing TeSys™ island Configuration in the TIA Portal

If it is necessary to update an existing TeSys™ island configuration with new avatars or changed avatar positions in TIA Portal, you must import an updated AML file into the TIA Portal and re-configure it. Instead of exporting a new AML file from SoMove™ directly, you can use an AML file merge functionality to preserve data such as I and Q addresses already set in the Siemens® TIA Portal.

This process might revert some of the configuration already done in the TIA Portal after the import. Therefore, it is recommended that you reconfirm all parameters that had been configured for that island in the TIA Portal after the re-import.

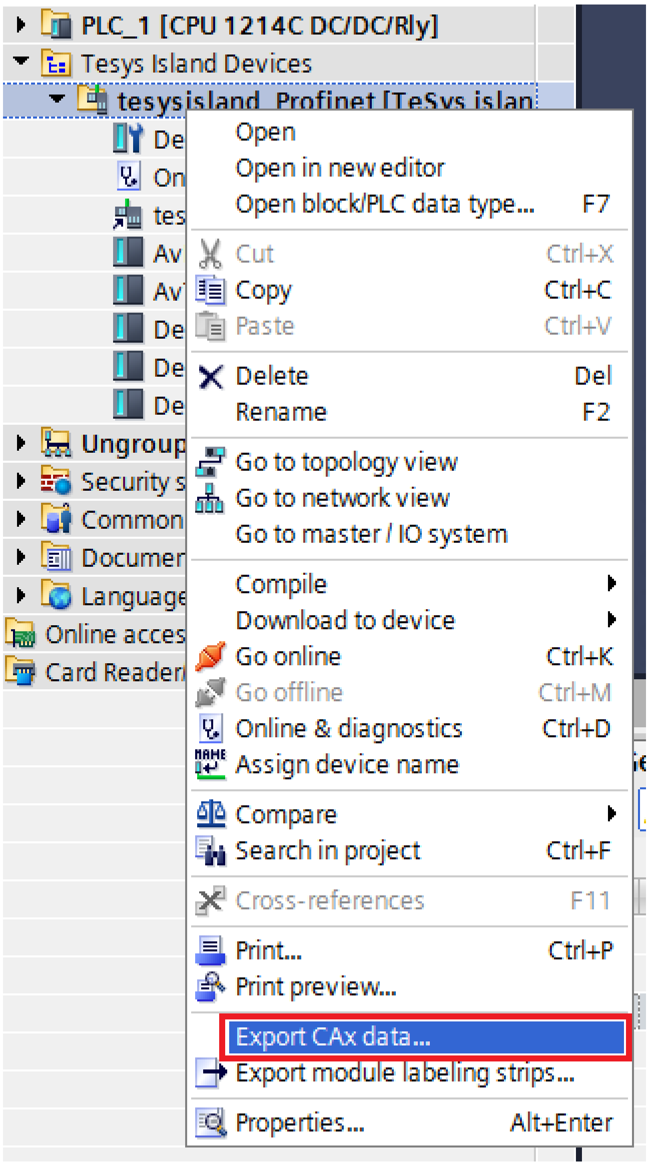
NOTE: Not all configuration changes in SoMove™ software will require an update of the TIA Portal project. The main reason an update would be required is if parameters are changed that also affect the PLC (that is, the bus addresses), or the topology of the TeSys island changes (such as, new or changed avatars or devices).

To update the TeSys island configuration in the Siemens TIA Portal, follow these steps:

1. Open the project in Siemens TIA Portal that contains the TeSys™ island in which the configuration was updated in SoMove™ software.

2. Locate the TeSys island in the Project tree and right-click on it. From the context menu, select **Export CAx data...**.

Export CAx Data



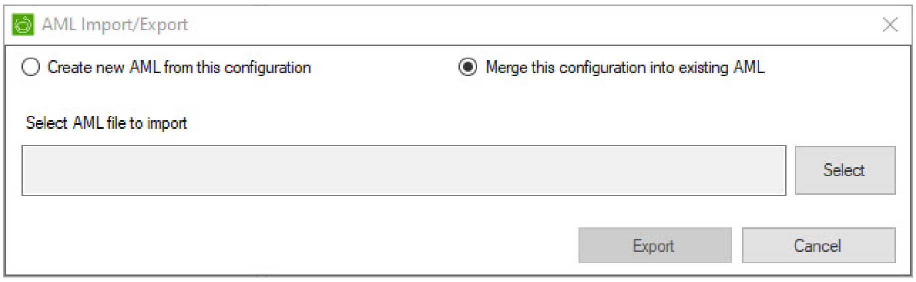
3. Give the file a name and a location to store the file, then click **Save**. The Siemens TIA Portal exports the current configuration of the selected TeSys™ island to that file.

4. Switch to SoMove™ software and open the SoMove project that contains the updated TeSys island configuration.

5. Select **Export** → **AML File Format** from the Device menu.

6. Select the **Merge this configuration into existing AML** option, then click **Export**.

Merge AML File


AML Import/Export

○ Create new AML from this configuration ● Merge this configuration into existing AML

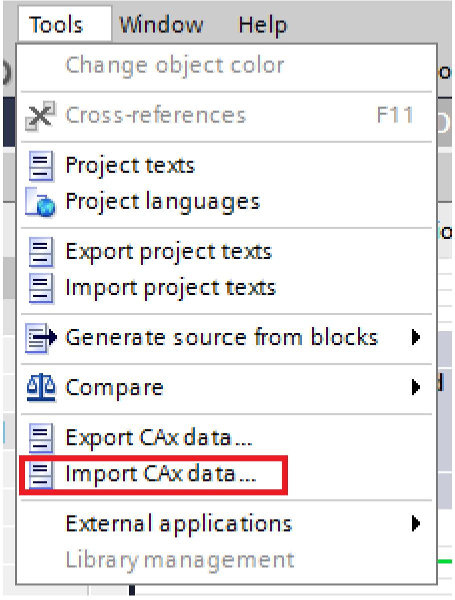
Select AML file to import

7. Select the file you exported from the TIA Portal, then click **Open**. SoMove™ software scans the AML file and identifies the device in the AML file that is being modified.

8. Select the appropriate device and click **Update**. SoMove™ software integrates the changes into the selected AML file, then saves the file.

9. In the Siemens TIA Portal, select **Import CAx data...**.

Import CAx Data

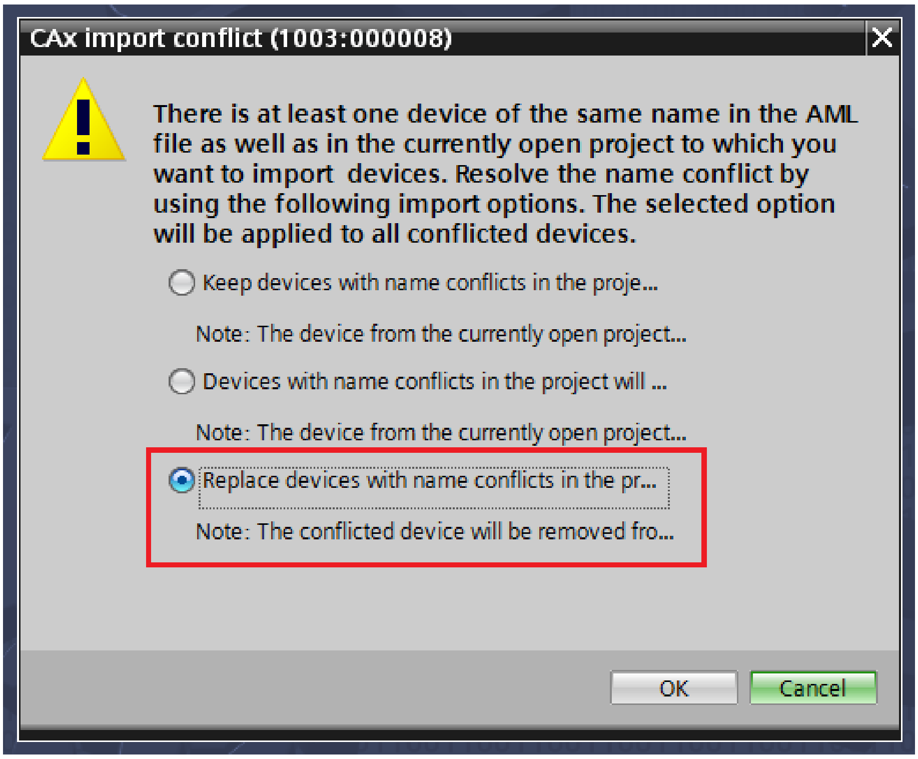


10. Double-click the updated AML file and open it.

11. If you are importing AML files in a project that has existing devices, the TIA Portal will issue a conflict message (such as the message shown below) before importing the devices.

    When updating the TeSys™ island configuration, you can replace the devices in the SoMove™ project directly as long as only one TeSys™ island was exported (see notes below for more information). However, it is recommended that you import the changed device into a separate folder in the TIA Portal and verify the information for the updated device before importing it into the project. Once you have verified the update, click **OK**. The Siemens TIA Portal proceeds to import the updated configuration into the project.

    CAx Import Conflict



CAx import conflict (1003:000008)

There is at least one device of the same name in the AML file as well as in the currently open project to which you want to import devices. Resolve the name conflict by using the following import options. The selected option will be applied to all conflicted devices.

○ Keep devices with name conflicts in the proje...

Note: The device from the currently open project...

○ Devices with name conflicts in the project will ...

Note: The device from the currently open project...

● Replace devices with name conflicts in the pr...

Note: The conflicted device will be removed fro...

NOTE:

- The Siemens TIA Portal also provides an option to export the complete hardware configuration of the TIA Portal project, including all devices (Tools → Export CAx data...). However, due to limitations of the TIA Portal AML import function, re-importing a file that also contains a PLC and directly replacing all devices, will cause the loss of the PLC program.

  If you choose to use the export all functionality to export the project, care needs to be taken when re-integrating the AML file into the TIA Portal. It is recommended that you import the updated versions along with the existing devices, then manually select the devices you want to keep. When you are done, keep all of the devices needed for configuration of the TeSys™ island and import the changes.

- SoMove™ software will retain from the TIA Portal AML file as much information as possible about the updated TeSys™ island. However, the contents of the SoMove™ project will take precedence over changes in the TIA Portal.

  For example, names of various elements such as the avatars, the bus address settings, and the island topology. It is recommended that you maintain these elements in SoMove and update the TIA Portal accordingly. For more information, refer to the *TeSys™ island DTM Online Help Guide*, document number 8536IB1907

- If you have overridden the automatically generated PROFINET name in the Siemens® TIA Portal to match it with the PROFINET name of the TeSys™ island as described in Configure TeSys™ island as a Bus Device, page 37, the TIA Portal will not reflect this in the exported AML file. You must ensure that the name matches the PROFINET name of the TeSys™ island after the updated AML file has been re-imported.

Function Block Library

What’s in This Part

The TeSys™ island IEC 61131-3 function block library for the TIA Portal provides function blocks that interact with the TeSys™ island through the bus system interface of Siemens SIMATIC PLCs for PROFINET IO and PROFIBUS DP. To access the data, both cyclic and acyclic communication is used, depending on the way the data is provided by the island. The function blocks themselves only act as a proxy forwarding information between the PLC program and the TeSys™ island without changing it. The implementation of the control logic for the avatars resides entirely within the island itself.

The following sections give an overview of what is required to use the function blocks in the TIA Portal, which data types are being used by the function blocks, which types of function blocks exist and how access to the data of the TeSys™ island from the PLC program.

In general, the function block interface and functionality for both PROFINET IO and PROFIBUS DP is identical. The implementation of the function blocks varies slightly due to the different inner mechanics of PROFIBUS DP.

Prerequisites

What’s in This Chapter

Installation requirements

To make full use of the function block library for the TIA Portal described here, we recommend you use it in the context of the full TIA Portal integration for TeSys island. Beyond installing the function block library, this usage requires installation of the device descriptions for TeSys island for PROFIBUS DP, PROFINET IO, or both. In addition, the AML import/export functionality for island configuration requires installation of the optional TIA Portal package, “TIA Portal Openness”.

For more information about the usage and installation of the other parts of the TeSys island TIA Portal integration, as well as instructions on how to import the function block library into the TIA Portal, refer section Import a Library Function Block into a TIA Portal Project, page 39.

NOTE: We recommended that you read the sections Software Installation, page 22 and Integration Procedure, page 33, before using the TeSys island function block library for the TIA Portal.

Library compatibility

The function block library described in this document has been developed for the Siemens® TIA Portal V15. It should be compatible with any Siemens SIMATIC S7-1200 or S7-1500 series PLCs that meet the PLC requirements (see below).

The library should be compatible with any firmware version of TeSys™ island and any DTM library version for TeSys™ island sharing the same major and minor version (at the time of writing this documentation, version 2.2.x)

All elements within the library (function blocks and data blocks) are configured to use non-optimized data access.

PLC requirements

In order to work, the function blocks in the TeSys™ island function block library for TIA Portal require specific Extended Instructions to be available on the PLC. The instructions required, and the versions of these instructions that the library was tested for, are as follows:

- IO2MOD V1.1
- DPRD_DAT V1.0
- LOG2GEO V1.2
- RDREC V1.0
- WRREC V1.1

At the time of writing this document, these instructions are all available on any Siemens SIMATIC S7-1200 PLC with firmware version V4.1 and above, or SIMATIC S7-1500 PLC with firmware version V1.0 and above.

For more details on the availability and usage of the instruction listed above, refer to the Siemens TIA Portal Information System and the other Third Party Documentation by Siemens.

NOTE: The resources for acyclic communication through the instructions RDREC and WRREC are limited on the Siemens PLCs. Therefore, all function blocks in the library using acyclic communication provide an interlocking input to inhibit acyclic communication, should that be required. See Acyclic Data, page 69 for more details.

Data Types

What’s in This Chapter

In general, the TeSys™ island function block library for the TIA Portal at the function block interfaces uses the standard simple data types (as defined by IEC 61131-3) provided by the Siemens TIA Portal. The individual data types have been chosen to best match the representation and bit-size of the data on the TeSys™ island. Timestamps are represented using the complex data type “DTL” of the Siemens TIA Portal.

For parametrization of the data exchange, the two special data types “Variant” (cyclic input data) and “HW_IO” (acyclic data) are used as function block inputs where appropriate (see also Data Access, page 67).

For additional information on the usage, data ranges, bit-sizes, and conversion functions for the Siemens TIA Portal data types, refer to the Siemens TIA Portal Information System.

For some purposes, the function blocks use User Defined Types (UDTs) in the Siemens TIA Portal. See UDTs, page 52.

UDTs

To provide structured access to coherent data (for example, the energy data of an avatar), User Defined Types (UDTs) have been created in the Siemens TIA Portal. Some of the UDTs are used at the function block interface, while some are merely for internal use by the function block. The definitions of all required UDTs are included in the TeSys™ island IEC 61131-3 library for the Siemens TIA Portal.

Some UDTs also use structured sub-types for further structuring of the data (for example, individual bits within a bit-field).

The following sections specify the contents of all UDTs included in the library and their sub-structures where appropriate.

NOTE: When importing function blocks to your TIA Portal project, you need only import the function blocks for the functionalities required by your application. The TIA Portal also automatically imports the UDT definitions required by the function blocks. When updating the function block library to a new version, you also need to import the latest definitions of the used UDTs alongside the latest function block versions.

NOTE: Do not alter the definitions of the UDTs described in this document. Otherwise, the function blocks will not work correctly.

UDT_Control

The *Control* UDT is used to transfer the acyclic control data of an avatar, like alarm and trip messages. The interfaces of all PROFINET IO and PROFIBUS DP Control avatar function blocks contain this UDT as a parameter.

Not all Alarm and Trip Message bits are supported by all Control avatars. Details of which bits are supported by the individual avatars are included in the descriptions of the Avatar Control Function Blocks, page 97.

Elements

Element	Data Type	Description
MotorTemperature	UINT	Indicates the temperature of the motor in increments of 0.1 °C. Depending on the temperature sensor type, the range is as follows: • –200 to 850 °C (–328 to 1562 °F) (for PT100) • –200 to 600 °C (–328 to 1112 °F) (for PT1000) • –60 to 180 °C (–76 to 356 °F) (for NI 100/1000)
SIL[7] Group	USINT	Indicates the number of the SIL group.
ThermalCapacity	USINT	Provides the percentage (%) of the thermal capacity of the motor.
AlarmMsg1	STRUCT	The alarm messages of a Control avatar. See Struct AlarmMsg1, page 53.
AlarmMsg2	STRUCT	The alarm messages of a Control avatar. See Struct AlarmMsg2, page 54.
TripMsg1	STRUCT	The trip messages of a Control avatar. See Struct TripMsg1, page 54.
TripMsg2	STRUCT	The trip messages of a Control avatar. See Struct TripMsg2, page 55.
TimeToTrip	UINT	Estimated time (s) before a Thermal Overload trip will occur given the current conditions.
TimeToReset	UINT	Estimated time to wait before a reset could acknowledge a Thermal Overload trip.
PAStatusReg1	Struct	Predictive Alarm Status Register 1

Struct AlarmMsg1

Element	Data Type	BIT	Description
(reserved)	BOOL	8	
(reserved)	BOOL	9	
(reserved)	BOOL	10	
(reserved)	BOOL	11	
(reserved)	BOOL	12	
(reserved)	BOOL	13	
(reserved)	BOOL	14	
(reserved)	BOOL	15	
(reserved)	BOOL	0	
(reserved)	BOOL	1	
GroundCurrent	BOOL	2	The avatar has detected the conditions to cause a ground current event.
ThermalOverload	BOOL	3	The avatar thermal capacity has increased above the thermal overload level.
(reserved)	BOOL	4	
Jam	BOOL	5	The avatar has detected the conditions to cause a jam event.
CurrentPhaseUnbalance	BOOL	6	The avatar has detected the conditions to cause a current phase unbalance event.
UnderCurrent	BOOL	7	The avatar has detected the conditions to cause an undercurrent event.

[7] Safety Integrity Level according to standard IEC 61508.

Struct AlarmMsg2

Element	Data Type	BIT	Description
(reserved)	BOOL	8	
(reserved)	BOOL	9	
(reserved)	BOOL	10	
(reserved)	BOOL	11	
(reserved)	BOOL	12	
(reserved)	BOOL	13	
(reserved)	BOOL	14	
(reserved)	BOOL	15	
(reserved)	BOOL	0	
(reserved)	BOOL	1	
(reserved)	BOOL	2	
OverCurrent	BOOL	3	The avatar has detected the conditions to cause an overcurrent event.
(reserved)	BOOL	4	
(reserved)	BOOL	5	
MotorOverheat	BOOL	6	The motor temperature has increased above the motor overheat level.
(reserved)	BOOL	7	

Struct TripMsg1

Element	Data Type	BIT	Description
Stall	BOOL	8	The avatar has detected the conditions to cause a stall trip event.
(reserved)	BOOL	9	
(reserved)	BOOL	10	
(reserved)	BOOL	11	
(reserved)	BOOL	12	
(reserved)	BOOL	13	
(reserved)	BOOL	14	
(reserved)	BOOL	15	
(reserved)	BOOL	0	
(reserved)	BOOL	1	
GroundCurrent	BOOL	2	The avatar has detected the conditions to cause a ground current trip event.
ThermalOverload	BOOL	3	The avatar thermal capacity has increased above 100%.
LongStart	BOOL	4	The avatar has detected the conditions to cause a long start trip event.
Jam	BOOL	5	The avatar has detected the conditions to cause a jam trip event.
CurrentPhaseUnbalance	BOOL	6	The avatar has detected the conditions to cause a current phase unbalance trip event.
UnderCurrent	BOOL	7	The avatar has detected the conditions to cause an undercurrent trip event.

Struct TripMsg2

Element	Data Type	BIT	Description
(reserved)	BOOL	8	
(reserved)	BOOL	9	
(reserved)	BOOL	10	
(reserved)	BOOL	11	
(reserved)	BOOL	12	
(reserved)	BOOL	13	
(reserved)	BOOL	14	
(reserved)	BOOL	15	
(reserved)	BOOL	0	
(reserved)	BOOL	1	
PhaseConfig	BOOL	2	The avatar has detected the conditions to cause a phase configuration trip event.
OverCurrent	BOOL	3	The avatar has detected the conditions to cause an overcurrent trip event.
CurrentPhaseLoss	BOOL	4	The avatar has detected the conditions to cause a current phase loss trip event.
CurrentPhaseReversal	BOOL	5	The avatar has detected the conditions to cause a current phase reversal trip event.
MotorOverheat	BOOL	6	The motor temperature has increased above the motor overheat trip level.
(reserved)	BOOL	7	

Struct PAStatusReg1

Element	Data Type	BIT	Description
PA8Status	BOOL	8	Status of Predictive Alarm Bit 8
PA9Status	BOOL	9	Status of Predictive Alarm Bit 9
(reserved)	BOOL	10	
(reserved)	BOOL	11	
(reserved)	BOOL	12	
(reserved)	BOOL	13	
(reserved)	BOOL	14	
(reserved)	BOOL	15	
PA0Status	BOOL	0	Status of Predictive Alarm Bit 0
PA1Status	BOOL	1	Status of Predictive Alarm Bit 1
PA2Status	BOOL	2	Status of Predictive Alarm Bit 2
PA3Status	BOOL	3	Status of Predictive Alarm Bit 3
PA4Status	BOOL	4	Status of Predictive Alarm Bit 4
PA5Status	BOOL	5	Status of Predictive Alarm Bit 5
PA6Status	BOOL	6	Status of Predictive Alarm Bit 6
PA7Status	BOOL	7	Status of Predictive Alarm Bit 7

UDT_Diagnostic

The *Diagnostic* UDT is used to transfer the acyclic diagnostic data of an avatar, like Irms, alarm and trip information. The interfaces of the PROFINET IO and PROFIBUS DP generic diagnostic avatar function blocks contain this UDT as a parameter.

Elements

Element	Data Type	Description
MaxAvgIRMS	UDINT	Maximum average Irms current value.
MaxAvgIRMSTImeStamp	DTL	Date and time when the maximum average value of the Irms current was recorded.
IRMSPhase1	UDINT	Value of the phase L1 Irms. (Unit: mA).
IRMSPhase2	UDINT	Value of the phase L2 Irms. (Unit: mA).
IRMSPhase3	UDINT	Value of the phase L3 Irms. (Unit: mA).
AlarmCnt	STRUCT	Alarm counters. See Struct AlarmCnt, page 56.
TripCnt	STRUCT	Trip counters. See Struct TripCnt, page 56.
TripRecReg1	STRUCT	Record 1 of a trip event. See Struct TripRecRegX, page 57.
TripRecReg2	STRUCT	Record 2 of a trip event. See Struct TripRecRegX, page 57.
TripRecReg3	STRUCT	Record 3 of a trip event. See Struct TripRecRegX, page 57.
TripRecReg4	STRUCT	Record 4 of a trip event. See Struct TripRecRegX, page 57.
TripRecReg5	STRUCT	Record 5 of a trip event. See Struct TripRecRegX, page 57.

Struct AlarmCnt

Element	Data Type	Description
ThermOverload	UInt	Counter of advisories related to thermal overload protection.
Jam	UInt	Counter of advisories related to jam protection.
Undercurrent	UInt	Counter of advisories related to undercurrent protection.
Overcurrent	UInt	Counter of advisories related to overcurrent protection.
IPhaseUnbal	UInt	Counter of advisories related to phase unbalance protection.
GroundCurrent	UInt	Counter of advisories related to ground current protection.
Overheat	UInt	Counter of motor overheat events.
AllAlarms	UInt	Counter of advisories related to protections.

Struct TripCnt

Element	Data Type	Description
ThermOverload	UInt	Counter of trips related to thermal overload protection.
Jam	UInt	Counter of trips related to jam protection.
Undercurrent	UInt	Counter of trips related to undercurrent protection.
LongStart	UInt	Counter of trips related to long start protection.
Overcurrent	UInt	Counter of trips related to overcurrent protection.
Overheat	UInt	Counter of motor overheat trip events.
Stall	UInt	Counter of trips related to stall protection.
IPhaseUnbal	UInt	Counter of trips related to phase unbalance protection.
PhaseConfT	UInt	Counter of trips related to phase configuration protection.
GroundCurrent	UInt	Counter of trips related to ground current protection.

Struct TripCnt (Continued)

Element	Data Type	Description
PhaseReversal	UInt	Counter of trips related to phase reversal protection.
CurrentPhaseLoss	UInt	Counter of trips related to phase loss protection.
AllTrips	UInt	Counter of trips related to protections.

Struct TripRecRegX

Element	Data Type	Description
DTL	DTL	TIA timestamp format
TTTT	WORD	Trip event identifier 0000—No Detected Event 0001—Thermal Overload 0002—Motor Overheat 0003—Jam 0004—Undercurrent 0005—Long Start 0006—Overcurrent 0007—Stall 0008—Ground Current 0009—Current Phase Reversal 0010—Phase Configuration 0011—Current Phase Unbalance 0012—Current Phase Loss

UDT_Energy

The *Energy* UDT is used to transfer the acyclic energy data of an avatar, like power and active/reactive energy information. The interfaces of the PROFINET IO and PROFIBUS DP generic energy avatar function blocks contain this UDT as a parameter.

Elements

Element	Data Type	Description
TotActPower	DINT	Total active power for the avatar. (Unit: W)
MaxTotActPow	DINT	Maximum value of the active power value for the avatar (Unit: W)
MaxTotActPowTimestamp	DTL	Date and time when the maximum active power value was recorded.
TotReactPower	DINT	Total reactive power value of the avatar (Unit: VAR)
MaxTotReactPow	DINT	Maximum value of the reactive power value for the avatar (Unit: VAR)
MaxTotReactPowTimestamp	DTL	Date and time when the maximum reactive power value was recorded.
TruePowFact	USINT	True power factor value.
MinTruePowFact	USINT	Minimum value of the true power factor.
MaxTruePowFact	USINT	Maximum value of the true power factor.
MinTruePowFactTimestamp	DTL	Date and time when the minimum true power factor was recorded.

Elements (Continued)

Element	Data Type	Description
MaxTruePowFactTimestamp	DTL	Date and time when the maximum value of the true power factor was recorded.
TotActEnergy	UDINT	Total active energy value for all avatars in the system with Energy Monitoring Enabled. (Unit: Wh)
TotReactEnergy	UDINT	Total reactive energy value for all avatars in the system with Energy Monitoring Enabled. (Unit VARh)
ToU_TotActEnergyCh1	UDINT	Channel 1: ToU (Time of Use) total active energy value.
ToU_TotActEnergyCh2	UDINT	Channel 2: ToU (Time of Use) total active energy value.
ToU_TotActEnergyCh3	UDINT	Channel 3: ToU (Time of Use) total active energy value.
ToU_TotActEnergyCh4	UDINT	Channel 4: ToU (Time of Use) total active energy value.
ToU_TotReactEnergyCh1	UDINT	Channel 1: ToU (Time of Use) total reactive energy value.
ToU_TotReactEnergyCh2	UDINT	Channel 2: ToU (Time of Use) total reactive energy value.
ToU_TotReactEnergyCh3	UDINT	Channel 3: ToU (Time of Use) total reactive energy value.
ToU_TotReactEnergyCh4	UDINT	Channel 4: ToU (Time of Use) total reactive energy value.

UDT_PVControl

The *PVControl* UDT is used to transfer an avatar's cyclic process variable data, like analog values from sensors and discrete signals from switches. The interfaces of the PROFINET IO and PROFIBUS DP application avatar function blocks contain this UDT as a parameter.

Elements

Element	Data Type	Description
PVInput0	INT	Process variable input for analog value
PVInput1	INT	Process variable input for analog value
PVInput2	INT	Process variable input for analog value
PVInput3	INT	Process variable input for analog value
PVInput4	INT	Process variable input for analog value
PVSwitch0	BOOL	Process variable switch for discrete signals
PVSwitch1	BOOL	Process variable switch for discrete signals
PVSwitch2	BOOL	Process variable switch for discrete signals
PVSwitch3	BOOL	Process variable switch for discrete signals
PVSwitch4	BOOL	Process variable switch for discrete signals
Reserve	BOOL	
Reserve	BOOL	
Reserve	BOOL	
PVControlIn0	BOOL	Status of PV Control-Input 0
PVControlIn1	BOOL	Status of PV Control-Input 1
Reserve	BOOL	
Reserve	BOOL	
Reserve	BOOL	
Reserve	BOOL	

Elements (Continued)

Element	Data Type	Description
Reserve	BOOL	
Reserve	BOOL	

UDT_PredictiveAlarmMessages

The *PredictiveAlarmMessages* UDT is used as a template when importing the predictive alarm status messages from SoMove™ software into the TIA Portal. This UDT contains the predictive alarm status messages for one avatar. See the *TeSys™ island Quick Start Guide for PROFINET and PROFIBUS Applications*, document number 8536IB1916 for more details on importing predictive alarm status messages.

Elements

Element	Data Type	Description
PAStatusMessage0	STRING[150]	Message associated with Predictive Alarm Bit 0
PAStatusMessage1	STRING[150]	Message associated with Predictive Alarm Bit 1
PAStatusMessage2	STRING[150]	Message associated with Predictive Alarm Bit 2
PAStatusMessage3	STRING[150]	Message associated with Predictive Alarm Bit 3
PAStatusMessage4	STRING[150]	Message associated with Predictive Alarm Bit 4
PAStatusMessage5	STRING[150]	Message associated with Predictive Alarm Bit 5
PAStatusMessage6	STRING[150]	Message associated with Predictive Alarm Bit 6
PAStatusMessage7	STRING[150]	Message associated with Predictive Alarm Bit 7
PAStatusMessage8	STRING[150]	Message associated with Predictive Alarm Bit 8
PAStatusMessage9	STRING[150]	Message associated with Predictive Alarm Bit 9

UDT_ProductData

The *ProductData* UDT is used to provide an avatar’s product data as one coherent variable. The interfaces of the PROFINET IO and PROFIBUS DP System Asset Management and the Device Asset Management avatar function blocks use this UDT as a parameter.

Elements

Element	Data Type	Description
VendorName	STRING[20]	Vendor name.
ProductCode	STRING[32]	Vendor specific product code.
MajorMinorRev	STRING[7]	Revision number of the application firmware in format xxx.yyy.
VendorURL	STRING[64]	Vendor URL.
ProductName	STRING[32]	Product name.
ModelName	STRING[20]	Model name.
SerialNumber	STRING[20]	Serial number of the module.

UDT_RDREC

The *RDREC* UDT is used to store status information of an acyclic read request internally.

Elements

Element	Data Type	Description
xQ_RTrig	BOOL	Rising edge result, to transfer data to function block variables
qxValid	BOOL	Acyclic read request result is VALID
qxBusy	BOOL	Acyclic read function is busy
qxError	BOOL	Acyclic read request ends with ERROR
qdwStatus	DWORD	Status result of acyclic read function
quiLen	UINT	Length of read data as feedback value.
ixStartRead	BOOL	Enable/Start acyclic read function

UDT_SystemDiag

The *SystemDiag* UDT is used to transfer the acyclic diagnostic data of the system. The interfaces of the PROFINET IO and PROFIBUS DP SystemDiag avatar function blocks contain this UDT as a parameter.

Elements

Element	Data Type	Description
CntErrComm	UINT	Number of detected errors for the fieldbus communication.
CntAllAlarms	UINT	Number of detected alarms for the system.
CntSystemEvt	UINT	Number of detected minor events for the system.
EvtRecReg1	STRUCT	Information on a detected minor event – register 1. See Struct EvtRecRegX, page 61.
EvtRecReg2	STRUCT	Information on a detected minor event – register 2. See Struct EvtRecRegX, page 61.
EvtRecReg3	STRUCT	Information on a detected minor event – register 3. See Struct EvtRecRegX, page 61.
EvtRecReg4	STRUCT	Information on a detected minor event – register 4. See Struct EvtRecRegX, page 61
EvtRecReg5	STRUCT	Information on a detected minor event – register 5. See Struct EvtRecRegX, page 61.
SILStopMsgGrp1	STRUCT	Status for SIL Stop 0[8] function of SIL group 1. See Struct SILStopMsgX, page 61.
SILStopMsgGrp2	STRUCT	Status for SIL Stop 0 function of SIL group 2. See Struct SILStopMsgX, page 61.
SILStopMsgGrp3	STRUCT	Status for SIL Stop 0 function of SIL group 3. See Struct SILStopMsgX, page 61.
SILStopMsgGrp4	STRUCT	Status for SIL Stop 0 function of SIL group 4. See Struct SILStopMsgX, page 61.
SILStopMsgGrp5	STRUCT	Status for SIL Stop 0 function of SIL group 5. See Struct SILStopMsgX, page 61.
SILStopMsgGrp6	STRUCT	Status for SIL Stop 0 function of SIL group 6. See Struct SILStopMsgX, page 61.
SILStopMsgGrp7	STRUCT	Status for SIL Stop 0 function of SIL group 7. See Struct SILStopMsgX, page 61.

[8] Safety Integrity Level according to standard IEC 61508. Stop categories according to EN/IEC 60204-1.

Elements (Continued)

Element	Data Type	Description
SILStopMsgGrp8	STRUCT	Status for SIL Stop 0 function of SIL group 8. See Struct SILStopMsgX, page 61.
SILStopMsgGrp9	STRUCT	Status for SIL Stop 0 function of SIL group 9. See Struct SILStopMsgX, page 61.
SILStopMsgGrp10	STRUCT	Status for SIL Stop 0 function of SIL group 10. See Struct SILStopMsgX, page 61.

Struct EvtRecRegX

Element	Data Type	Description
DTL	DTL	TIA timestamp format
FFFF	WORD	Detected minor event identifier. 0000: No Detected Minor Event. 0001: No module in the island. 0002: Number of physical devices detected in the island is beyond the limit allowed. 0003: Modules mismatch. 0004: Island control power supply voltage fluctuation.

Struct SILStopMsgX

Element	Data Type	Description
SILGrpNotInSysConf	BOOL	SIL Group not present in system configuration
Reserved	BOOL	SIL Group impacted by Avatar Device event
SILCmdRecSILStarterNotOpen	BOOL	SIL Group stop command received; SIL starters not open yet
SILCmdOkSILStarterOpened	BOOL	SIL Group stop command successfully issued; all SIL starters are open
SILCmdTo1Terminal	BOOL	SIL Group stop command issued to only 1 SIL interface module (SIM) input channel (jumper or SIM input wiring is causing an issue) but SIL starters did successfully open
NormalOperation	BOOL	Normal operation, SIL starters can be open or closed
<Reserved>	BOOL	Byte-filler
<Reserved>	BOOL	Byte-filler

UDT_SystemEnergy1

The *SystemEnergy1* UDT is used to transfer the acyclic voltage energy data of the System avatar. The interfaces of the PN_SystemEnergy and DP_SystemEnergy function blocks contain this UDT as a parameter.

Elements

Element	Data Type	Description
VrmsAverg	UINT	Average Vrms voltage on three phases. (Unit: V)
VrmsMaxAvg	UINT	This register indicates maximum voltage the avatar has measured in its lifetime. (Unit: V)
VtimeMaxAvg	DTL	Timestamp of maximum average.
VrmsPhase1	UINT	Average Vrms voltage between L1 and neutral. (Unit: V)
VrmsPhase2	UINT	Average Vrms voltage between L2 and neutral. (Unit: V)

Elements (Continued)

Element	Data Type	Description
VrmsPhase3	UINT	Average Vrms voltage between L3 and neutral. (Unit: V)
VrmsL1L2	UINT	Average Vrms voltage between L1 and L2. (Unit: V)
VrmsL2L3	UINT	Average Vrms voltage between L2 and L3. (Unit: V)
VrmsL3L1	UINT	Average Vrms voltage between L3 and L1. (Unit: V)
VUnbalPerc	USINT	Unbalance voltage in percent (%).
VUnbalMax	USINT	Maximum unbalance voltage in percent (%).
VtimeUnbalMax	DTL	Date and time of maximum unbalance voltage.
PhaseSequ	Byte	Phase sequence (ABC or ACB).
Frequency	USINT	Main power voltage frequency (line frequency as measured on phase 1). (Unit: Hz)
VDipRecordReg1	STRUCT	Voltage Dip record register 1. See Struct VDipRecordRegX, page 62.
VDipRecordReg2	STRUCT	Voltage Dip record register 2. See Struct VDipRecordRegX, page 62.
VDipRecordReg3	STRUCT	Voltage Dip record register 3. See Struct VDipRecordRegX, page 62.
VDipRecordReg4	STRUCT	Voltage Dip record register 4. See Struct VDipRecordRegX, page 62.
VDipRecordReg5	STRUCT	Voltage Dip record register 5. See Struct VDipRecordRegX, page 62.
VDipCnt	UINT	Voltage dip counter value.

Struct VDipRecordRegX

Element	Data Type	Description
uiMagnitude	UINT	Magnitude of the voltage register.
dtlStartDate	DTL	Start timestamp of the voltage register.
dtlStopDate	DTL	Stop timestamp of the voltage register.

UDT_SystemEnergy2

The *SystemEnergy2* UDT is used to transfer the acyclic energy and power data of the System avatar. The interfaces of the PN_SystemEnergy and DP_SystemEnergy function blocks contain this UDT as a parameter.

Elements

Element	Data Type	Description
VSwellRecordReg1	STRUCT	See Struct VSwellRecordRegX, page 63.
VSwellRecordReg2	STRUCT	See Struct VSwellRecordRegX, page 63.
VSwellRecordReg3	STRUCT	See Struct VSwellRecordRegX, page 63.
VSwellRecordReg4	STRUCT	See Struct VSwellRecordRegX, page 63.
VSwellRecordReg5	STRUCT	See Struct VSwellRecordRegX, page 63.
VSwellCnt	UINT	Voltage swell counter value.
TotActPow	DINT	Total active power for the system. (Unit: W)
MaxTotActPow	DINT	Maximum value of the active power for the system. (Unit: W)
MaxTotActPowTimestamp	DTL	Date and time when the maximum active power value was recorded.
TotReactPower	DINT	Total reactive power for the system. (Unit: VAR)
MaxTotReactPow	DINT	Maximum value of the reactive power for the system. (Unit: VAR)

Elements (Continued)

Element	Data Type	Description
MaxTotReactPowTimestamp	DTL	Date and time when the maximum reactive power value was recorded.
TruePowFact	USINT	True power factor value.
MinTruePowFact	USINT	True minimum power factor value.
MaxTruePowFact	USINT	True maximum power factor value.
MinTruePowFactTimestamp	DTL	Date and time when the minimum true power factor value was recorded.
MaxTruePowFactTimestamp	DTL	Date and time when the maximum power factor value was recorded.
TotActEnergy	UDINT	Total active energy value for all avatars in the system with Energy Monitoring Enabled. (Unit: Wh)
TotReactEnergy	UDINT	Total reactive energy value for all avatars in the system with Energy Monitoring Enabled. (Unit: VARh)

Struct VSwellRecordRegX

Element	Data Type	Description
uiMagnitude	UINT	Magnitude of the voltage register.
dtlStartDate	DTL	Start timestamp of the voltage register.
dtlStopDate	DTL	Stop timestamp of the voltage register.

UDT_SystemWriteCmds

The *SystemWriteCmds* UDT is used to transfer the acyclic system reset commands and avatar energy preset data. The interfaces of the PN_ SystemWriteCmds and DP_SystemWriteCmds function blocks contain this UDT as a parameter.

Elements

Element	Data Type	Description
stSystemWrite	STRUCT	The acyclic write commands of the system. See Struct SystemWrite, page 63.
aAvatarWrite	Array[1..20] of STRUCT	Contains an array of the acyclic write commands of all 20 possible avatars. See Struct AvatarWrite, page 64.

Struct SystemWrite

Element	Data Type	Description
usiResetVoltageDipCnt	USINT	If this input is set to 1, the value of the parameter VDipCnt is reset.
usiResetVoltageSwellCnt	USINT	If this input is set to 1, the value of the parameter VSwellCnt is reset.
usiResetMaxTotActive	USINT	If this input is set to 1, the value of the parameter MaxTotActPow and the associated time stamp MaxTotActPowTimestamp are reset.
usiResetMaxTotReactive	USINT	If this input is set to 1, the value of the parameter MaxTotReactPow and the associated time stamp MaxTotReactPowTimestamp are reset.
usiResetMinTruePow	USINT	If this input is set to 1, the true value of the parameter MinTruePowFact is reset to 1 and the associated time stamp MinTruePowFactTimestamp is reset.
usiResetMaxTruePow	USINT	If this input is set to 1, the true value of the parameter MaxTruePowFact is reset to 1 and the associated time stamp MaxTruePowFactTimestamp is reset.

Struct SystemWrite (Continued)

Element	Data Type	Description
usiResetTotReactEng	USINT	If this input is set to 1, the value of the parameter TotReactEnergy is reset.
usiResetTotActEng	USINT	If this input is set to 1, the value of the parameter TotActEnergy is reset.

Struct AvatarWrite

Element	Data Type	Description
usiSetActEng	USINT	Command to set the Total Active Energy value to Total Active Energy Preset value.
usiSetReactEng	USINT	Command to set the Total Reactive Energy value to Total Reactive Energy Preset value.
udiTotActEngValue	UDINT	Preset the Total Active Energy value.
udiTotReactEngValue	UDINT	Preset the Total Reactive Energy value.

UDT_WRREC

The *WRREC* UDT is used to store status information of an acyclic write request internally.

Elements

Element	Data Type	Description
xQ_RTrig	BOOL	Rising edge result, to transfer data to function block variables.
qxDone	BOOL	Acyclic write request result is DONE.
qxBusy	BOOL	Acyclic write function is busy.
qxError	BOOL	Acyclic write request ends with ERROR.
qdwStatus	DWORD	Status result of acyclic write function.
ixStartWrite	BOOL	Enable/Start acyclic write function.

Function Block Types

What’s in This Chapter

There are three basic types of function blocks:

- System Avatar function blocks
- Avatar function blocks
- Device Asset Management function blocks

The following sections give an overview of the functionalities that each of these function block types provides.

The function blocks in the TeSys™ island function block library for the TIA Portal potentially use both cyclic and acyclic communication with the island. Details on the control mechanisms and parameters involved with the two communication types are given in Data Access, page 67.

System Avatar Function Blocks

The System avatar represents the whole island as a system. The System avatar allows the setting of the network configuration, and it computes island-level data. The function blocks of this category provide access to the various functionalities of the System avatar as well as its Diagnostic, Energy, and Asset Management Data. The function blocks of this category are contained in the folders “DP_System” (PROFIBUS DP) and “PN_System” (PROFINET IO) of the library respectively.

System avatar function blocks use a combination of cyclic input and output data as well as acyclic read and write requests, depending on the functionality of the individual function block.

Avatar Function Blocks

The Avatar function blocks are sub-divided into two categories:

- Avatar Control function blocks
- Generic Avatar function blocks

Avatar Control Function Blocks

For each avatar type (Device, Load, or Application), the TeSys™ island function block library for the TIA Portal provides a dedicated control function block in the folder “DP_Avatar” or “PN_Avatar”. These function blocks can be used to control the functionalities of the respective avatars and retrieve status information about the avatar. Each instance of an avatar on a TeSys™ island also requires its own instance of the appropriate avatar control function block in the PLC program, if it will be controlled from the PLC program.

Avatar Control function blocks use a combination of cyclic input and output data as well as acyclic read requests, depending on the functionality of the individual function block and the data the avatar provides.

Generic Avatar Function Blocks

Some data sets and functionalities are common to all avatars. For these, generic function blocks have been defined to access the data. They work with any avatar that supports the data sets and functionality (see individual function block descriptions for details). The function blocks can be found in the folders “DP_ Common” and “PN_Common”. For each avatar that the functionality is used for, or that the data is retrieved for, an individual instance of the function blocks must be created using the same data access parameters as used for the avatar control function blocks. The access to acyclic data should be interlocked externally to avoid detected errors, due to resource limitations of the PLC as described in Acyclic Data, page 69. For cyclic communication, the protocol is designed so that there is no interference between different function blocks.

Generic avatar function blocks use a combination of cyclic input and output data as well as acyclic read requests, depending on the functionality they provide.

Device Asset Management Function Blocks

Asset Management Data supplements the data of the System avatar and the individual Device, Load, and Application avatars. Asset Management Data of the island devices can be accessed through the Device Asset Management function blocks. For each type of device, an individual function block has been created. Starters and SIL[9] Starters share a common implementation, as do Digital and Analog I/O modules. These function blocks can be found in the folders “DP_ Devices” and “PN_Devices”. The function blocks must be instantiated for each device that the Asset Management data is retrieved for.

Device Asset Management function blocks only use acyclic read requests.

> **NOTE:** The devices of the TeSys™ island are represented in the slots starting at number 101 on the TeSys™ island bus device for both PROFINET IO and PROFIBUS DP.

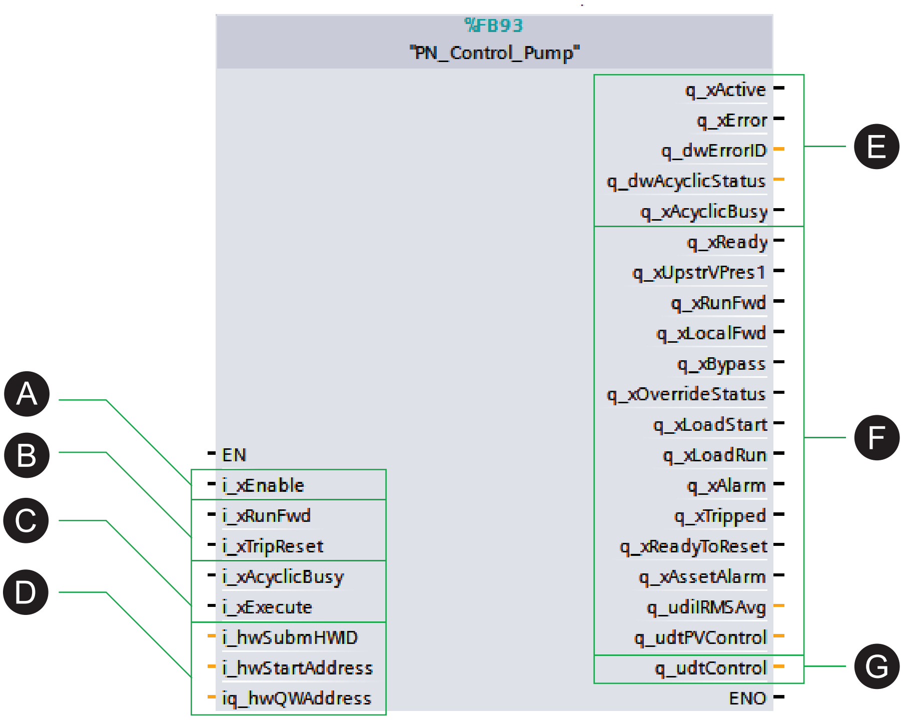
[9] Safety Integrity Level according to standard IEC 61508.

Data Access

What's in This Chapter

As previously stated, the function blocks use both cyclic and acyclic communication mechanisms to retrieve data from and send data to the TeSys™ island, depending on their functionality.

The following sections give an overview of how these two communication mechanisms can be controlled at the function block interface, what status information about the communication is available, and what parameters are involved to configure data access.



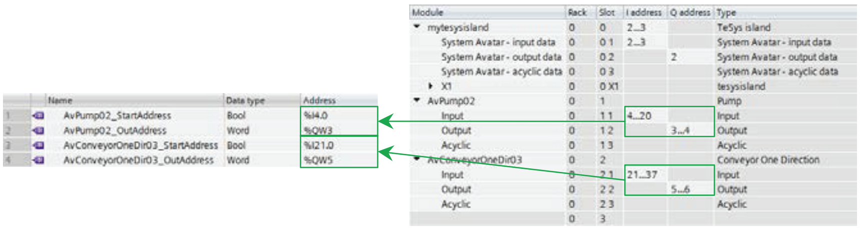
A	Variables to control cyclic communication	E	Function block status
B	Variables to control the Avatar communication	F	Avatar status
C	Variables to control acyclic communication	G	Results of acyclic communication
D	Parameters for cyclic and acyclic communication		

Cyclic Data

All function blocks accessing cyclic data use two interface variables to parametrize the access to the cyclic data:

- i_hwStartAddress for cyclic input data
- iq_hwQWAddress for cyclic output data

Interface variable "i_hwStartAddress" (data type "Variant") is used by the function blocks to retrieve the full amount of cyclic input data associated with the avatar. For the function block to do that, it must be parametrized with the starting address of the input range of the avatar (input sub-module of the avatar for PROFINET IO, avatar module for PROFIBUS DP). The usual method is to create PLC Tags for an address at the beginning of the range. The data size of the PLC tag is of no consequence, as long as it is at the beginning of the range (for example, BOOL “AvPump02_StartAddress” pointing to Address “%I4.0” for the Pump avatar in the example below).

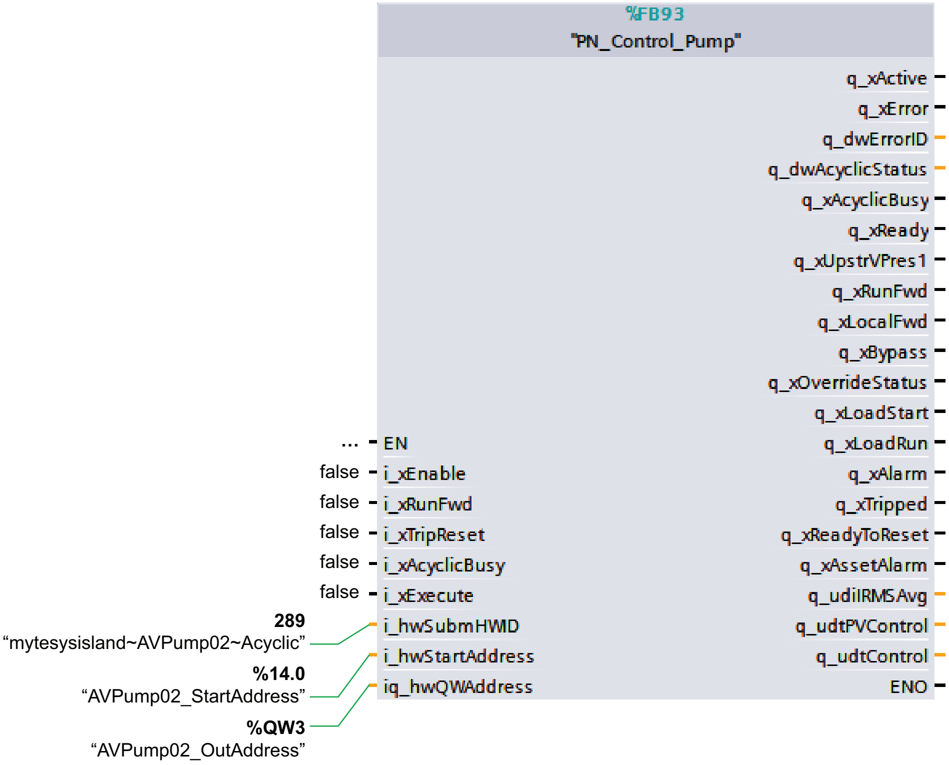


	Name	Data type	Address
1	AvPump02_StartAddress	Bool	%I4.0
2	AvPump02_OutAddress	Word	%QW3
3	AvConveyorOneDir03_StartAddress	Bool	%I21.0
4	AvConveyorOneDir03_OutAddress	Word	%QW5

Module	Rack	Slot	I address	Q address	Type
mytesysisland	0	0	2...3		TeSys island
System Avatar - input data	0	0 1	2...3		System Avatar - input data
System Avatar - output data	0	0 2		2	System Avatar - output data
System Avatar - acyclic data	0	0 3			System Avatar - acyclic data
X1	0	0 X1			tesysisland
AvPump02	0	1			Pump
Input	0	1 1	4...20		Input
Output	0	1 2		3...4	Output
Acyclic	0	1 3			Acyclic
AvConveyorOneDir03	0	2			Conveyor One Direction
Input	0	2 1	21...37		Input
Output	0	2 2		5...6	Output
Acyclic	0	2 3			Acyclic
	0	3			

For the output data direction, the function blocks use the in/out variable “iq_ hwQWAddress” (in some cases also “iq_hwQBAddress” because there is only a single Byte of output data for the avatar). In contrast to the input data, this variable refers to the complete output data of the island (in most cases one Word) and is connected as an in/out variable to the function block—as in some cases, multiple function blocks access the output data of the same avatar in different areas (for example, Avatar Control function block in conjunction with an Avatar Diagnostic function block) and therefore the current status needs to be read before a change is written to the output. Usually access to the output area of the avatar should also be configured using a PLC tag for the complete output area (in the example above, “AvConveyorOneDir03_OutAddress” of type Word referencing address “% QW5”).

NOTE: The function blocks in the TeSys™ island library for the TIA Portal cannot detect any misconfiguration of input/output areas of the avatars. Therefore, the user is responsible for ensuring that the correct input and output areas are configured and used together with the appropriate function block types. Neglecting to do so might result in unexpected behavior of the TeSys™ island.



As soon as the input variable “i_xEnable” is set to “true” in the program, function blocks involving cyclic communication will execute and update the cyclic data. This means that any input variables at the function block interface controlling the avatar through cyclic communication (for example, “i_xRunFwd” in the example above) will be written to the output data area of the avatar. At the same time, any relevant status information from the input data area of the avatar are read by the function block and copied to the output interface (for example, “q_xRunFwd”).

The status output “q_xActive” of the function blocks is the counterpart of “i_xEnable”. If the execution of the function block is enabled and no detected error occurred, “q_xActive” is “true”. If a detected error occurs during the execution of the function block, this is signaled by a rising edge on “q_xError”. The retrieval of cyclic input data and execution of the function block is halted. Additionally, “q_dwErrorID” provides details about what kind of detected error occurred using one of the error code values from the data block ConstTeSysIsland (see Data Block ConstTeSysIsland, page 73). At the same time, all outputs of the function block which are results of either cyclic or acyclic communication with the TeSys island are reset to their respective default values. This means the following:

- All results of numeric (INT, UINT, …) of bitfield (WORD, DWORD, …) type are set to 0.
- All results of string type are set to an empty string.
- All results of Boolean type are set to FALSE.
- All results of the date and time type (DTL in this library) are reset to the minimal value of that type (DTL#1970-01-01-00:00:00.0).
- All results grouped into a user defined type (UDT) or struct are reset to the default values specified in the type definition. These in general also follow the rules above.

NOTE: Changes to the definitions of the UDTs contained in the library also result in changes in the default values used by the function blocks when an error occurs, and therefore **must be avoided**.

⚠WARNING

UNINTENDED EQUIPMENT OPERATION

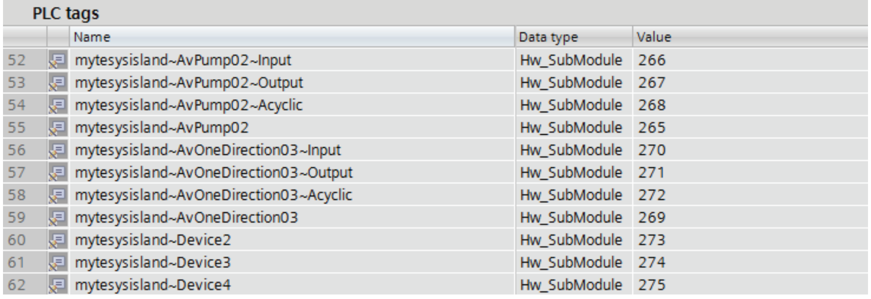
- To avoid interruptions in the operation of the load represented by the avatar when an error occurs, the cyclic output data sent to the island (for example, the Run commands) are left untouched. If a shutdown of the load is required when an error occurs, **you are responsible** for removing any active input bits of the function block as required due to the situation.

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

The function block only leaves the detected error state after a new rising edge on “i_xEnable” is detected, and if the condition that led the function block to enter the detected error state has been cleared.

Acyclic Data

For those function blocks that use acyclic communication with TeSys™ island, a reference to a hardware module containing the acyclic data is required for the function block. Usually, this reference is provided at the function block interface as an input of the type “HW_IO” called “i_hwSubmHWID”. When a device is created in the Siemens TIA Portal (for example, via import of an AML file), system constants with the values for the hardware ID are created automatically for all devices, modules, and sub-modules. These can be directly used to parametrize the access to the acyclic data of an avatar or a device. The names of these constants are usually created using the following pattern: <device name>~<module name>~<submodule name> (for example, “mytesysisland~AvPump02~Acyclic”).



PLC tags

	Name	Data type	Value
52	mytesysisland~AvPump02~Input	Hw_SubModule	266
53	mytesysisland~AvPump02~Output	Hw_SubModule	267
54	mytesysisland~AvPump02~Acyclic	Hw_SubModule	268
55	mytesysisland~AvPump02	Hw_SubModule	265
56	mytesysisland~AvOneDirection03~Input	Hw_SubModule	270
57	mytesysisland~AvOneDirection03~Output	Hw_SubModule	271
58	mytesysisland~AvOneDirection03~Acyclic	Hw_SubModule	272
59	mytesysisland~AvOneDirection03	Hw_SubModule	269
60	mytesysisland~Device2	Hw_SubModule	273
61	mytesysisland~Device3	Hw_SubModule	274
62	mytesysisland~Device4	Hw_SubModule	275

For access to the Asset Management data of the devices of the TeSys™ island, these too are represented as modules in the TeSys™ island. The device modules are placed in the slots starting at slot number 101 for both PROFINET IO and PROFIBUS DP.

NOTE:

- For PROFIBUS DP no sub-modules exist within a module. Therefore, all cyclic and acyclic data belong to the module itself instead of the individual sub-modules. This means that for PROFIBUS DP function blocks that involve cyclic as well as acyclic communication, the hardware ID can be directly derived from the input start address, and no additional hardware ID needs to be parametrized. However, for those function blocks that only use acyclic communication (mainly the Device Asset Management function blocks), the hardware ID of the module must be provided to the function block interface (interface variable “i_hwModuleHWID”).
- The function blocks cannot detect whether there is a match between the module that the hardware ID provided at the function block interface references, and the type of avatar or device that the function block is made for. Therefore, it remains your responsibility to ensure that the correct hardware IDs are used with the appropriate function blocks. Neglecting to do so might result in unexpected behavior of the TeSys™ island.

The reading or writing of the acyclic data has to be triggered using a rising edge at the function block input “i_xExecute”. However, the acyclic reading or writing is only started if the function block is enabled (“i_xEnable”), no detected error has occurred, and the input variable “i_xAcyclicBusy” is “false”. Variable “i_ xAcyclicBusy” can be used to interlock multiple instances of function blocks using acyclic communication (either from the TeSys™ island library for the TIA Portal or any other part of the PLC program using the functions RDREC or WRREC) to avoid detected errors due to the limited resources for acyclic communication of the Siemens PLCs. Each function block involving acyclic communication also provides an output “q_xAcyclicBusy” that is set to “true” while an acyclic read or write request is being executed by the function block and can be used to interlock with other function blocks.

NOTE: You are responsible for ensuring that sufficient resources are available for acyclic communication when starting an acyclic request with one of the function blocks in the library. For details on the limitations of the different PLC types, refer to the Third-Party documentation by Siemens.

If a detected error occurs during acyclic communication, the function block indicates this occurrence with a rising edge on the output “q_xError”, combined with an appropriate detected error code in “q_dwErrorID” from the data block ConstTeSysIsland. All outputs based on communication results are reset to their default values (the same reaction as for all other detected errors—see also Cyclic Data, page 67; for details on the values from ConstTeSysIsland, see Data Block ConstTeSysIsland, page 73). In addition to the detected error ID, the function block in some cases also provides a status code for the acyclic communication in the variable “q_dwAcyclicStatus”. The table below provides an overview of the acyclic status codes in use.

Acyclic status codes

Acyclic status code	Description
Cdw_StsNoMsg	No status details available (also see Data Block ConstTeSysIsland, page 73)
All others	TIA Portal specific status/error code of the used system function • RDREC/WRREC if “q_dwErrorID” is one of the following values from ConstTeSysIsland: ◦ Cdw_ErrReadAcycData1 ◦ Cdw_ErrReadAcycData2 ◦ Cdw_ErrWriteAcycData • LOG2GEO (if “q_dwErrorID” is equal to Cdw_ErrInvalidHwid from ConstTeSysIsland) For further information about the TIA specific detected error codes, see the Siemens TIA Portal information System and .

If the function block finishes the acyclic communication successfully, “q_ xAcyclicBusy” returns to false, “q_dwErrorID” shows the appropriate status code (see Data Block ConstTeSysIsland, page 73) and the data read from the island acyclically is copied to the output variables. The data remains present on the outputs until it is overwritten after a new acyclic request has been completed, the function block is disabled, or an error is detected and the output data is overwritten with default values.

Data Blocks

What’s in This Part

The function block library currently only contains the definition of a single global data block. The following section defines the content of this data block and its intended usage.

Data Block ConstTeSysIsland

The data block ConstTeSysIsland serves as a global storage of constants relevant across multiple function blocks. Currently it only contains the definition of the common set of status and error codes that are used by all function blocks. See Function Blocks, page 74 for a description of each of the function blocks contained in this library and details about which error and status codes are used by the function block.

NOTE: As the function blocks directly reference this global data block, it must always be present within a PLC project using the function blocks described in this document.

To avoid unintended behavior of the function blocks, this data block is protected against changes to the values contained within it. To review the contents of the data block, double-click on it. If prompted for a password, choose Cancel. As protected function blocks within a library are pre-compiled for a type of Siemens® PLC, the function block library contains one Master copy for each PLC type that the library is designed for. All Master copies of the DataBlock (DB) contain the same members and values.

Content of Data Block ConstTeSysIsland

Member	Data Type	Value	Description
Cdw_ErrNoMsg	DWORD	16#0000	Error code: No error message available
Cdw_ErrInternal	DWORD	16#1101	Error code: Internal function block error detected
Cdw_ErrInvalidCycInAddr	DWORD	16#1201	Error code: Invalid start address for cyclic input data configured
Cdw_ErrInvalidHwid	DWORD	16#1202	Error code: Invalid HWID for acyclic communication configured (Details in q_dwAcyclicStatus)
Cdw_ErrInvalidNumOfAvatars	DWORD	16#1203	Error code: Invalid number of avatars specified (i_uiNoConfAvatar must be less or equal 20)
Cdw_ErrReadCycIn	DWORD	16#1301	Error code: Error reading cyclic input data
Cdw_ErrReadAcycData1	DWORD	16#1401	Error code: Error reading acyclic data set 1 (Details in q_dwAcyclicStatus)
Cdw_ErrReadAcycData2	DWORD	16#1402	Error code: Error reading acyclic data set 2 (Details in q_dwAcyclicStatus)
Cdw_ErrWriteAcycData	DWORD	16#1481	Error code: Error writing acyclic data set (Details in q_dwAcyclicStatus)
Cdw_StsNoMsg	DWORD	16#0000	Status code: No status details available
Cdw_StsReadAcycData1Compl	DWORD	16#2001	Status code: Reading acyclic data set 1 completed successfully
Cdw_StsReadAcycData2Compl	DWORD	16#2002	Status code: Reading acyclic data set 2 completed successfully
Cdw_StsWriteAcycDataCompl	DWORD	16#2081	Status code: Writing acyclic data set completed successfully

Function Blocks

What’s in This Part

The following sections define the interfaces of each of the function blocks of the library, their functionality, and their intended use. Besides names, data types, and a description, for each interface variable of the function blocks, one of the following categories is specified:

Categories

Category	Meaning
Function block control	Input variable to control the execution of the function block (for example, trigger acyclic communication).
Function block status	Output variable that gives information on the status of the function block execution.
Communication parameter	Input or in/out variable used to parametrize either cyclic or acyclic data access.
Cyclic data	Input or output variable that is part of the data transferred through cyclic communication to or from TeSys™ island.
Acyclic data	Input or output variable that is part of the data transferred through acyclic communication to or from TeSys™ island.

Additionally, the sections document which values (e.g. which common error and status codes) from ConsTeSysisland (see Data Block ConstTeSysIsland, page 73) are used by each of the function blocks.

System Function Blocks

What's in This Chapter

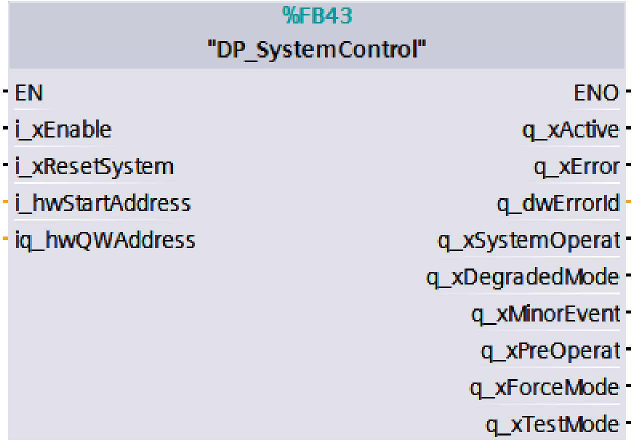
System Control

Function block profile

Function block type	System avatar function block
Type of data access	Cyclic (read/write)
Intended use	System avatar (Bus coupler)
Functionality	The System Control function block returns the status of the acyclic control data of the System avatar.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_SystemControl (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xResetSystem	BOOL	If this input is set to TRUE, the system is reset.	Cyclic data
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication	Communication parameter

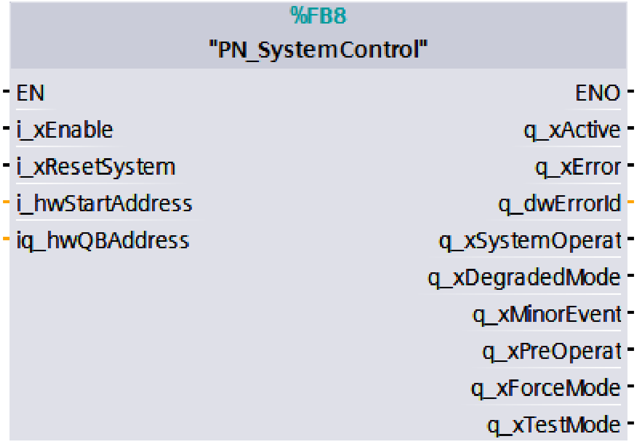
Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_xSystemOperat	BOOL	If this output is set to TRUE, the System avatar is in Operational mode. For detailed information on machine states, refer to the *TeSys island – System, Installation, and Operation Guide* document.	Cyclic data
q_xDegradedMode	BOOL	If this output is set to TRUE, the System avatar is in Degraded mode.	Cyclic data
q_xMinorEvent	BOOL	If this output is set to TRUE, the System avatar is in detected Minor Event mode.	Cyclic data
q_xPreOperat	BOOL	If this output is set to TRUE, the System avatar is in Preoperational mode.	Cyclic data
q_xForceMode	BOOL	If this output is set to TRUE, the System avatar is in Force mode.	Cyclic data
q_xTestMode	BOOL	If this output is set to TRUE, the System avatar is in Test mode.	Cyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	No
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	No
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	No
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

PN_SystemControl (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xResetSystem	BOOL	If this input is set to TRUE, the system is reset.	Cyclic data
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQBAddress	Byte	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_xSystemOperat	BOOL	If this output is set to TRUE, the System avatar is in Operational mode. For detailed information on machine states, refer to the *TeSys island – System, Installation, and Operation Guide* document.	Cyclic data
q_xDegradedMode	BOOL	If this output is set to TRUE, the System avatar is in Degraded mode.	Cyclic data
q_xMinorEvent	BOOL	If this output is set to TRUE, the System avatar is in detected Minor Event mode.	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_xPreOperat	BOOL	If this output is set to TRUE, the System avatar is in Preoperational mode.	Cyclic data
q_xForceMode	BOOL	If this output is set to TRUE, the System avatar is in Force mode.	Cyclic data
q_xTestMode	BOOL	If this output is set to TRUE, the System avatar is in Test mode.	Cyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	No
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	No
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	No
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

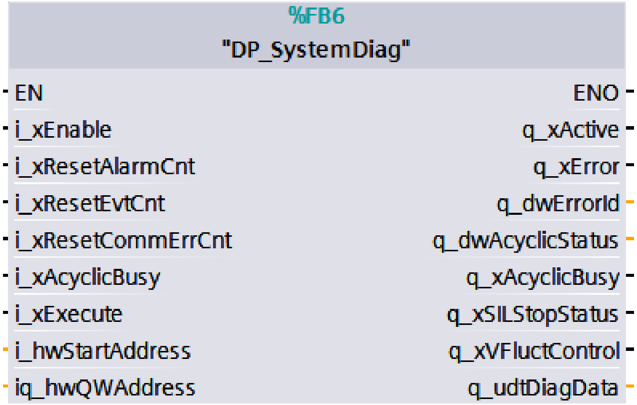
System Diagnostic

Function block profile

Function block type	System avatar function block
Type of data access	Cyclic (read/write) and acyclic (read)
Intended use	System avatar (Bus coupler)
Functionality	The System Diagnostic function block returns the status of the acyclic diagnostic data of the System avatar.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_SystemDiag (PROFIBUS DP)



%FB6
"DP_SystemDiag"

EN ENO
i_xEnable q_xActive
i_xResetAlarmCnt q_xError
i_xResetEvtCnt q_dwErrorId
i_xResetCommErrCnt q_dwAcyclicStatus
i_xAcyclicBusy q_xAcyclicBusy
i_xExecute q_xSILStopStatus
i_hwStartAddress q_xVFluctControl
iq_hwQWAddress q_udtDiagData

The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/ written to the hardware IO interface.	Function block control
i_xResetAlarmCnt	BOOL	If this input is set to TRUE, the counter of detected alarms for the system is set to 0.	Cyclic data
i_xResetEvtCnt	BOOL	If this input is set to TRUE, the counter of detected minor events for the system is set to 0.	Cyclic data
i_xResetCommErrCnt	BOOL	If this input is set to TRUE, the counter of detected errors for the fieldbus communication is set to 0.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related Avatar to be able to execute cyclic read communication	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related Avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar FB is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, an error was detected while executing the Avatar FB. For detailed information about the error see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the error that	Function block status

Outputs (Continued)

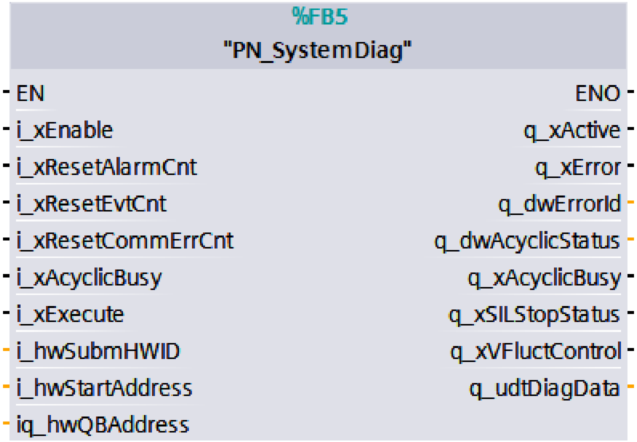
Output	Data Type	Description	Variable Category
		occurred. Possible values are shown in the table below.	
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xSILStopStatus	BOOL	Status of SIL[10] Group Stop function. If this output is set to FALSE, no SIL group has received a SIL Group Stop command.	Cyclic data
q_xVFluctControl	BOOL	If this output is set to TRUE, a control voltage fluctuation is detected.	Cyclic data
q_udtDiagData	UDT_SystemDiag	This is a structure of acyclic system diagnostic data which includes information about the SIL Group stop status and event counters of the system. For detailed information about the structure of this UDT, see UDT_ SystemDiag, page 60.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

[10] Safety Integrity Level according to standard IEC 61508.

PN_SystemDiag (PROFINET IO)



%FB5

"PN_SystemDiag"

EN	ENO
i_xEnable	q_xActive
i_xResetAlarmCnt	q_xError
i_xResetEvtCnt	q_dwErrorId
i_xResetCommErrCnt	q_dwAcyclicStatus
i_xAcyclicBusy	q_xAcyclicBusy
i_xExecute	q_xSILStopStatus
i_hwSubmHWID	q_xVFluctControl
i_hwStartAddress	q_udtDiagData
iq_hwQBAddress	

The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xResetAlarmCnt	BOOL	If this input is set to TRUE, the counter of detected alarms for the system is set to 0.	Cyclic data
i_xResetEvtCnt	BOOL	If this input is set to TRUE, the counter of detected minor events for the system is set to 0.	Cyclic data
i_xResetCommErrCnt	BOOL	If this input is set to TRUE, the counter of detected errors for the fieldbus communication is set to 0.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQBAddress	Byte	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information	Function block status

Outputs (Continued)

Output	Data Type	Description	Variable Category
		about the detected error, see the value of the Error ID.	
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xSILStopStatus	BOOL	Status of SIL[11] Group Stop function. If this output is set to FALSE, no SIL group has received a SIL Group Stop command.	Cyclic data
q_xVFluctControl	BOOL	If this output is set to TRUE, a control voltage fluctuation is detected.	Cyclic data
q_udtDiagData	BOOL	This is a structure of acyclic system diagnostic data which includes information about the SIL Group stop status and detected event counters of the system. For detailed information about the structure of this UDT, see UDT_SystemDiag, page 60.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

System Energy Management

Function block profile

Function block type	System avatar function block
Type of data access	Cyclic (read/write) and acyclic (read)

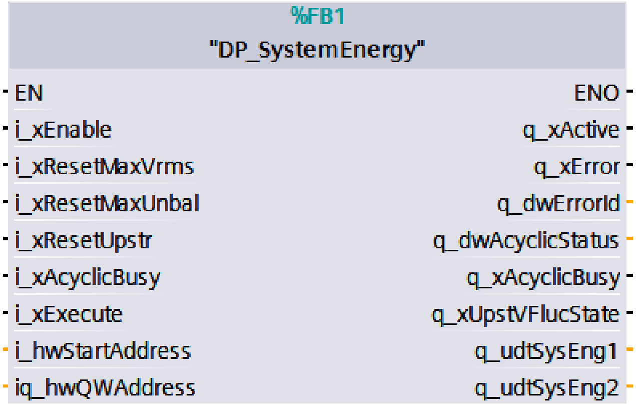
[11] Safety Integrity Level according to standard IEC 61508.

Function block profile (Continued)

Intended use	System avatar (Bus coupler)
Functionality	The System Energy function block returns the status of the acyclic energy data of the System avatar.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_SystemEnergy (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xResetMaxVRMS	BOOL	If this input is set to TRUE, the value of the parameter ResetMaxVRMS is reset.	Cyclic data
i_xResetMaxUnbal	BOOL	If this input is set to TRUE, the value of the parameter ResetMaxUnbalanceVltg is reset.	Cyclic data
i_xResetUpstr	BOOL	If this input is set to TRUE, the voltage fluctuation status is reset.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

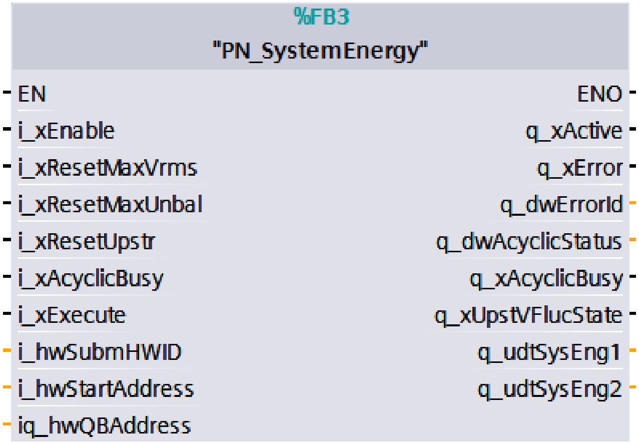
Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xUpstrVFluctState	BOOL	If this output is set to TRUE, a voltage dip or swell is detected.	Cyclic data
q_udtSysEng1	UDT_ SystemEnergy1	This is a structure of acyclic system energy data. See UDT_SystemEnergy1, page 61.	Acyclic data
q_udtSysEng2	UDT_ SystemEnergy2	This is a structure of acyclic system energy data. See UDT_SystemEnergy2, page 62.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	Yes
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	Yes
Cdw_StsWriteAcycDataCompl	No

PN_SystemEnergy (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xResetMaxVRMS	BOOL	If this input is set to TRUE, the value of the parameter ResetMaxVRMS is reset.	Cyclic data
i_xResetMaxUnbal	BOOL	If this input is set to TRUE, the value of the parameter ResetMaxUnbalanceVltg is reset.	Cyclic data
i_xResetUpstr	BOOL	If this input is set to TRUE, the voltage fluctuation status is reset.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQBAddress	Byte	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected	Function block status

Outputs (Continued)

Output	Data Type	Description	Variable Category
		error details. For detailed information about the status values, see Function Block Library, page 49.	
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xUpstrVFluctState	BOOL	If this output is set to TRUE, a voltage dip or swell is detected.	Cyclic data
q_udtSysEng1	UDT_ SystemEnergy1	This is a structure of acyclic system energy data. See UDT_SystemEnergy1, page 61.	Acyclic data
q_udtSysEng2	UDT_ SystemEnergy2	This is a structure of acyclic system energy data. See UDT_SystemEnergy2, page 62.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	Yes
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	Yes
Cdw_StsWriteAcycDataCompl	No

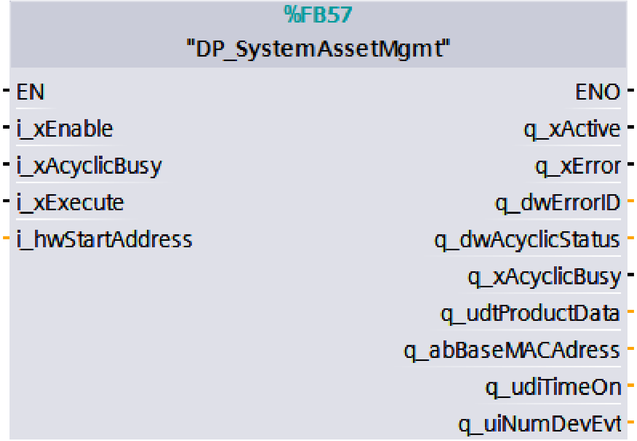
System Asset Management

Function block profile

Function block type	System avatar function block
Type of data access	Acyclic (read)
Intended use	System avatar (Bus coupler)
Functionality	The System Asset Management function block returns the status of the acyclic asset management data of the System avatar.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_SystemAssetMgmt (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the input address range of the related avatar to be able to execute acyclic read communication	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_udtProductData	UDT_ProductData	Product data of the bus coupler.	Acyclic data
q_abBaseMACAddress	Array[0..5] of Byte	The MAC address of the fieldbus Ethernet port 1.	Acyclic data

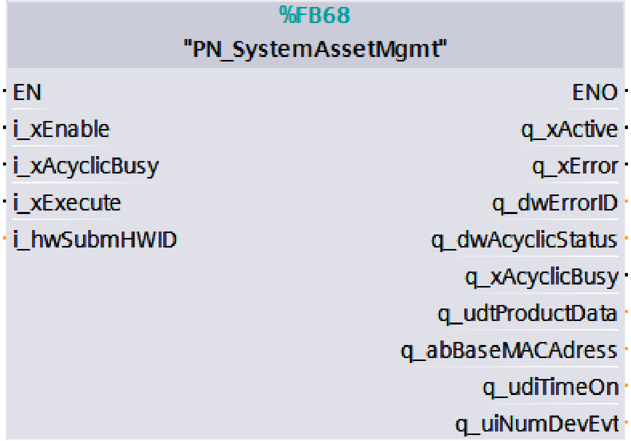
Outputs (Continued)

Output	Data Type	Description	Variable Category
q_udiTimeOn	UDINT	The length of time that the system is switched on.	Acyclic data
q_uiNumDevEvt	UINT	The number of detected System minor events.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	No
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

PN_SystemAssetMgmt (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_udtProductData	UDT_ ProductData	Product data of the bus coupler.	Acyclic data
q_abBaseMACAddress	Array[0..5] of Byte	The MAC address of the fieldbus Ethernet port 1.	Acyclic data
q_udiTimeOn	UDINT	The length of time that the system is switched on.	Acyclic data
q_uiNumDevEvt	UINT	The number of detected System minor events.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	No
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	No
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No

Status and error codes from ConstTeSysIsland (Continued)

Member name	Used by the function block
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

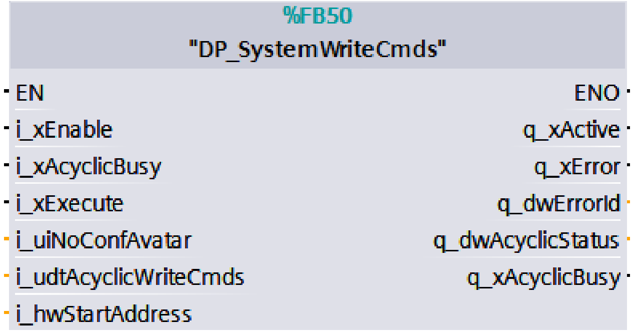
System Write Commands

Function block profile

Function block type	System avatar function block
Type of data access	Acyclic (write)
Intended use	System avatar (Bus coupler)
Functionality	The System Write Commands function block writes the acyclic reset commands and energy values for the complete system at once.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_SystemWriteCmds (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/ written to the hardware IO interface.	Function block control
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_uiNoConfAvatar	UINT	Used to parameterize the number of configured avatars to define the length of the data, which needs to be written. Minimum value is 0 (only Bus coupler) and maximum value is 20.	Function block control
i_udtAcyclicWriteCmds	UDT_SystemWriteCmds	This is a structure of acyclic system write data, which includes reset commands and energy data of the system and all avatars. See UDT_ SystemWriteCmds, page 63.	Acyclic data
i_hwStartAddress	Variant	Must correspond to the first address of the I address of the related avatar to be able to execute the acyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status

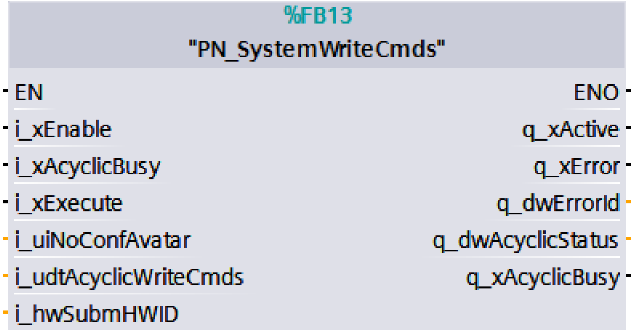
Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	Yes
Cdw_ErrReadCycIn	No
Cdw_ErrReadAcycData1	No

Status and error codes from ConstTeSysIsland (Continued)

Member name	Used by the function block
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	Yes
Cdw_StsReadAcycData1Compl	No
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	Yes

PN_SystemWriteCmds (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_uiNoConfAvatar	UINT	Used to parameterize the number of configured avatars to define the length of the data, which needs to be written. Minimum value is 0 (only Bus coupler) and maximum value is 20.	Function block control
i_udtAcyclicWriteCmds	UDT_SystemWriteCmds	This is a structure of acyclic system write data, which includes reset commands and energy data of the system and all avatars. For detailed information about the structure of this UDT, see UDT_SystemWriteCmds, page 63.	Acyclic data
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information	Function block status

Outputs (Continued)

Output	Data Type	Description	Variable Category
		about the detected error, see the value of the Error ID.	
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	No
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	Yes
Cdw_ErrReadCycIn	No
Cdw_ErrReadAcycData1	No
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	Yes
Cdw_StsReadAcycData1Compl	No
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	Yes

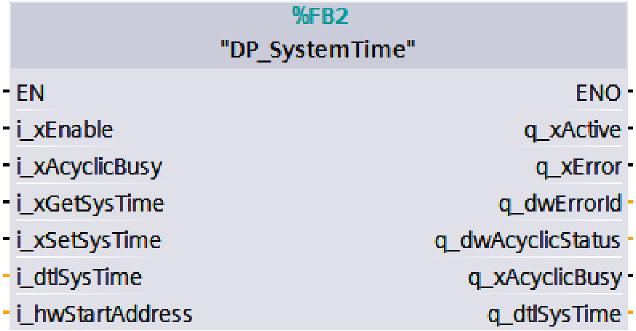
System Time

Function block profile

Function block type	System avatar function block.
Type of data access	Acyclic (read/write).
Intended use	System avatar (Bus coupler).
Functionality	The System Time function block reads and writes the system time of the Bus coupler. Write command has higher priority.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_SystemTime (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xGetSysTime	BOOL	If this input is set to TRUE, the system time of the Bus coupler is read to the PLC.	Function block control
i_xSetSysTime	BOOL	If this input is set to TRUE, the time is written from PLC to the Bus coupler.	Function block control
i_dtlSysTime	DTL	Corresponds to PLC time to be written to the Bus coupler.	Acyclic data
i_hwStartAddress	Variant	Must correspond to the first address of the I address of the related avatar to be able to execute the acyclic communication.	Communication parameter

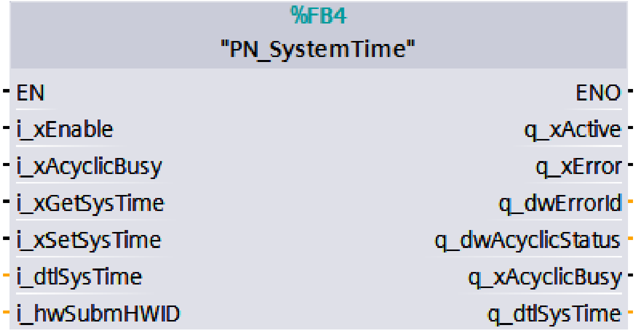
Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_dtlSysTime	DTL	Corresponds to the system time read from the Bus coupler.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	No
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	Yes
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

PN_SystemTime (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xGetSysTime	BOOL	If this input is set to TRUE, the system time of the Bus coupler is read to the engineering system.	Function block control
i_xSetSysTime	BOOL	If this input is set to TRUE, the engineering system time is written to the Bus coupler.	Function block control
i_dtlSysTime	DTL	Corresponds to engineering system time written to the Bus coupler.	Acyclic data
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_dtlSysTime	DTL	Corresponds to the system time read from the Bus coupler.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	No
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	No
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	Yes
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	Yes

Avatar Control Function Blocks

What’s in This Chapter

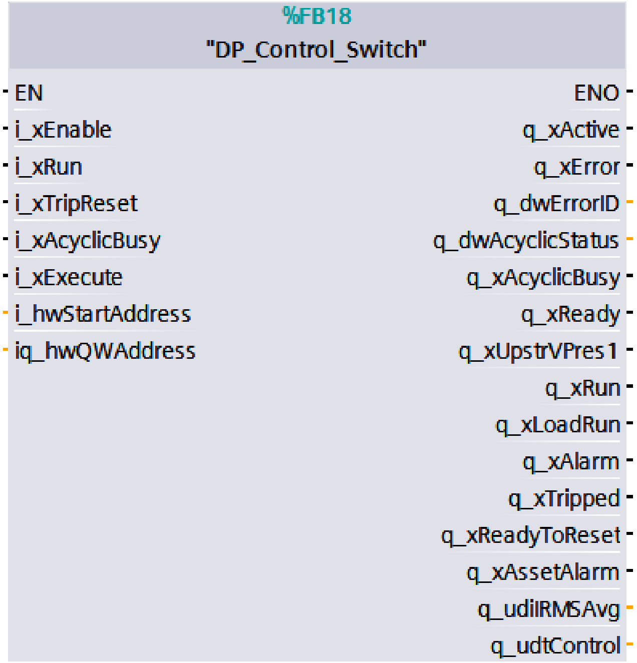
Switch

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Device avatar “Switch”.
Functionality	The Switch function block establishes or interrupts a power line in an electric circuit.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Switch (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRun	BOOL	If this input is set to TRUE, the avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information	Function block status

Outputs (Continued)

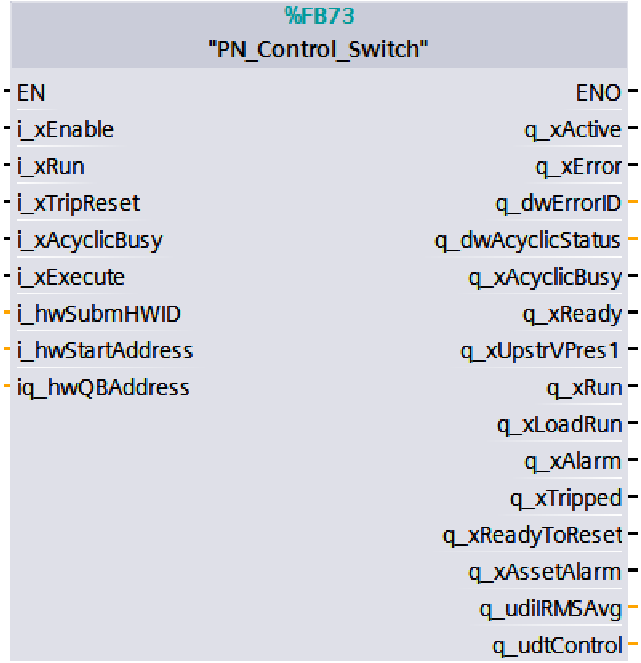
Output	Data Type	Description	Variable Category
		about the detected error, see the value of the Error ID.	
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xRun	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[12] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_ Control, page 102.	Acyclic data

[12] Safety Integrity Level according to standard IEC 61508.

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

PN_Control_Switch (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRun	BOOL	If this input is set to TRUE, the avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQBAddress	Byte	Must correspond to the address of the cyclic output submodule of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xRun	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[13] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 102.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

[13] Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	No
SILGroup	n/A	No
ThermalCapacity	n/A	No
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	No
	PhaseUnbalance	Yes
	UnderCurrent	No
AlarmMsg2	OverCurrent	No
	MotorOverheat	No
TripMsg1	GroundCurrent	Yes
	ThermalOverload	No
	LongStart	No
	Jam	No
	PhaseUnbalance	Yes
	UnderCurrent	No
	Stall	No
TripMsg2	PhConfig	Yes
	OverCurrent	No
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	No
TimeToTrip	n/A	No
TimeToReset	n/A	No
PAStatusReg1	PA0Status	No
	PA1Status	No
	PA2Status	No
	PA3Status	No
	PA4Status	No
	PA5Status	No
	PA6Status	No
	PA7Status	No
	PA8Status	No
	PA9Status	No

Switch – SIL Stop, W. Cat. 1/2

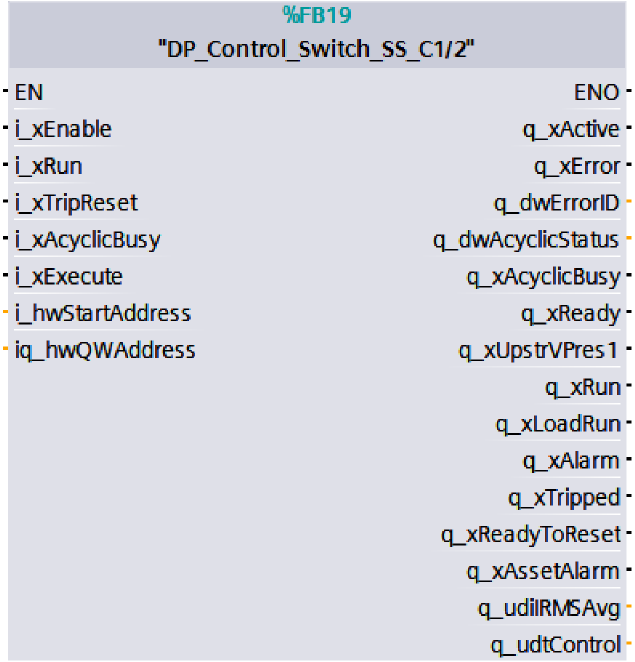
NOTE: Safety Integrity Level according to standard IEC 61508. Wiring Category 1 and Category 2 according to ISO 13849.

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Device avatar “Switch – SIL Stop, W. Cat. 1/2”
Functionality	The Switch SIL Stop Cat function block establishes or interrupts a power line in an electric circuit with Stop Category 0 or Stop Category 1[14], Wiring Category 1 and Category 2.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Switch_SS_C1/2 (PROFIBUS DP)



[14] Stop categories according to EN/IEC 60204-1.

The following tables give information about the parameters of the function block interface.

Input

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRun	BOOL	If this input is set to TRUE, the avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Output

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xRun	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data

Output (Continued)

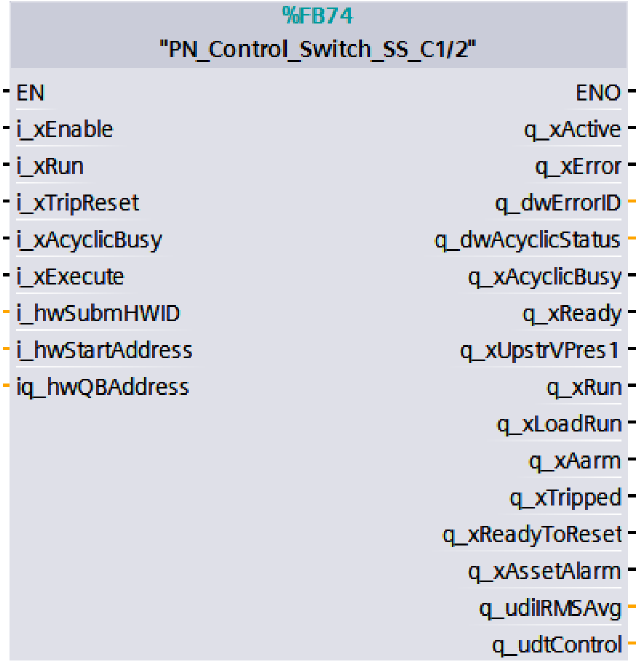
Output	Data Type	Description	Variable Category
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[15] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 109.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

[15] Safety Integrity Level according to standard IEC 61508.

PN_Control_Switch_SS_C1/2 (PROFINET IO)



%FB74

"PN_Control_Switch_SS_C1/2"

EN	ENO
i_xEnable	q_xActive
i_xRun	q_xError
i_xTripReset	q_dwErrorID
i_xAcyclicBusy	q_dwAcyclicStatus
i_xExecute	q_xAcyclicBusy
i_hwSubmHWID	q_xReady
i_hwStartAddress	q_xUpstrVPres1
iq_hwQBAddress	q_xRun
	q_xLoadRun
	q_xAarm
	q_xTripped
	q_xReadyToReset
	q_xAssetAlarm
	q_udiIRMSAvg
	q_udtControl

The following tables give information about the parameters of the function block interface.

Input

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRun	BOOL	If this input is set to TRUE, the avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQBAddress	Byte	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Output

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xRun	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[16] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 109.	Acyclic data

[16] Safety Integrity Level according to standard IEC 61508.

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	No
SILGroup	n/A	Yes
ThermalCapacity	n/A	No
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	No
	PhaseUnbalance	Yes
	UnderCurrent	No
AlarmMsg2	OverCurrent	No
	MotorOverheat	No
TripMsg1	GroundCurrent	Yes
	ThermalOverload	No
	LongStart	No
	Jam	No
	PhaseUnbalance	Yes
	UnderCurrent	No
	Stall	No
TripMsg2	PhConfig	Yes
	OverCurrent	No
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	No

Elements (Continued)

Element	Sub-Element	Supported
TimeToTrip	n/A	No
TimeToReset	n/A	No
PAStatusReg1	PA0Status	No
	PA1Status	No
	PA2Status	No
	PA3Status	No
	PA4Status	No
	PA5Status	No
	PA6Status	No
	PA7Status	No
	PA8Status	No
	PA9Status	No

Switch – SIL Stop, W. Cat. 3/4

NOTE: Safety Integrity Level according to standard IEC 61508. Wiring Category 3 and Category 4 according to ISO 13849.

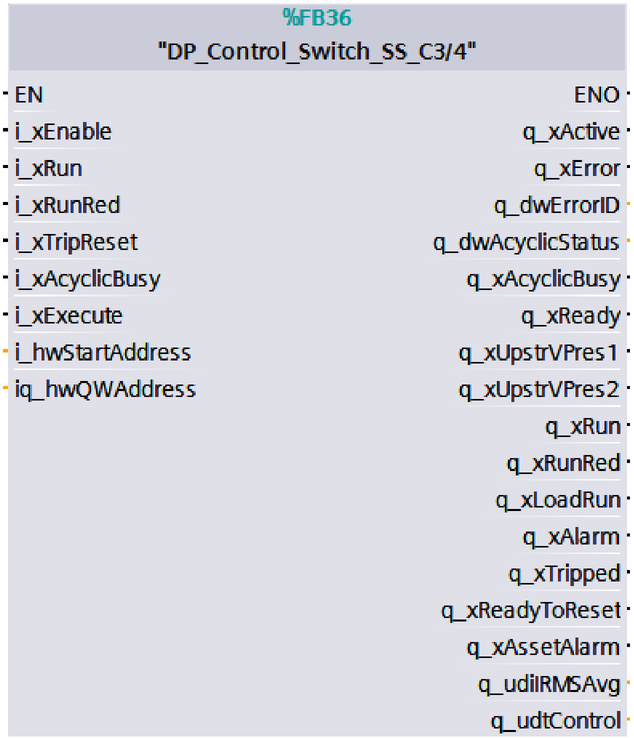
Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Device avatar “Switch – SIL Stop, W. Cat. 3/4”
Functionality	The Switch SIL Stop Cat function block establishes or interrupts a power line in an electric circuit with Stop Category 0 or Stop Category 1[17], Wiring Category 3 and Category 4.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

[17] Stop categories according to EN/IEC 60204-1.

DP_Control_Switch_SS_C3/4 (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRun	BOOL	If this input is set to TRUE, the avatar forward switch is closed.	Cyclic data
i_xRunRed	BOOL	If this input is set to TRUE, the redundant avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

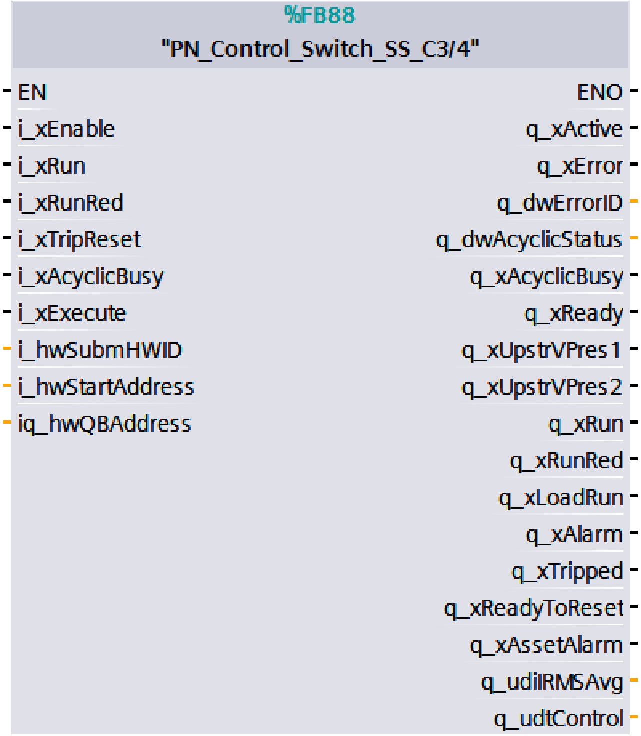
Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Communication parameter
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xRun	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunRed	BOOL	If this output is set to TRUE, the redundant avatar forward switch is closed.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[18] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 115.	Acyclic data

[18] Safety Integrity Level according to standard IEC 61508.

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

PN_Control_Switch_SS_C3/4 (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRun	BOOL	If this input is set to TRUE, the avatar forward switch is closed.	Cyclic data
i_xRunRed	BOOL	If this input is set to TRUE, the redundant avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQBAddress	Byte	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_xRun	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunRed	BOOL	If this output is set to TRUE, the redundant avatar forward switch is closed.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[19] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 115.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which

[19] Safety Integrity Level according to standard IEC 61508.

elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	No
SILGroup	n/A	Yes
ThermalCapacity	n/A	No
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	No
	PhaseUnbalance	Yes
	UnderCurrent	No
AlarmMsg2	OverCurrent	No
	MotorOverheat	No
TripMsg1	GroundCurrent	Yes
	ThermalOverload	No
	LongStart	No
	Jam	No
	PhaseUnbalance	Yes
	UnderCurrent	No
	Stall	No
TripMsg2	PhConfig	Yes
	OverCurrent	No
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	No
TimeToTrip	n/A	No
TimeToReset	n/A	No
PAStatusReg1	PA0Status	No
	PA1Status	No
	PA2Status	No
	PA3Status	No
	PA4Status	No
	PA5Status	No
	PA6Status	No
	PA7Status	No
	PA8Status	No
	PA9Status	No

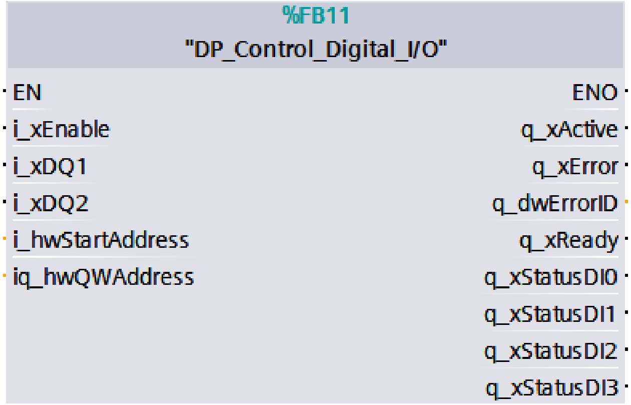
Digital I/O

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write).
Intended use	Device avatar “Digital I/O”.
Functionality	The Digital I/O function block provides information about the Digital I/O avatar with four inputs and two outputs.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Digital_I/O (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xDQ1	BOOL	If this input is set to TRUE, the digital output 0 is set to TRUE.	Cyclic data
i_xDQ2	BOOL	If this input is set to TRUE, the digital output 1 is set to TRUE.	Cyclic data
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

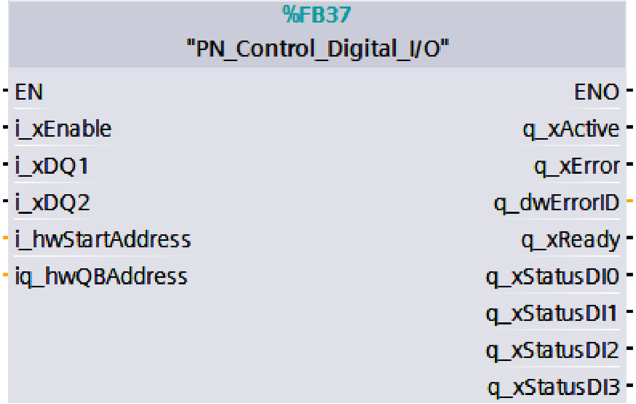
Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xStatusDI0	BOOL	If this output is set to TRUE, the digital input 0 of the Digital I/O avatar is set to TRUE.	Cyclic data
q_xStatusDI1	BOOL	If this output is set to TRUE, the digital input 1 of the Digital I/O avatar is set to TRUE.	Cyclic data
q_xStatusDI2	BOOL	If this output is set to TRUE, the digital input 2 of the Digital I/O avatar is set to TRUE.	Cyclic data
q_xStatusDI3	BOOL	If this output is set to TRUE, the digital input 3 of the Digital I/O avatar is set to TRUE.	Cyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	No
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	No
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	No
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

PN_Control_Digital_I/O (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xDQ1	BOOL	If this input is set to TRUE, the digital output 0 is set to TRUE.	Cyclic data
i_xDQ2	BOOL	If this input is set to TRUE, the digital output 1 is set to TRUE.	Cyclic data
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQBAddress	Byte	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xStatusDI0	BOOL	If this output is set to TRUE, the digital input 0 of the Digital I/O avatar is set to TRUE.	Cyclic data
q_xStatusDI1	BOOL	If this output is set to TRUE, the digital input 1 of the Digital I/O avatar is set to TRUE.	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_xStatusDI2	BOOL	If this output is set to TRUE, the digital input 2 of the Digital I/O avatar is set to TRUE.	Cyclic data
q_xStatusDI3	BOOL	If this output is set to TRUE, the digital input 3 of the Digital I/O avatar is set to TRUE.	Cyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	No
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	No
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	No
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The Digital I/O avatar does not support any of the contained members, therefore no output variable of that type exists.

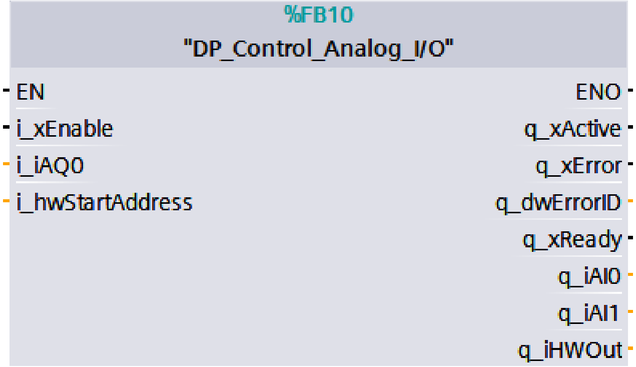
Analog I/O

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write).
Intended use	Device avatar “Analog I/O”.
Functionality	The Analog I/O function block provides information about the Analog I/O avatar with two inputs and one output.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Analog_I/O (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_iAQ0	INT	Value to be written to analog output 0. Unit and scaling depends on the analog output type configured. • Type 0 (Unit: mV) • Type 1 (Unit: mV) • Type 2 (Unit: µA) • Type 3 (Unit: µA)	Cyclic data
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

Outputs

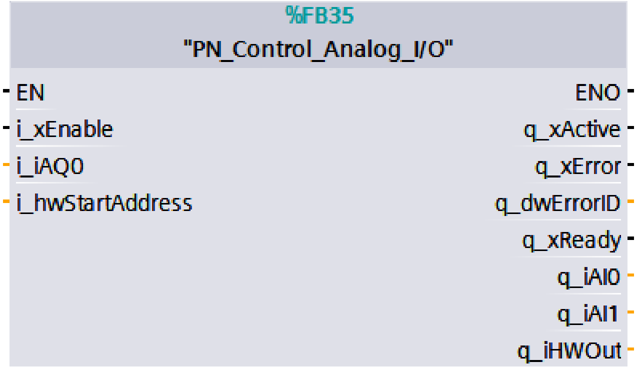
Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_iAI0	INT	Indicates the value read from the analog input 0. Unit and scaling depends on the analog output type configured. • Type 0 to 12 (Unit: 0.1 °C) • Type 13 (Unit: mV) • Type 14 (Unit: mV) • Type 15 (Unit: µA) • Type 16 (Unit: µA)	Cyclic data
q_iAI1	INT	Indicates the value read from the analog input 1.	Cyclic data
q_iHWOut	INT	Hardware address of the analog output.	Communication parameter

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	No
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	No
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	No
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

PN_Control_Analog_I/O (PROFINET IO)

The following tables give information about the parameters of the function block interface.



Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_iAQ0	INT	Value to be written to analog output 0. Unit and scaling depends on the analog output type configured. • Type 0 (Unit: mV) • Type 1 (Unit: mV) • Type 2 (Unit: µA) • Type 3 (Unit: µA)	Cyclic data
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_iAI0	INT	Indicates the value read from the analog input 0. Unit and scaling depends on the analog output type configured. • Type 0 to 12 (Unit: 0.1 °C) • Type 13 (Unit: mV) • Type 14 (Unit: mV) • Type 15 (Unit: µA) • Type 16 (Unit: µA)	Cyclic data
q_iAI1	INT	Indicates the value read from the analog input 1.	Cyclic data
q_iHWOut	INT	Hardware address of the analog output.	Communication parameter

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	No
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	No
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	No
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The Digital I/O avatar does not support any of the contained members, therefore no output variable of that type exists.

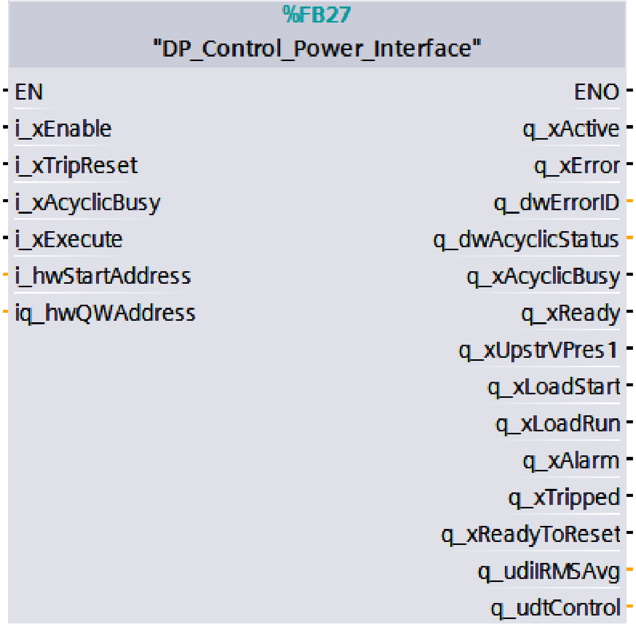
Power Interface – without I/O (measure)

Function block profile

Function block type	Avatar Control function block
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Load avatar “Power Interface – without I/O”.
Functionality	The Power Interface function block is used to monitor current on an external power device, such as a solid-state relay, soft starter, or variable speed drive.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Power_Interface (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

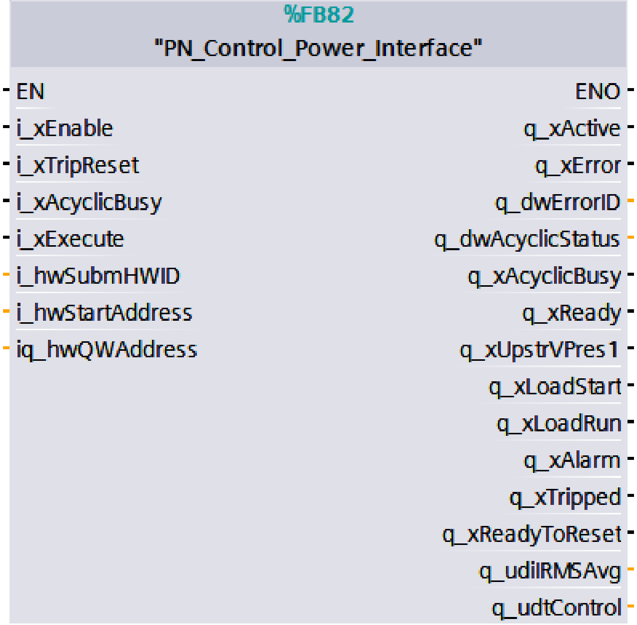
Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 128.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

PN_Control_Power_Interface (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
		to motor running but also for non-motor avatars).	
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A).	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 128.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	Yes
SILGroup	n/A	No

Elements (Continued)

Element	Sub-Element	Supported
ThermalCapacity	n/A	Yes
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	Yes
TripMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	LongStart	Yes
	Jam	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	Yes
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	Yes
TimeToTrip	n/A	Yes
TimeToReset	n/A	Yes
PAStatusReg1	PA0Status	No
	PA1Status	No
	PA2Status	No
	PA3Status	No
	PA4Status	No
	PA5Status	No
	PA6Status	No
	PA7Status	No
	PA8Status	No
	PA9Status	No

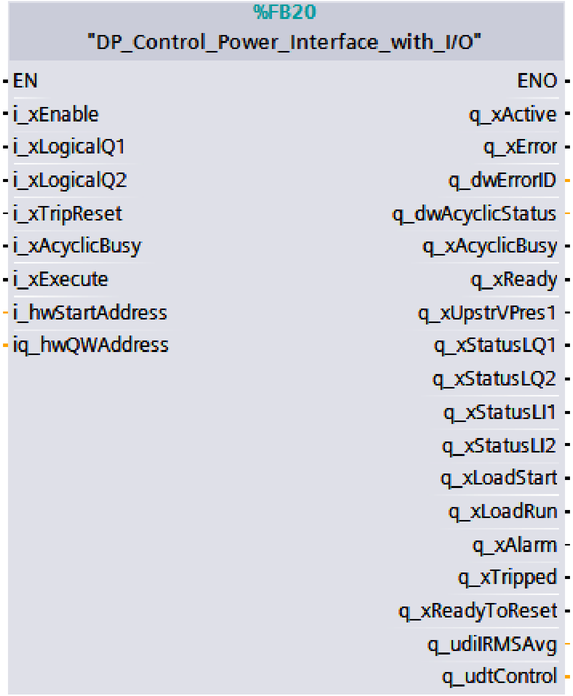
Power Interface – with I/O (control)

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Load Avatar “Power Interface – with I/O”.
Functionality	The Power Interface with I/O function block is used to monitor current and control an external power device, such as a solid-state relay, soft starter, or variable speed drive.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Power_Interface_with_I/O (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xLogicalQ1	BOOL	If this input is set to TRUE, the logical output 1 is set to TRUE.	Cyclic data
i_xLogicalQ2	BOOL	If this input is set to TRUE, the logical output 2 is set to TRUE.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xStatusLQ1	BOOL	If this output is set to TRUE, the logical output 1 is set to TRUE.	Cyclic data
q_xStatusLQ2	BOOL	If this output is set to TRUE, the logical output 2 is set to TRUE.	Cyclic data

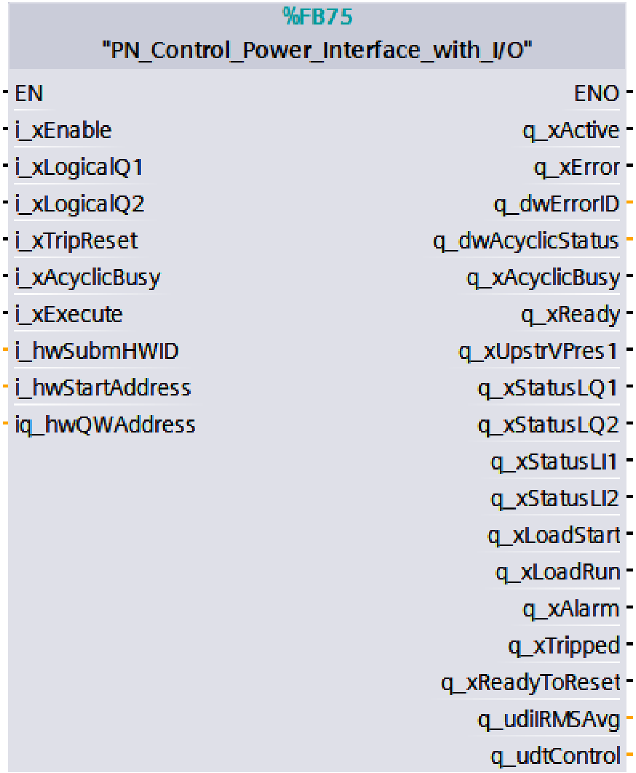
Outputs (Continued)

Output	Data Type	Description	Variable Category
q_xStatusLI1	BOOL	If this output is set to TRUE, the logical input 1 of the avatar is set to TRUE.	Cyclic data
q_xStatusLI2	BOOL	If this output is set to TRUE, the logical input 2 of the avatar is set to TRUE.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 135.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

PN_Control_Power_Interface_with_I/O (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xLogicalQ1	BOOL	If this input is set to TRUE, the logical output 1 is set to TRUE.	Cyclic data
i_xLogicalQ2	BOOL	If this input is set to TRUE, the logical output 2 is set to TRUE.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xStatusLQ1	BOOL	If this output is set to TRUE, the logical output 1 is set to TRUE.	Cyclic data
q_xStatusLQ2	BOOL	If this output is set to TRUE, the logical output 2 is set to TRUE.	Cyclic data
q_xStatusLI1	BOOL	If this output is set to TRUE, the logical input 1 of the avatar is set to TRUE.	Cyclic data
q_xStatusLI2	BOOL	If this output is set to TRUE, the logical input 2 of the avatar is set to TRUE.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 135.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	Yes
SILGroup	n/A	No
ThermalCapacity	n/A	Yes
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	Yes
TripMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	LongStart	Yes
	Jam	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	Yes
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	Yes

Elements (Continued)

Element	Sub-Element	Supported
TimeToTrip	n/A	Yes
TimeToReset	n/A	Yes
PAStatusReg1	PA0Status	No
	PA1Status	No
	PA2Status	No
	PA3Status	No
	PA4Status	No
	PA5Status	No
	PA6Status	No
	PA7Status	No
	PA8Status	No
	PA9Status	No

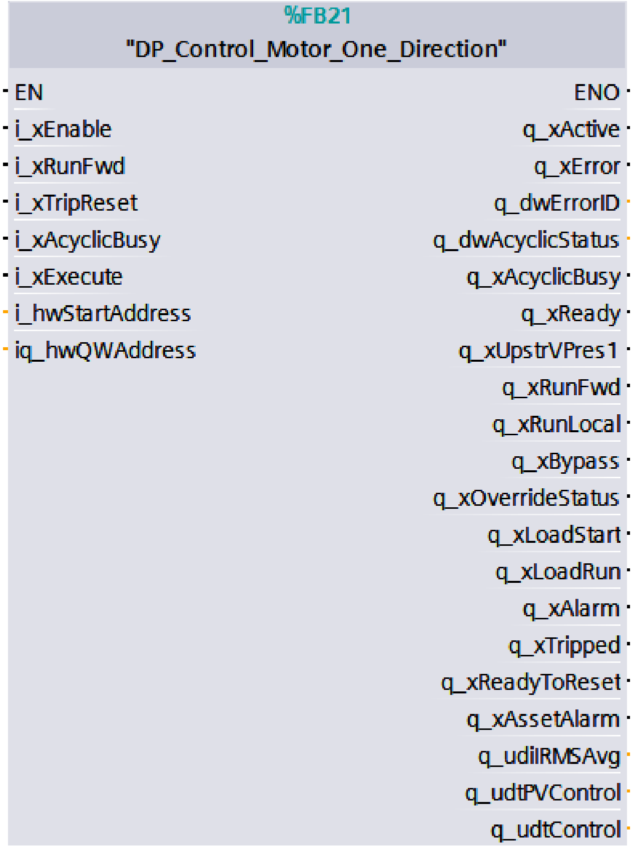
Motor One Direction

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Load avatar “Motor One Direction”.
Functionality	The Motor One Direction function block is used to manage a motor in one direction.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Motor_One_Direction (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwd	BOOL	If this input is TRUE, the avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control

Inputs (Continued)

Input	Data Type	Description	Variable Category
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xRunFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunLocal	BOOL	If this output is set to TRUE, the avatar Local Forward command is set.	Cyclic data
q_xBypass	BOOL	If this output is set to TRUE, the avatar Bypass command is set.	Cyclic data
q_xOverrideStatus	BOOL	If this output is set to TRUE, the avatar Manual mode is set.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data

Outputs (Continued)

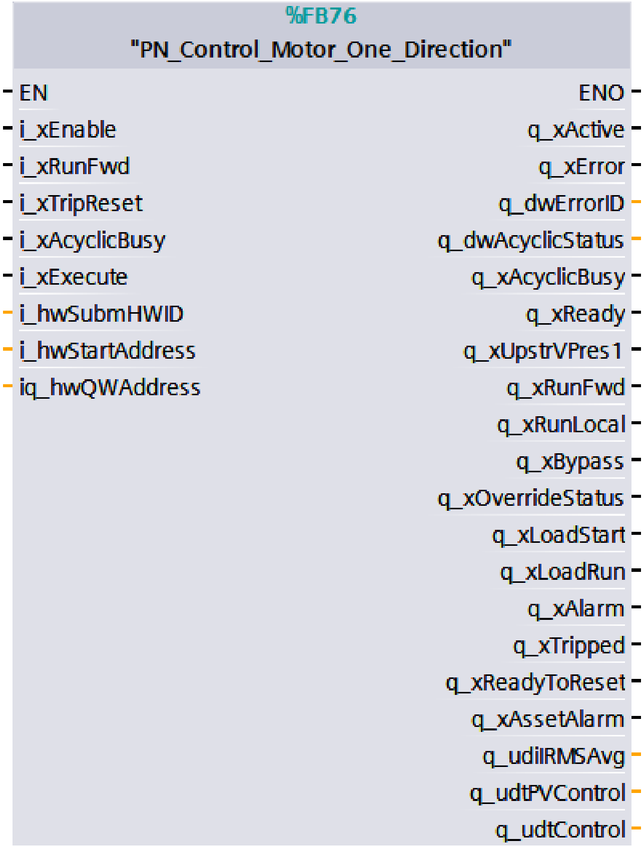
Output	Data Type	Description	Variable Category
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[20] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtPVControl	UDT_ PVControl	This is a structure of cyclic PV status data for Control avatars. For detailed information about the structure of this UDT, see UDT_PVControl, page 58.	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 142.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

[20] Safety Integrity Level according to standard IEC 61508.

PN_Control_Motor_One_Direction (PROFINET IO)



%FB76
"PN_Control_Motor_One_Direction"

EN	ENO
i_xEnable	q_xActive
i_xRunFwd	q_xError
i_xTripReset	q_dwErrorID
i_xAcyclicBusy	q_dwAcyclicStatus
i_xExecute	q_xAcyclicBusy
i_hwSubmHWID	q_xReady
i_hwStartAddress	q_xUpstrVPres1
iq_hwQWAddress	q_xRunFwd
	q_xRunLocal
	q_xBypass
	q_xOverrideStatus
	q_xLoadStart
	q_xLoadRun
	q_xAlarm
	q_xTripped
	q_xReadyToReset
	q_xAssetAlarm
	q_udiIRMSAvg
	q_udtPVControl
	q_udtControl

The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwd	BOOL	If this input is TRUE, the avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xRunFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunLocal	BOOL	If this output is set to TRUE, the avatar Local Forward command is set.	Cyclic data
q_xBypass	BOOL	If this output is set to TRUE, the avatar Bypass command is set.	Cyclic data
q_xOverrideStatus	BOOL	If this output is set to TRUE, the avatar Manual mode is set.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[21] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data

[21] Safety Integrity Level according to standard IEC 61508.

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_udtPVControl	UDT_ PVControl	This is a structure of cyclic PV status data for Control avatars. For detailed information about the structure of this UDT, see UDT_PVControl, page 58.	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 142.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	Yes
SILGroup	n/A	No
ThermalCapacity	n/A	Yes
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	Yes

Elements (Continued)

Element	Sub-Element	Supported
TripMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	LongStart	Yes
	Jam	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	Yes
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	Yes
TimeToTrip	n/A	Yes
TimeToReset	n/A	Yes
PAStatusReg1	PA0Status	Yes
	PA1Status	Yes
	PA2Status	Yes
	PA3Status	Yes
	PA4Status	Yes
	PA5Status	Yes
	PA6Status	Yes
	PA7Status	Yes
	PA8Status	Yes
	PA9Status	Yes

Motor One Direction – SIL Stop, W. Cat. 1/2

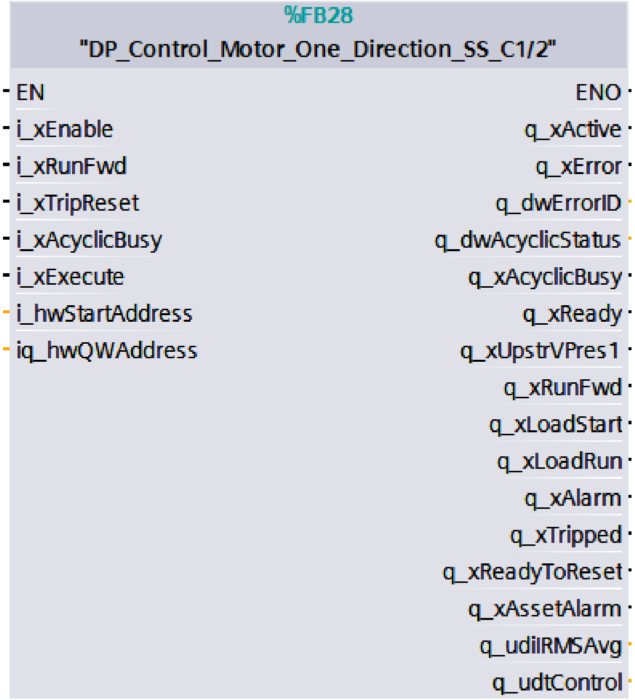
NOTE: Safety Integrity Level according to standard IEC 61508. Wiring Category 1 and Category 2 according to ISO 13849.

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Load avatar “Motor One Direction – SIL Stop, W. Cat. 1/2”.
Functionality	The Motor One Direction SIL function block is used to manage a motor in one direction with Stop Category 0 or Stop Category 1[22], Wiring Category 1 and Category 2.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Motor_One_Direction_SS_C1/2 (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwd	BOOL	If this input is TRUE, the avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data

[22] Stop categories according to EN/IEC 60204-1.

Inputs (Continued)

Input	Data Type	Description	Variable Category
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xRunFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data

Outputs (Continued)

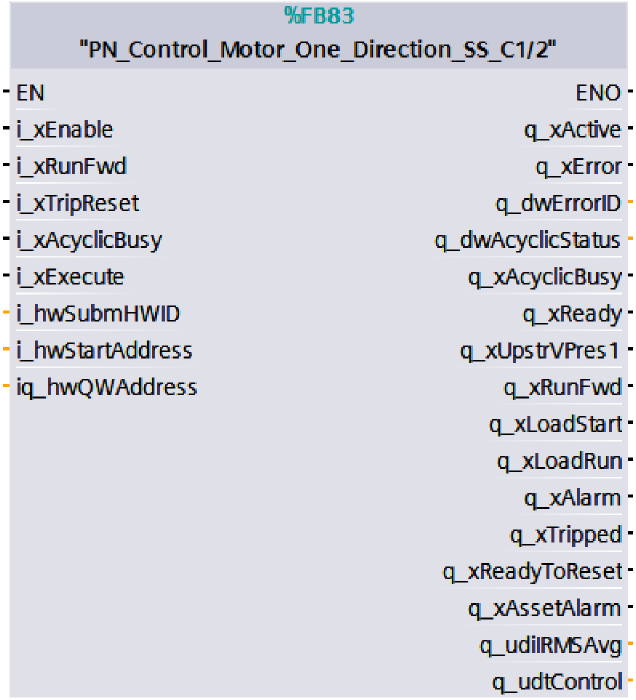
Output	Data Type	Description	Variable Category
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[23] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 149.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

[23] Safety Integrity Level according to standard IEC 61508.

PN_Control_Motor_One_Direction_SS_C1/2 (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwd	BOOL	If this input is TRUE, the avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xRunFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[24] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 149.	Acyclic data

[24] Safety Integrity Level according to standard IEC 61508.

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Element	Sub-Element	Supported
MotorTemperature	n/A	Yes
SILGroup	n/A	Yes
ThermalCapacity	n/A	Yes
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	Yes
TripMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	LongStart	Yes
	Jam	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	Yes
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	Yes
TimeToTrip	n/A	Yes

Element	Sub-Element	Supported
TimeToReset	n/A	Yes
PAStatusReg1	PA0Status	No
	PA1Status	No
	PA2Status	No
	PA3Status	No
	PA4Status	No
	PA5Status	No
	PA6Status	No
	PA7Status	No
	PA8Status	No
	PA9Status	No

Motor One Direction – SIL Stop, W. Cat. 3/4

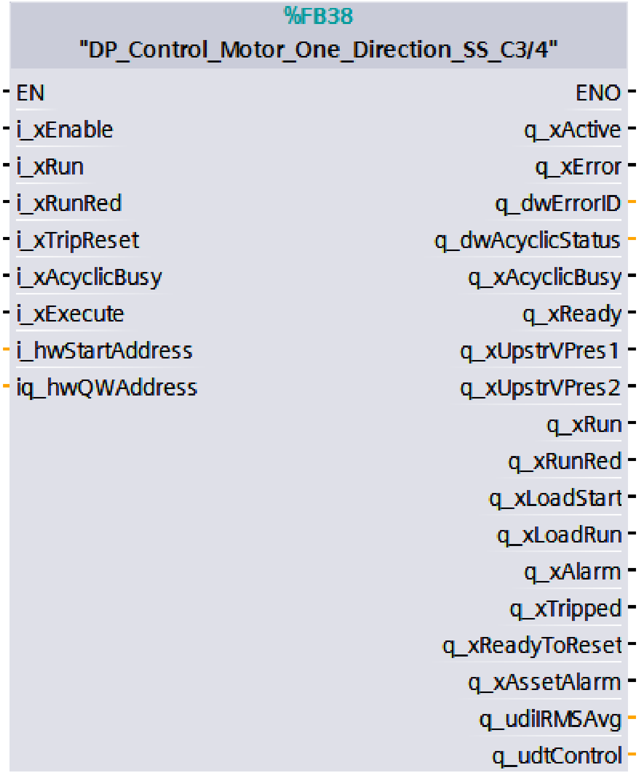
NOTE: Safety Integrity Level according to standard IEC 61508. Wiring Category 3 and Category 4 according to ISO 13849.

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Load avatar “Motor One Direction – SIL Stop, W. Cat. 3/4”
Functionality	The Motor One Direction SIL function block is used to manage a motor in one direction with Stop Category 0 or Stop Category 1[25], Wiring Category 3 and Category 4.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Motor_One_Direction_SS_C3/4 (PROFIBUS DP)



%FB38
"DP_Control_Motor_One_Direction_SS_C3/4"

EN
i_xEnable
i_xRun
i_xRunRed
i_xTripReset
i_xAcyclicBusy
i_xExecute
i_hwStartAddress
iq_hwQWAddress

ENO
q_xActive
q_xError
q_dwErrorID
q_dwAcyclicStatus
q_xAcyclicBusy
q_xReady
q_xUpstrVPres1
q_xUpstrVPres2
q_xRun
q_xRunRed
q_xLoadStart
q_xLoadRun
q_xAlarm
q_xTripped
q_xReadyToReset
q_xAssetAlarm
q_udiIRMSAvg
q_udtControl

The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRun	BOOL	If this input is TRUE, the avatar forward switch is closed.	Cyclic data
i_xRunRed	BOOL	If this input is TRUE, the redundant avatar forward switch is closed.	Cyclic data

[25] Stop categories according to EN/IEC 60204-1.

Inputs (Continued)

Input	Data Type	Description	Variable Category
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xRun	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunRed	BOOL	If this output is set to TRUE, the redundant avatar forward switch is closed.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data

Outputs (Continued)

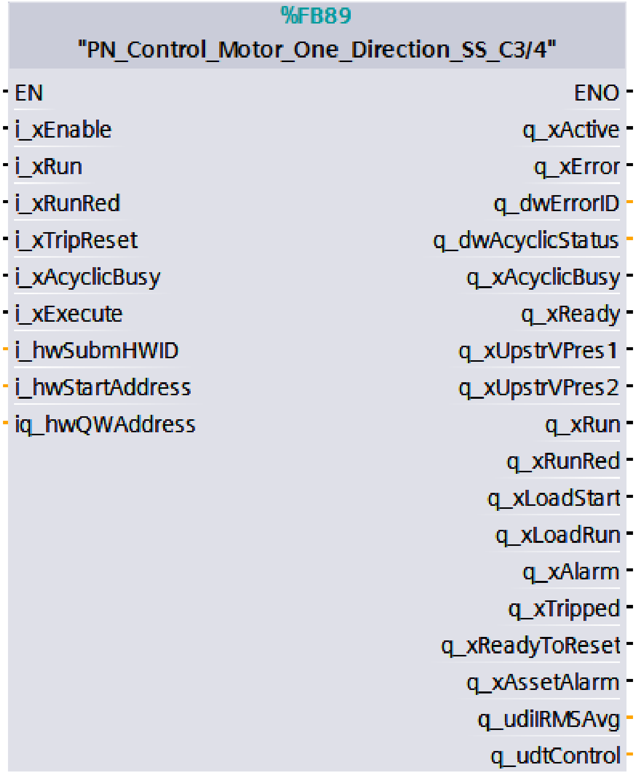
Output	Data Type	Description	Variable Category
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[26] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 156.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

[26] Safety Integrity Level according to standard IEC 61508.

PN_Control_Motor_One_Direction_SS_C3/4 (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRun	BOOL	If this input is TRUE, the avatar forward switch is closed.	Cyclic data
i_xRunRed	BOOL	If this input is TRUE, the redundant avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xRun	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunRed	BOOL	If this output is set to TRUE, the redundant avatar forward switch is closed.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[27] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data

[27] Safety Integrity Level according to standard IEC 61508.

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 156.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	Yes
SILGroup	n/A	Yes
ThermalCapacity	n/A	Yes
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	Yes

Elements (Continued)

Element	Sub-Element	Supported
TripMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	LongStart	Yes
	Jam	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	Yes
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	Yes
TimeToTrip	n/A	Yes
TimeToReset	n/A	Yes
PAStatusReg1	PA0Status	No
	PA1Status	No
	PA2Status	No
	PA3Status	No
	PA4Status	No
	PA5Status	No
	PA6Status	No
	PA7Status	No
	PA8Status	No
	PA9Status	No

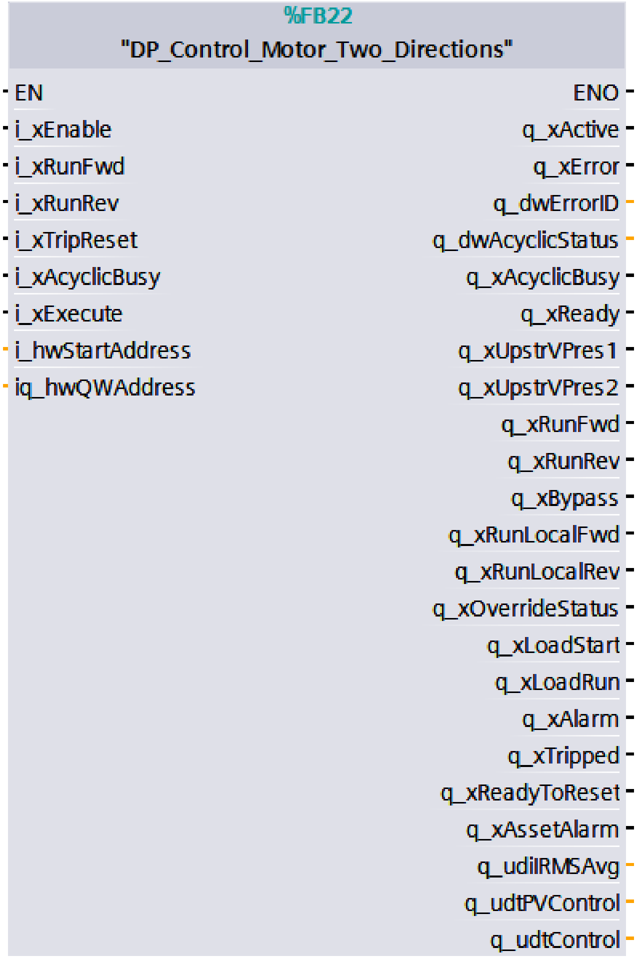
Motor Two Directions

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Load avatar “Motor Two Direction”.
Functionality	The Motor Two Directions function block is used to manage a motor in two directions (forward and reverse).

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Motor_Two_Directions (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwd	BOOL	If this input is TRUE, the avatar forward switch is closed.	Cyclic data
i_xRunRev	BOOL	Cyclic data.	

Inputs (Continued)

Input	Data Type	Description	Variable Category
i_xTripReset	BOOL	Cyclic data.	
i_xAcyclicBusy	BOOL	Function block control.	
i_xExecute	BOOL	Function block control.	
i_hwStartAddress	Variant	Communication parameter.	

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xRunFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunRev	BOOL	If this output is set to TRUE, the avatar reverse switch is closed.	Cyclic data
q_xBypass	BOOL	If this output is set to TRUE, the avatar Bypass command is set.	Cyclic data
q_xRunLocalFwd	BOOL	If this output is set to TRUE, the avatar Local Forward command is set.	Cyclic data
q_xRunLocalRev	BOOL	If this output is set to TRUE, the avatar Local Reverse command is set.	Cyclic data
q_xOverrideStatus	BOOL	If this output is set to TRUE, the avatar Manual mode is set.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data

Outputs (Continued)

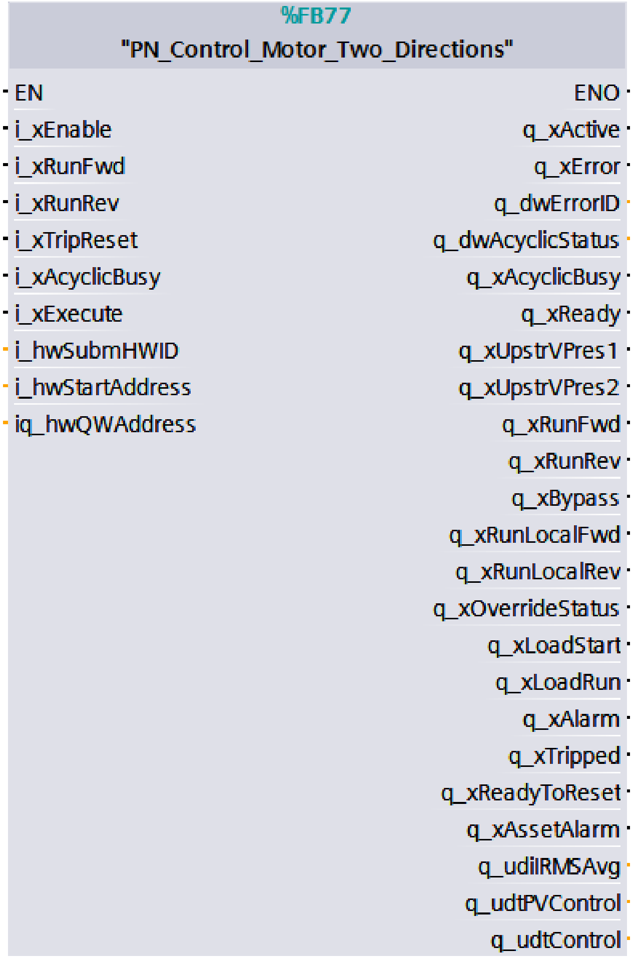
Output	Data Type	Description	Variable Category
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[28] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtPVControl	UDT_ PVControl	This is a structure of cyclic PV status data for Control avatars. For detailed information about the structure of this UDT, see UDT_PVControl, page 58.	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 163.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

[28] Safety Integrity Level according to standard IEC 61508.

PN_Control_Motor_Two_Directions (PROFINET IO)



The following tables provide information about the function block interface parameters.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwd	BOOL	If this input is TRUE, the avatar forward switch is closed.	Cyclic data
i_xRunRev	BOOL	If this input is TRUE, the avatar reverse switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xRunFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunRev	BOOL	If this output is set to TRUE, the avatar reverse switch is closed.	Cyclic data
q_xBypass	BOOL	If this output is set to TRUE, the avatar Bypass command is set.	Cyclic data
q_xRunLocalFwd	BOOL	If this output is set to TRUE, the avatar Local Forward command is set.	Cyclic data
q_xRunLocalRev	BOOL	If this output is set to TRUE, the avatar Local Reverse command is set.	Cyclic data
q_xOverrideStatus	BOOL	If this output is set to TRUE, the avatar Manual mode is set.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[29] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtPVControl	UDT_ PVControl	This is a structure of cyclic PV status data for Control avatars. For detailed information about the structure of this UDT, see UDT_PVControl, page 58.	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 163.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

[29] Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	Yes
SILGroup	n/A	No
ThermalCapacity	n/A	Yes
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	Yes
TripMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	LongStart	Yes
	Jam	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	Yes
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	Yes
TimeToTrip	n/A	Yes
TimeToReset	n/A	Yes
PAStatusReg1	PA0Status	No
	PA1Status	No
	PA2Status	No
	PA3Status	No
	PA4Status	No
	PA5Status	No
	PA6Status	No
	PA7Status	No
	PA8Status	No
	PA9Status	No

Motor Two Directions – SIL Stop, W. Cat. 1/2

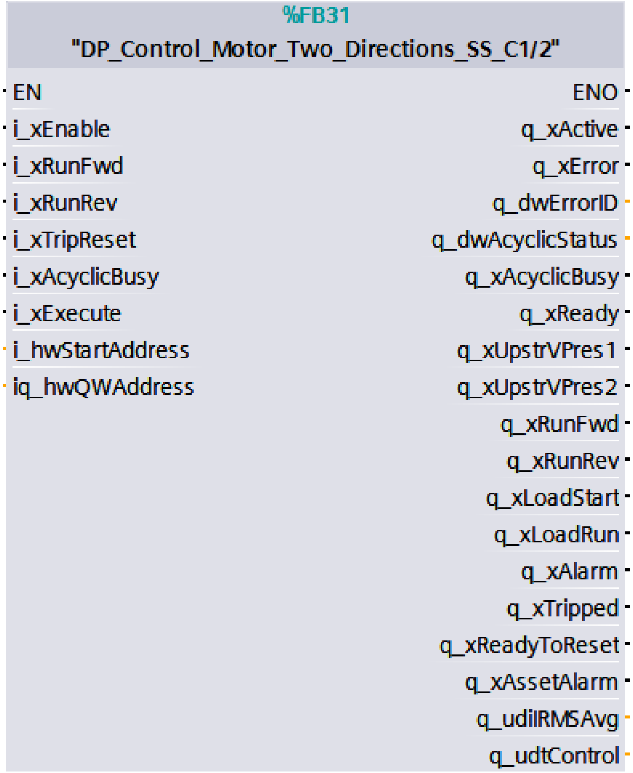
NOTE: Safety Integrity Level according to standard IEC 61508. Wiring Category 1 and Category 2 according to ISO 13849.

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Load avatar “Motor Two Directions - SIL Stop, W. Cat. 1/2".
Functionality	The Motor Two Direction SIL function block is used to manage a motor in two directions (forward and reverse) with Stop Category 0 or Stop Category 1[30], Wiring Category 1 and Category 2.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Motor_Two_Directions_SS_C1/2 (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwd	BOOL	If this input is TRUE, the avatar forward switch is closed.	Cyclic data
i_xRunRev	BOOL	If this input is TRUE, the avatar reverse switch is closed.	Cyclic data

[30] Stop categories according to EN/IEC 60204-1.

Inputs (Continued)

Input	Data Type	Description	Variable Category
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xRunFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunRev	BOOL	If this output is set to TRUE, the avatar reverse switch is closed.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data

Outputs (Continued)

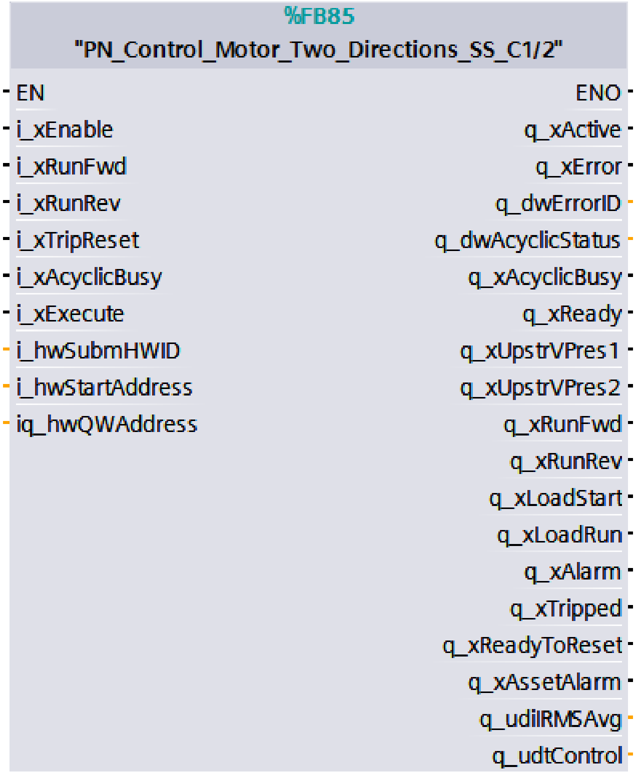
Output	Data Type	Description	Variable Category
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[31] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 170.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

[31] Safety Integrity Level according to standard IEC 61508.

PN_Control_Motor_Two_Directions_SS_C1/2 (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwd	BOOL	If this input is TRUE, the avatar forward switch is closed.	Cyclic data
i_xRunRev	BOOL	If this input is TRUE, the avatar reverse switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xRunFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunRev	BOOL	If this output is set to TRUE, the avatar reverse switch is closed.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[32] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data

[32] Safety Integrity Level according to standard IEC 61508.

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A).	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 170.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	Yes
SILGroup	n/A	Yes
ThermalCapacity	n/A	Yes
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	Yes

Elements (Continued)

Element	Sub-Element	Supported
TripMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	LongStart	Yes
	Jam	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	Yes
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	Yes
TimeToTrip	n/A	Yes
TimeToReset	n/A	Yes
PAStatusReg1	PA0Status	No
	PA1Status	No
	PA2Status	No
	PA3Status	No
	PA4Status	No
	PA5Status	No
	PA6Status	No
	PA7Status	No
	PA8Status	No
	PA9Status	No

Motor Two Directions – SIL Stop, W. Cat. 3/4

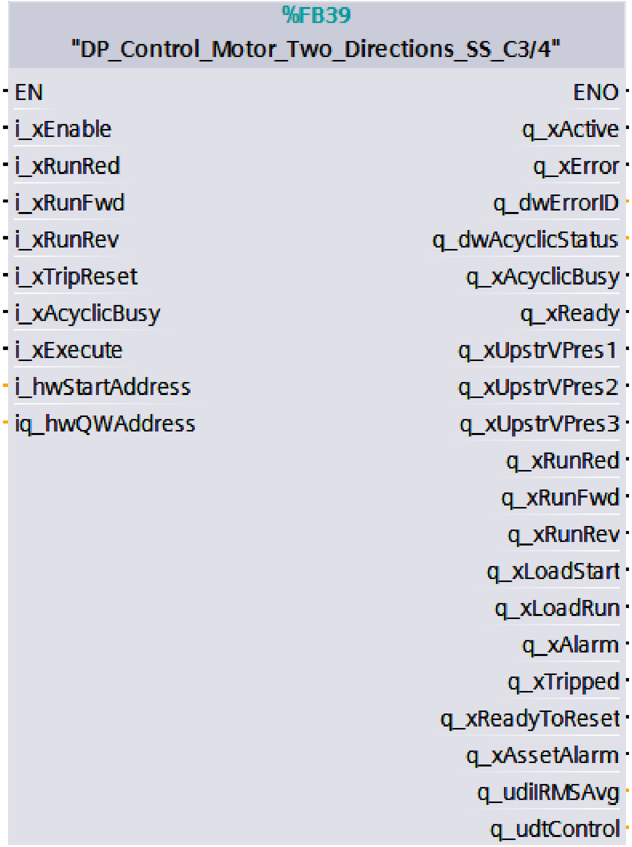
NOTE: Safety Integrity Level according to standard IEC 61508. Wiring Category 3 and Category 4 according to ISO 13849.

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Load avatar “Motor Two Directions - SIL Stop, W. Cat. 3/4".
Functionality	The Motor Two Direction SIL function block is used to manage a motor in two directions (forward and reverse) with Stop Category 0 or Stop Category 1[33], Wiring Category 3 and Category 4.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Motor_Two_Directions_SS_C3/4 (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunRed	BOOL	If this input is TRUE, the redundant avatar switch is closed.	Cyclic data

[33] Stop categories according to EN/IEC 60204-1.

Inputs (Continued)

Input	Data Type	Description	Variable Category
i_xRunFwd	BOOL	If this input is TRUE, the avatar forward switch is closed.	Cyclic data
i_xRunRev	BOOL	If this input is TRUE, the avatar reverse switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xUpstrVPres3	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the third starter/power device in this avatar.	Cyclic data
q_xRunRed	BOOL	If this output is set to TRUE, the redundant avatar switch is closed.	Cyclic data

Outputs (Continued)

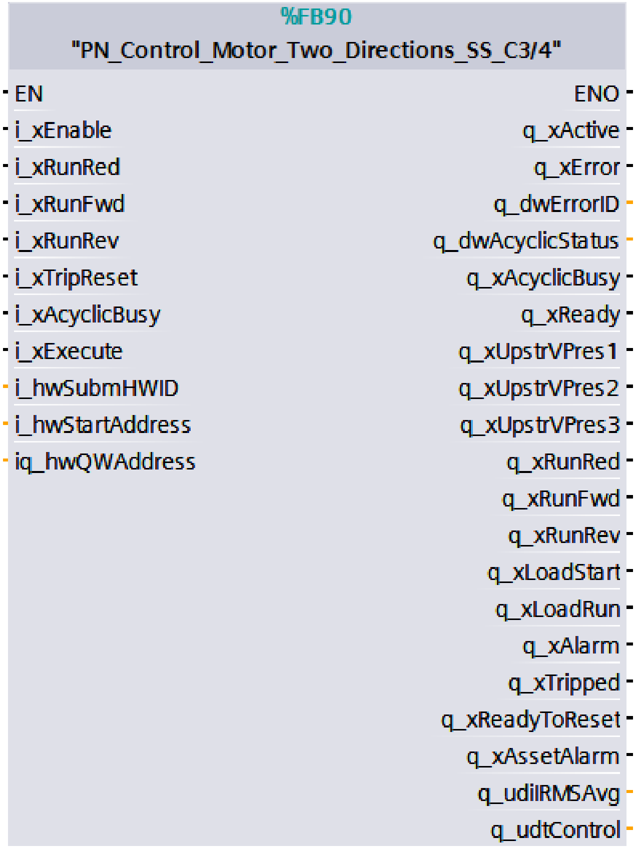
Output	Data Type	Description	Variable Category
q_xRunFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunRev	BOOL	If this output is set to TRUE, the avatar reverse switch is closed.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[34] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 177.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

[34] Safety Integrity Level according to standard IEC 61508.

PN_Control_Motor_Two_Directions_SS_C3/4 (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunRed	BOOL	If this input is TRUE, the redundant avatar switch is closed.	Cyclic data
i_xRunFwd	BOOL	If this input is TRUE, the avatar forward switch is closed.	Cyclic data
i_xRunRev	BOOL	If this input is TRUE, the avatar reverse switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xUpstrVPres3	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the third starter/power device in this avatar.	Cyclic data
q_xRunRed	BOOL	If this output is set to TRUE, the redundant avatar switch is closed.	Cyclic data
q_xRunFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunRev	BOOL	If this output is set to TRUE, the avatar reverse switch is closed.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[35] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 177.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Element	Sub-Element	Supported
MotorTemperature	n/A	Yes
SILGroup	n/A	Yes
ThermalCapacity	n/A	Yes
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes

[35] Safety Integrity Level according to standard IEC 61508.

Element	Sub-Element	Supported
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	Yes
TripMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	LongStart	Yes
	Jam	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	Yes
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	Yes
TimeToTrip	n/A	Yes
TimeToReset	n/A	Yes
PAStatusReg1	PA0Status	No
	PA1Status	No
	PA2Status	No
	PA3Status	No
	PA4Status	No
	PA5Status	No
	PA6Status	No
	PA7Status	No
	PA8Status	No
	PA9Status	No

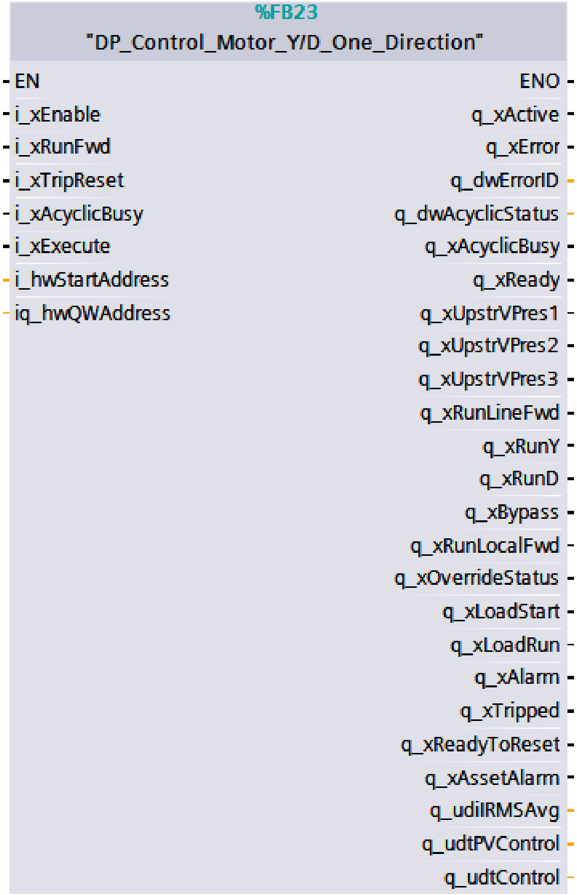
Motor Y/D, One Direction

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Load avatar “Motor Y/D, One Direction“.
Functionality	The Motor YD One Direction function block is used to manage a wye/ delta (star/triangle) motor in one direction.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Motor_Y/D_One_Direction (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwd	BOOL	If this input is TRUE, the avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data

Inputs (Continued)

Input	Data Type	Description	Variable Category
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xUpstrVPres3	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the third starter/power device in this avatar.	Cyclic data
q_xRunLineFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunY	BOOL	If this output is set to TRUE, the Y switch for Motor Y/D avatar is closed.	Cyclic data
q_xRunD	BOOL	If this output is set to TRUE, the D switch for Motor Y/D avatar is closed.	Cyclic data
q_xBypass	BOOL	If this output is set to TRUE, the avatar Bypass command is set.	Cyclic data
q_xRunLocalFwd	BOOL	If this output is set to TRUE, the avatar Local Forward command is set.	Cyclic data

Outputs (Continued)

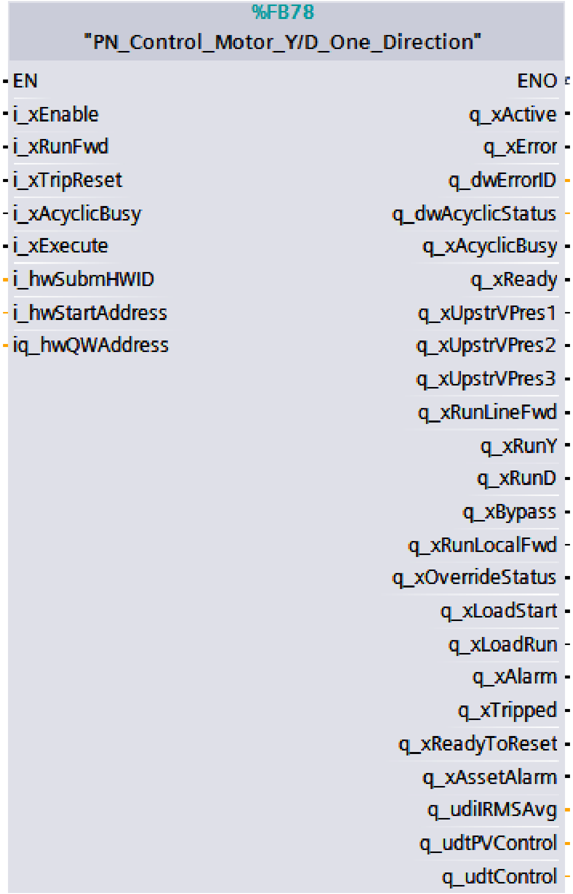
Output	Data Type	Description	Variable Category
q_xOverrideStatus	BOOL	If this output is set to TRUE, the avatar Manual mode is set.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[36] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtPVControl	UDT_ PVControl	This is a structure of cyclic PV status data for Control avatars. For detailed information about the structure of this UDT, see UDT_PVControl, page 58.	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 184.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

[36] Safety Integrity Level according to standard IEC 61508.

PN_Control_Motor_Y/D_One_Direction (PROFINET IO)



%FB78

"PN_Control_Motor_Y/D_One_Direction"

EN	ENO
i_xEnable	q_xActive
i_xRunFwd	q_xError
i_xTripReset	q_dwErrorID
i_xAcyclicBusy	q_dwAcyclicStatus
i_xExecute	q_xAcyclicBusy
i_hwSubmHWID	q_xReady
i_hwStartAddress	q_xUpstrVPres1
iq_hwQWAddress	q_xUpstrVPres2
	q_xUpstrVPres3
	q_xRunLineFwd
	q_xRunY
	q_xRunD
	q_xBypass
	q_xRunLocalFwd
	q_xOverrideStatus
	q_xLoadStart
	q_xLoadRun
	q_xAlarm
	q_xTripped
	q_xReadyToReset
	q_xAssetAlarm
	q_udiIRMSAvg
	q_udtPVControl
	q_udtControl

The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwd	BOOL	If this input is TRUE, the avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xUpstrVPres3	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the third starter/power device in this avatar.	Cyclic data
q_xRunLineFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunY	BOOL	If this output is set to TRUE, the Y switch for Motor Y/D avatar is closed.	Cyclic data
q_xRunD	BOOL	If this output is set to TRUE, the D switch for Motor Y/D avatar is closed.	Cyclic data
q_xBypass	BOOL	If this output is set to TRUE, the avatar Bypass command is set.	Cyclic data
q_xRunLocalFwd	BOOL	If this output is set to TRUE, the avatar Local Forward command is set.	Cyclic data
q_xOverrideStatus	BOOL	If this output is set to TRUE, the avatar Manual mode is set.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[37] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtPVControl	UDT_ PVControl	This is a structure of cyclic PV status data for Control avatars. For detailed information about the structure of this UDT, see UDT_PVControl, page 58.	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 184.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

[37] Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	Yes
SILGroup	n/A	No
ThermalCapacity	n/A	Yes
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	Yes
TripMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	LongStart	Yes
	Jam	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	Yes
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	Yes
TimeToTrip	n/A	Yes
TimeToReset	n/A	Yes
PAStatusReg1	PA0Status	Yes
	PA1Status	Yes
	PA2Status	Yes
	PA3Status	Yes
	PA4Status	Yes
	PA5Status	Yes
	PA6Status	Yes
	PA7Status	Yes
	PA8Status	Yes
	PA9Status	Yes

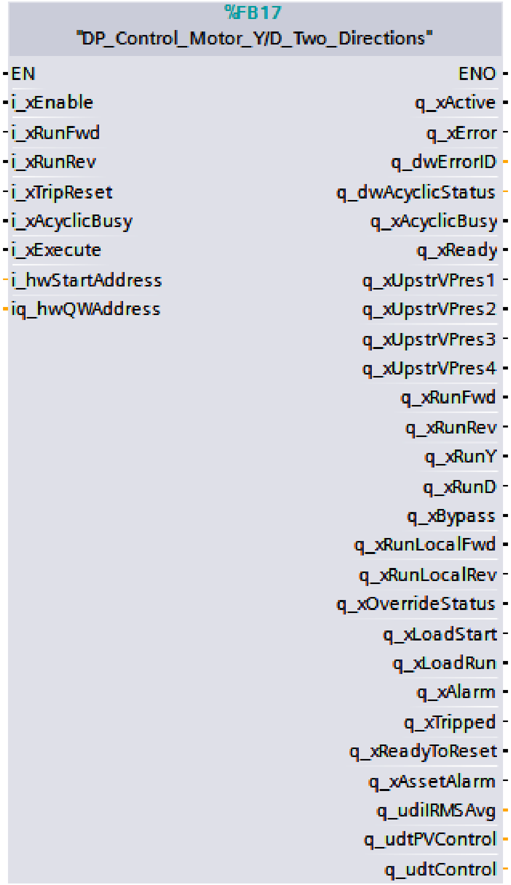
Motor Y/D, Two Directions

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Load avatar “Motor Y/D, Two Directions“.
Functionality	The Motor YD Two Direction function block is used to manage a wye/delta (star/triangle) motor in two directions (forward and reverse).

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Motor_Y/D_Two_Directions (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwd	BOOL	If this input is TRUE, the avatar forward switch is closed.	Cyclic data
i_xRunRev	BOOL	If this input is TRUE, the avatar reverse switch is closed.	Cyclic data

Inputs (Continued)

Input	Data Type	Description	Variable Category
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xUpstrVPres3	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the third starter/power device in this avatar.	Cyclic data
q_xUpstrVPres4	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the fourth starter/power device in this avatar.	Cyclic data
q_xRunFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunRev	BOOL	If this output is set to TRUE, the avatar reverse switch is closed.	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_xRunY	BOOL	If this output is set to TRUE, the Y switch for Motor Y/D avatar is closed.	Cyclic data
q_xRunD	BOOL	If this output is set to TRUE, the D switch for Motor Y/D avatar is closed.	Cyclic data
q_xBypass	BOOL	If this output is set to TRUE, the avatar Bypass command is set.	Cyclic data
q_xRunLocalFwd	BOOL	If this output is set to TRUE, the avatar Local Forward command is set.	Cyclic data
q_xRunLocalRev	BOOL	If this output is set to TRUE, the avatar Local Reverse command is set.	Cyclic data
q_xOverrideStatus	BOOL	If this output is set to TRUE, the avatar Manual mode is set.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[38] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtPVControl	UDT_ PVControl	This is a structure of cyclic PV status data for Control avatars. For detailed information about the structure of this UDT, see UDT_PVControl, page 58.	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 192.	Acyclic data

Status and error codes from ConstTeSysIsland

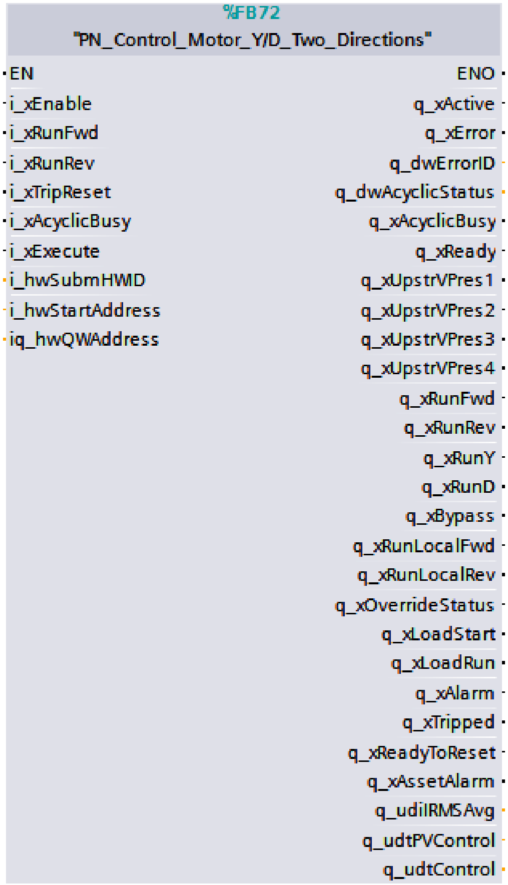
Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes

[38] Safety Integrity Level according to standard IEC 61508.

Status and error codes from ConstTeSysIsland (Continued)

Member name	Used by the function block
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

PN_Control_Motor_Y/D_Two_Directions (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwd	BOOL	If this input is TRUE, the avatar forward switch is closed.	Cyclic data
i_xRunRev	BOOL	If this input is TRUE, the avatar reverse switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data

Inputs (Continued)

Input	Data Type	Description	Variable Category
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xUpstrVPres3	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the third starter/power device in this avatar.	Cyclic data
q_xUpstrVPres4	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the fourth starter/power device in this avatar.	Cyclic data
q_xRunFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_xRunRev	BOOL	If this output is set to TRUE, the avatar reverse switch is closed.	Cyclic data
q_xRunY	BOOL	If this output is set to TRUE, the Y switch for Motor Y/D avatar is closed.	Cyclic data
q_xRunD	BOOL	If this output is set to TRUE, the D switch for Motor Y/D avatar is closed.	Cyclic data
q_xBypass	BOOL	If this output is set to TRUE, the avatar Bypass command is set.	Cyclic data
q_xRunLocalFwd	BOOL	If this output is set to TRUE, the avatar Local Forward command is set.	Cyclic data
q_xRunLocalRev	BOOL	If this output is set to TRUE, the avatar Local Reverse command is set.	Cyclic data
q_xOverrideStatus	BOOL	If this output is set to TRUE, the avatar Manual mode is set.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[39] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtPVControl	UDT_ PVControl	This is a structure of cyclic PV status data for Control avatars. For detailed information about the structure of this UDT, see UDT_PVControl, page 58.	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 192.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No

[39] Safety Integrity Level according to standard IEC 61508.

Status and error codes from ConstTeSysIsland (Continued)

Member name	Used by the function block
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	Yes
SILGroup	n/A	No
ThermalCapacity	n/A	Yes
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	Yes
TripMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	LongStart	Yes
	Jam	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	Yes
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	Yes
TimeToTrip	n/A	Yes
TimeToReset	n/A	Yes

Elements (Continued)

Element	Sub-Element	Supported
PAStatusReg1	PA0Status	Yes
	PA1Status	Yes
	PA2Status	Yes
	PA3Status	Yes
	PA4Status	Yes
	PA5Status	Yes
	PA6Status	Yes
	PA7Status	Yes
	PA8Status	Yes
	PA9Status	Yes

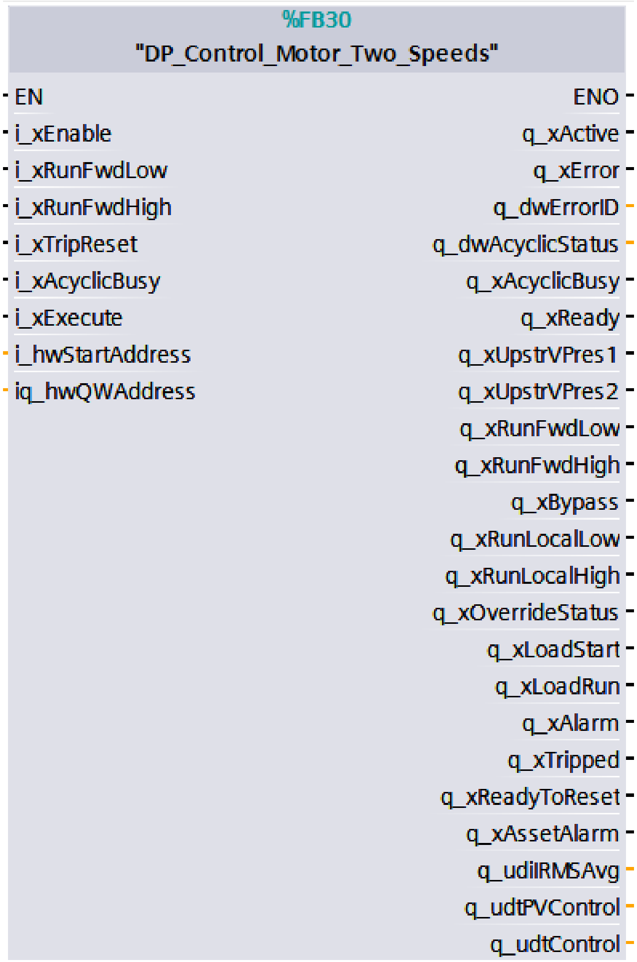
Motor Two Speeds

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Load avatar “Motor Two Speeds“.
Functionality	The Motor Two Speeds function block is used to manage a two–speed motor.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Motor_Two_Speeds (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwdLow	BOOL	If this input is set to TRUE, the motor starts in forward direction with low speed.	Cyclic data
i_xRunFwdHigh	BOOL	If this input is set to TRUE, the motor starts in forward direction with high speed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xRunFwdLow	BOOL	If this output is set to TRUE, the motor is running in low speed. If this output is set to FALSE, the motor is stopped or is running in high speed.	Cyclic data
q_xRunFwdHigh	BOOL	If this output is set to TRUE, the motor is running in high speed. If this output is set to FALSE, the motor is stopped or is running in low speed.	Cyclic data
q_xBypass	BOOL	If this output is set to TRUE, the avatar Bypass command is set.	Cyclic data
q_xRunLocalLow	BOOL	If this output is set to TRUE, the avatar Local Low Speed command is set.	Cyclic data
q_xRunLocalHigh	BOOL	If this output is set to TRUE, the avatar Local High Speed command is set.	Cyclic data
q_xOverrideStatus	BOOL	If this output is set to TRUE, the avatar Manual mode is set.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data

Outputs (Continued)

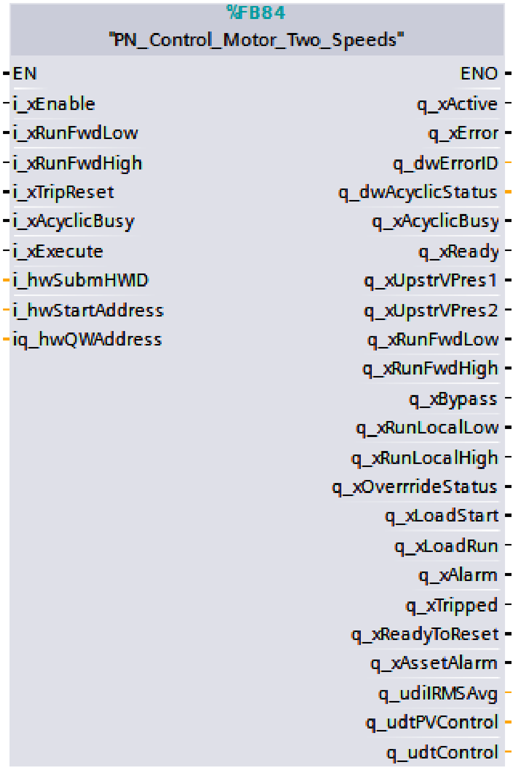
Output	Data Type	Description	Variable Category
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[40] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtPVControl	UDT_ PVControl	This is a structure of cyclic PV status data for Control avatars. For detailed information about the structure of this UDT, see UDT_PVControl, page 58.	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 199.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the Description
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

[40] Safety Integrity Level according to standard IEC 61508.

PN_Control_Motor_Two_Speeds (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwdLow	BOOL	If this input is set to TRUE, the motor starts in forward direction with low speed.	Cyclic data
i_xRunFwdHigh	BOOL	If this input is set to TRUE, the motor starts in forward direction with high speed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xRunFwdLow	BOOL	If this output is set to TRUE, the motor is running in low speed. If this output is set to FALSE, the motor is stopped or is running in high speed.	Cyclic data
q_xRunFwdHigh	BOOL	If this output is set to TRUE, the motor is running in high speed. If this output is set to FALSE, the motor is stopped or is running in low speed.	Cyclic data
q_xBypass	BOOL	If this output is set to TRUE, the avatar Bypass command is set.	Cyclic data
q_xRunLocalLow	BOOL	If this output is set to TRUE, the avatar Local Low Speed command is set.	Cyclic data
q_xRunLocalHigh	BOOL	If this output is set to TRUE, the avatar Local High Speed command is set.	Cyclic data
q_xOverrideStatus	BOOL	If this output is set to TRUE, the avatar Manual mode is set.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[41] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtPVControl	UDT_PVControl	This is a structure of cyclic PV status data for Control avatars. For detailed information about the structure of this UDT, see UDT_PVControl, page 58.	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 199.	Acyclic data

Status an error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	Yes
SILGroup	n/A	No
ThermalCapacity	n/A	Yes

[41] Safety Integrity Level according to standard IEC 61508.

Elements (Continued)

Element	Sub-Element	Supported
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	Yes
TripMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	LongStart	Yes
	Jam	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	Yes
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	Yes
TimeToTrip	n/A	Yes
TimeToReset	n/A	Yes
PAStatusReg1	PA0Status	Yes
	PA1Status	Yes
	PA2Status	Yes
	PA3Status	Yes
	PA4Status	Yes
	PA5Status	Yes
	PA6Status	Yes
	PA7Status	Yes
	PA8Status	Yes
	PA9Status	Yes

Motor Two Speeds – SIL Stop, W. Cat. 1/2

NOTE: Safety Integrity Level according to standard IEC 61508. Wiring Category 1 and Category 2 according to ISO 13849.

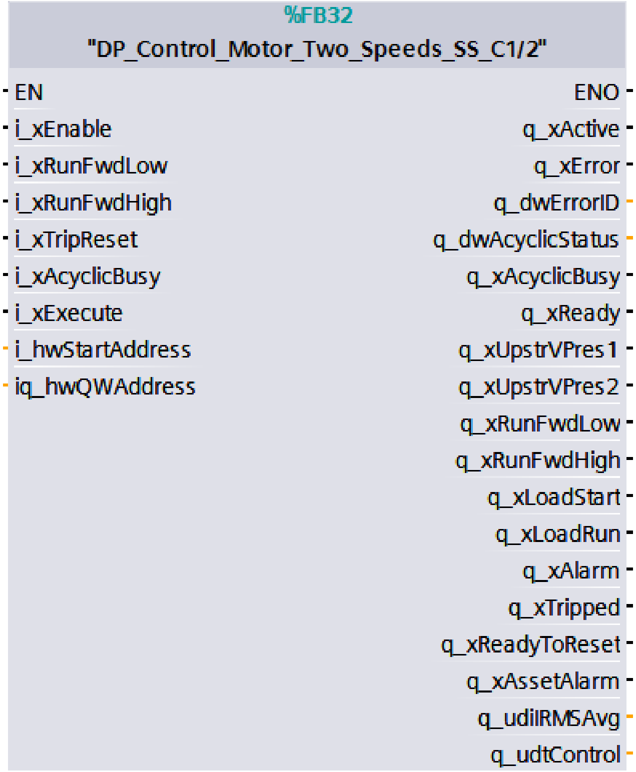
Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Load avatar “Motor Two Speeds – SIL Stop, W. Cat. 1/2“.
Functionality	The Motor Two Speeds SIL function block is used to manage a two speed motor with Stop Category 0 or Stop Category 1[42], Wiring Category 1 and Category 2.

[42] Stop categories according to EN/IEC 60204-1.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Motor_Two_Speeds_SS_C1/2 (PROFIBUS DP)



%FB32
"DP_Control_Motor_Two_Speeds_SS_C1/2"

EN	ENO
i_xEnable	q_xActive
i_xRunFwdLow	q_xError
i_xRunFwdHigh	q_dwErrorID
i_xTripReset	q_dwAcyclicStatus
i_xAcyclicBusy	q_xAcyclicBusy
i_xExecute	q_xReady
i_hwStartAddress	q_xUpstrVPres1
iq_hwQWAddress	q_xUpstrVPres2
	q_xRunFwdLow
	q_xRunFwdHigh
	q_xLoadStart
	q_xLoadRun
	q_xAlarm
	q_xTripped
	q_xReadyToReset
	q_xAssetAlarm
	q_udiIRMSAvg
	q_udtControl

The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwdLow	BOOL	If this input is set to TRUE, the motor starts in forward direction with low speed.	Cyclic data
i_xRunFwdHigh	BOOL	If this input is set to TRUE, the motor starts in forward direction with high speed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xRunFwdLow	BOOL	If this output is set to TRUE, the motor is running in low speed. If this output is set to FALSE, the motor is stopped or is running in high speed.	Cyclic data
q_xRunFwdHigh	BOOL	If this output is set to TRUE, the motor is running in high speed. If this output is set to FALSE, the motor is stopped or is running in low speed.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[43] Starter within the avatar	Cyclic data

[43] Safety Integrity Level according to standard IEC 61508.

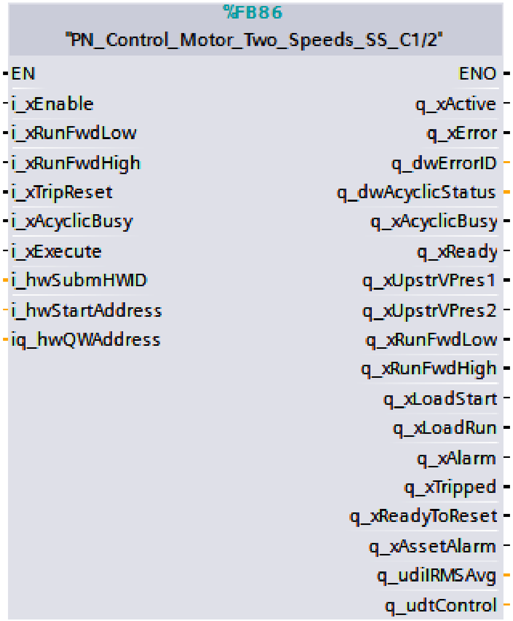
Outputs (Continued)

Output	Data Type	Description	Variable Category
		has reached or exceeded 90% of the expected durability (per the avatar parameter).	
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A).	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 206.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

PN_Control_Motor_Two_Speeds_SS_C1/2 (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwdLow	BOOL	If this input is set to TRUE, the motor starts in forward direction with low speed.	Cyclic data
i_xRunFwdHigh	BOOL	If this input is set to TRUE, the motor starts in forward direction with high speed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xRunFwdLow	BOOL	If this output is set to TRUE, the motor is running in low speed. If this output is set to FALSE, the motor is stopped or is running in high speed.	Cyclic data
q_xRunFwdHigh	BOOL	If this output is set to TRUE, the motor is running in high speed. If this output is set to FALSE, the motor is stopped or is running in low speed.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[44] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 206.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes

[44] Safety Integrity Level according to standard IEC 61508.

Status and error codes from ConstTeSysIsland (Continued)

Member name	Used by the function block
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	Yes
SILGroup	n/A	Yes
ThermalCapacity	n/A	Yes
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	Yes
TripMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	LongStart	Yes
	Jam	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	Yes
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	Yes
TimeToTrip	n/A	Yes
TimeToReset	n/A	Yes

Elements (Continued)

Element	Sub-Element	Supported
PAStatusReg1	PA0Status	No
	PA1Status	No
	PA2Status	No
	PA3Status	No
	PA4Status	No
	PA5Status	No
	PA6Status	No
	PA7Status	No
	PA8Status	No
	PA9Status	No

Motor Two Speeds – SIL Stop, W. Cat. 3/4

NOTE: Safety Integrity Level according to standard IEC 61508. Wiring Category 3 and Category 4 according to ISO 13849.

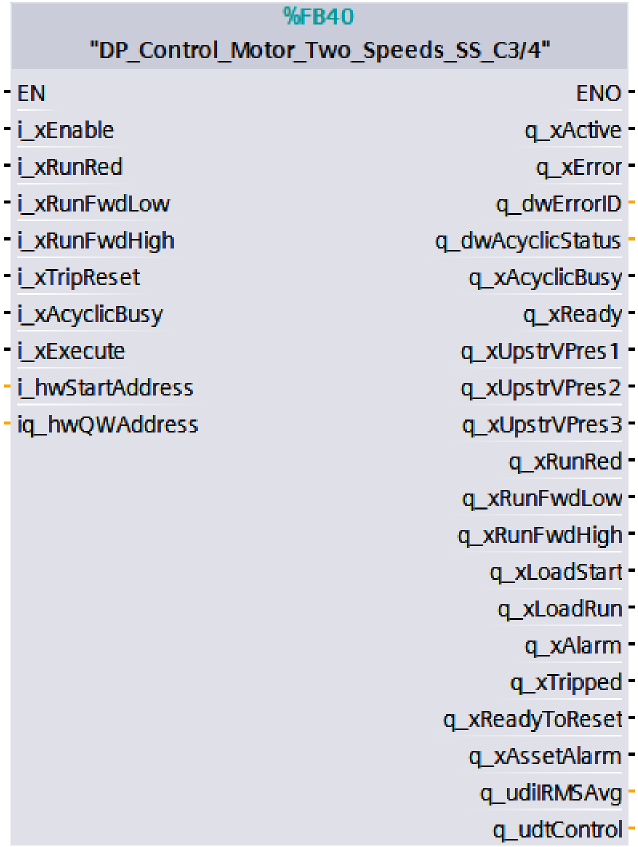
Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Load avatar “Motor Two Speeds – SIL Stop, W. Cat. 3/4“
Functionality	The Motor Two Speeds SIL function block is used to manage a two speed motor with Stop Category 0 or Stop Category 1[45], Wiring Category 3 and Category 4.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

[45] Stop categories according to EN/IEC 60204-1.

DP_Control_Motor_Two_Speeds_SS_C3/4 (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunRed	BOOL	If this input is TRUE, the redundant avatar switch is closed.	Cyclic data
i_xRunFwdLow	BOOL	If this input is set to TRUE, the motor starts in forward direction with low speed.	Cyclic data
i_xRunFwdHigh	BOOL	If this input is set to TRUE, the motor starts in forward direction with high speed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xUpstrVPres3	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the third starter/power device in this avatar.	Cyclic data
q_xRunRed	BOOL	If this input is TRUE, the redundant avatar switch is closed.	Cyclic data
q_xRunFwdLow	BOOL	If this output is set to TRUE, the motor is running in low speed. If this output is set to FALSE, the motor is stopped or is running in high speed.	Cyclic data
q_xRunFwdHigh	BOOL	If this output is set to TRUE, the motor is running in high speed. If this output is set to FALSE, the motor is stopped or is running in low speed.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[46] Starter within the avatar has reached or exceeded 90% of the	Cyclic data

[46] Safety Integrity Level according to standard IEC 61508.

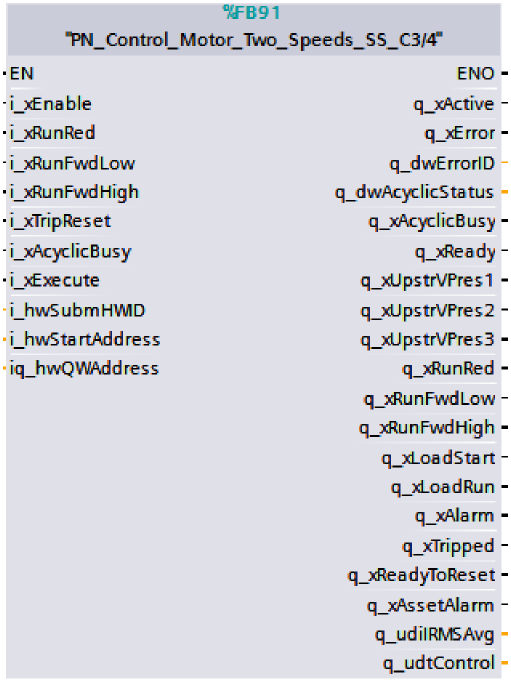
Outputs (Continued)

Output	Data Type	Description	Variable Category
		expected durability (per the avatar parameter).	
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 213.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

PN_Control_Motor_Two_Speeds_SS_C3/4 (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunRed	BOOL	If this input is TRUE, the redundant avatar switch is closed.	Cyclic data
i_xRunFwdLow	BOOL	If this input is set to TRUE, the motor starts in forward direction with low speed.	Cyclic data
i_xRunFwdHigh	BOOL	If this input is set to TRUE, the motor starts in forward direction with high speed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xUpstrVPres3	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the third starter/power device in this avatar.	Cyclic data
q_xRunRed	BOOL	If this input is TRUE, the redundant avatar switch is closed.	Cyclic data
q_xRunFwdLow	BOOL	If this output is set to TRUE, the motor is running in low speed. If this output is set to FALSE, the motor is stopped or is running in high speed.	Cyclic data
q_xRunFwdHigh	BOOL	If this output is set to TRUE, the motor is running in high speed. If this output is set to FALSE, the motor is stopped or is running in low speed.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[47] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 213.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	Yes
SILGroup	n/A	Yes
ThermalCapacity	n/A	Yes

[47] Safety Integrity Level according to standard IEC 61508.

Elements (Continued)

Element	Sub-Element	Supported
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	Yes
TripMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	LongStart	Yes
	Jam	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	Yes
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	Yes
TimeToTrip	n/A	Yes
TimeToReset	n/A	Yes
PAStatusReg1	PA0Status	No
	PA1Status	No
	PA2Status	No
	PA3Status	No
	PA4Status	No
	PA5Status	No
	PA6Status	No
	PA7Status	No
	PA8Status	No
	PA9Status	No

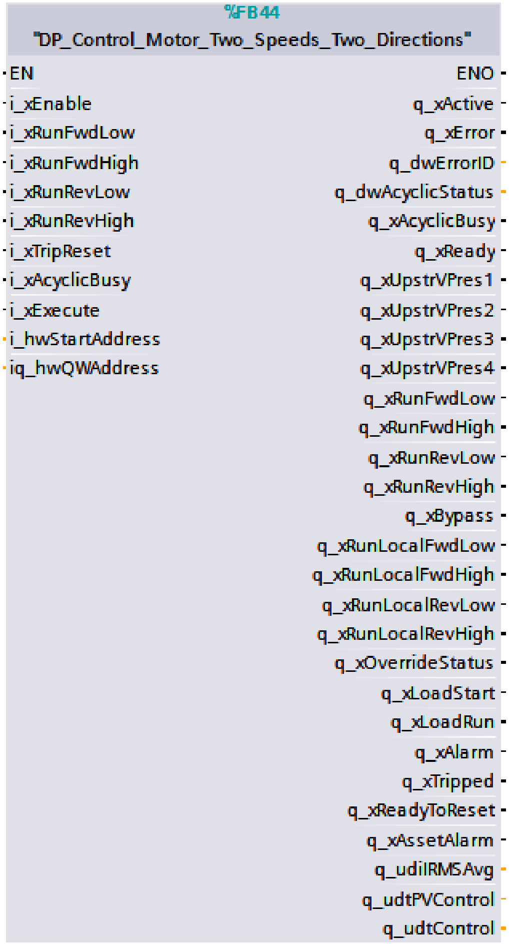
Motor Two Speeds, Two Directions

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Load avatar “Motor Two Speeds, Two Directions“.
Functionality	The Motor Two Speeds Two Directions function block is used to manage a two speed motor in two directions (forward and reverse).

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Motor_Two_Speeds_Two_Directions_ (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwdLow	BOOL	If this input is set to TRUE, the motor starts in forward direction with low speed.	Cyclic data

Inputs (Continued)

Input	Data Type	Description	Variable Category
i_xRunFwdHigh	BOOL	If this input is set to TRUE, the motor starts in forward direction with high speed.	Cyclic data
i_xRunRevLow	BOOL	If this input is set to TRUE, the motor starts in reverse direction with low speed.	Cyclic data
i_xRunRevHigh	BOOL	If this input is set to TRUE, the motor starts in reverse direction with high speed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xUpstrVPres3	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the third starter/power device in this avatar.	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_xUpstrVPres4	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the fourth starter/power device in this avatar.	Cyclic data
q_xRunFwdLow	BOOL	If this output is set to TRUE, the motor is running in low speed. If this output is set to FALSE, the motor is stopped or is running in high speed.	Cyclic data
q_xRunFwdHigh	BOOL	If this output is set to TRUE, the motor is running in high speed. If this output is set to FALSE, the motor is stopped or is running in low speed.	Cyclic data
q_xRunRevLow	BOOL	If this output is set to TRUE, the low speed reverser switch is closed.	Cyclic data
q_xRunRevHigh	BOOL	If this output is set to TRUE, the high speed reverser switch is closed.	Cyclic data
q_xBypass	BOOL	If this output is set to TRUE, the avatar Bypass command is set.	Cyclic data
q_xRunLocalFwdLow	BOOL	If this output is set to TRUE, the avatar Local Forward Low Speed command is set.	Cyclic data
q_xRunLocalFwdHigh	BOOL	If this output is set to TRUE, the avatar Local Forward High Speed command is set.	Cyclic data
q_xRunLocalRevLow	BOOL	If this output is set to TRUE, the avatar Local Reverse Low Speed command is set.	Cyclic data
q_xRunLocalRevHigh	BOOL	If this output is set to TRUE, the avatar Local Reverse High Speed command is set.	Cyclic data
q_xOverrideStatus	BOOL	If this output is set to TRUE, the avatar Manual mode is set.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[48] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data

[48] Safety Integrity Level according to standard IEC 61508.

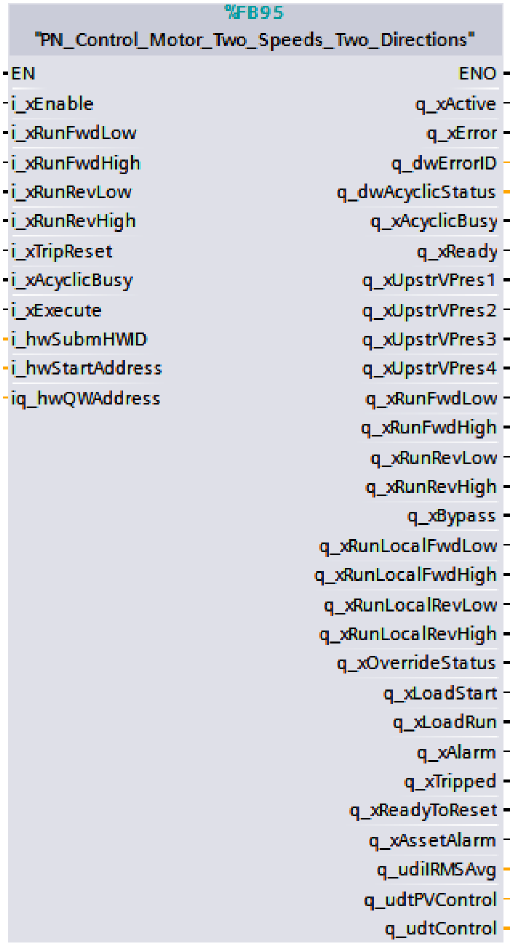
Outputs (Continued)

Output	Data Type	Description	Variable Category
q_udtPVControl	UDT_ PVControl	This is a structure of cyclic PV status data for Control avatars. For detailed information about the structure of this UDT, see UDT_PVControl, page 58.	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 222.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

PN_Control_Motor_Two_Speeds_Two_Directions (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwdLow	BOOL	If this input is set to TRUE, the motor starts in forward direction with low speed.	Cyclic data
i_xRunFwdHigh	BOOL	If this input is set to TRUE, the motor starts in forward direction with high speed.	Cyclic data
i_xRunRevLow	BOOL	If this input is set to TRUE, the motor starts in reverse direction with low speed.	Cyclic data
i_xRunRevHigh	BOOL	If this input is set to TRUE, the motor starts in reverse direction with high speed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xUpstrVPres3	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the third starter/power device in this avatar.	Cyclic data
q_xUpstrVPres4	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the fourth starter/power device in this avatar.	Cyclic data
q_xRunFwdLow	BOOL	If this output is set to TRUE, the motor is running in low speed. If this output is set to FALSE, the motor is stopped or is running in high speed.	Cyclic data
q_xRunFwdHigh	BOOL	If this output is set to TRUE, the motor is running in high speed. If this output is set to FALSE, the motor is stopped or is running in low speed.	Cyclic data
q_xRunRevLow	BOOL	If this output is set to TRUE, the low speed reverser switch is closed.	Cyclic data
q_xRunRevHigh	BOOL	If this output is set to TRUE, the high speed reverser switch is closed.	Cyclic data
q_xBypass	BOOL	If this output is set to TRUE, the avatar Bypass command is set.	Cyclic data
q_xRunLocalFwdLow	BOOL	If this output is set to TRUE, the avatar Local Forward Low Speed command is set.	Cyclic data
q_xRunLocalFwdHigh	BOOL	If this output is set to TRUE, the avatar Local Forward High Speed command is set.	Cyclic data
q_xRunLocalRevLow	BOOL	If this output is set to TRUE, the avatar Local Reverse Low Speed command is set.	Cyclic data
q_xRunLocalRevHigh	BOOL	If this output is set to TRUE, the avatar Local Reverse High Speed command is set.	Cyclic data
q_xOverrideStatus	BOOL	If this output is set to TRUE, the avatar Manual mode is set.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[49] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtPVControl	UDT_ PVControl	This is a structure of cyclic PV status data for Control avatars. For detailed information about the structure of this UDT, see UDT_PVControl, page 58.	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 222.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	Yes
SILGroup	n/A	No
ThermalCapacity	n/A	Yes

[49] Safety Integrity Level according to standard IEC 61508.

Elements (Continued)

Element	Sub-Element	Supported
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	Yes
TripMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	LongStart	Yes
	Jam	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	Yes
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	Yes
TimeToTrip	n/A	Yes
TimeToReset	n/A	Yes
PAStatusReg1	PA0Status	Yes
	PA1Status	Yes
	PA2Status	Yes
	PA3Status	Yes
	PA4Status	Yes
	PA5Status	Yes
	PA6Status	Yes
	PA7Status	Yes
	PA8Status	Yes
	PA9Status	Yes

Motor Two Speeds, Two Directions – SIL Stop, W. Cat. 1/2

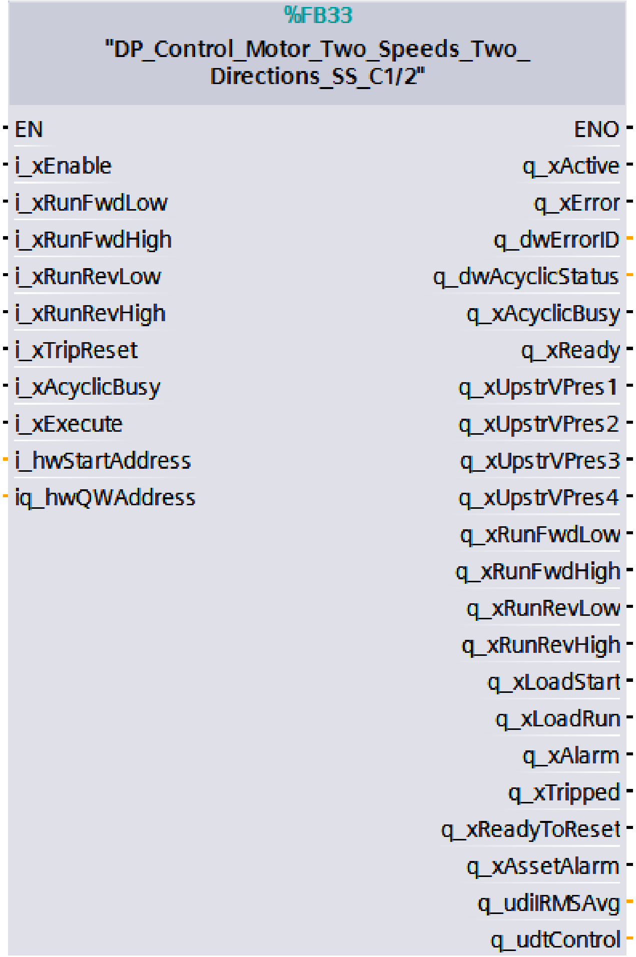
NOTE: Safety Integrity Level according to standard IEC 61508. Wiring Category 1 and Category 2 according to ISO 13849.

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Load avatar “Motor Two Speeds, Two Directions – SIL Stop, W. Cat. 1/2 “.
Functionality	The Motor Two Speeds Two Directions – SIL Stop, W. Cat. 1/2 function block is used to manage a two speed motor in two directions (forward and reverse) with Stop Category 0 or Stop Category 1[50], Wiring Category 1 and Category 2.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Motor_Two_Speeds_Two_Directions_SS_C1/2 (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

[50] Stop categories according to EN/IEC 60204-1.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwdLow	BOOL	If this input is set to TRUE, the motor starts in forward direction with low speed.	Cyclic data
i_xRunFwdHigh	BOOL	If this input is set to TRUE, the motor starts in forward direction with high speed.	Cyclic data
i_xRunRevLow	BOOL	If this input is set to TRUE, the motor starts in reverse direction with low speed.	Cyclic data
i_xRunRevHigh	BOOL	If this input is set to TRUE, the motor starts in reverse direction with high speed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
		present for the second starter/power device in this avatar.	
q_xUpstrVPres3	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the third starter/power device in this avatar.	Cyclic data
q_xUpstrVPres4	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the fourth starter/power device in this avatar.	Cyclic data
q_xRunFwdLow	BOOL	If this output is set to TRUE, the motor is running in low speed. If this output is set to FALSE, the motor is stopped or is running in high speed.	Cyclic data
q_xRunFwdHigh	BOOL	If this output is set to TRUE, the motor is running in high speed. If this output is set to FALSE, the motor is stopped or is running in low speed.	Cyclic data
q_xRunRevLow	BOOL	If this output is set to TRUE, the low speed reverser switch is closed.	Cyclic data
q_xRunRevHigh	BOOL	If this output is set to TRUE, the high speed reverser switch is closed.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[51] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 230.	Acyclic data

Status and error codes from ConstTeSysIsland

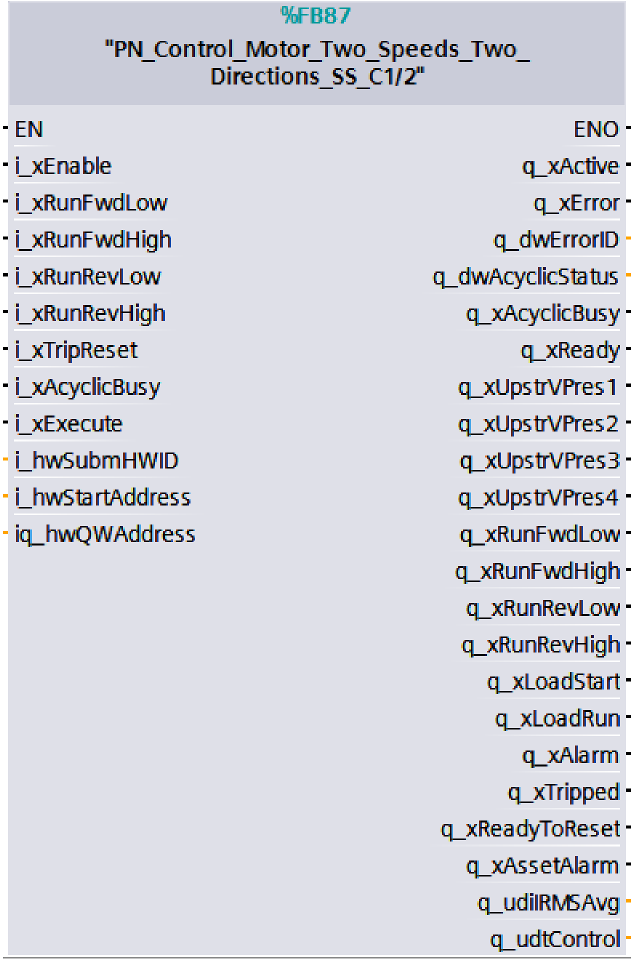
Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes

[51] Safety Integrity Level according to standard IEC 61508.

Status and error codes from ConstTeSysIsland (Continued)

Member name	Used by the function block
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

PN_Control_Motor_Two_Speeds_Two_Directions_SS_C1/2 (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwdLow	BOOL	If this input is set to TRUE, the motor starts in forward direction with low speed.	Cyclic data
i_xRunFwdHigh	BOOL	If this input is set to TRUE, the motor starts in forward direction with high speed.	Cyclic data
i_xRunRevLow	BOOL	If this input is set to TRUE, the motor starts in reverse direction with low speed.	Cyclic data
i_xRunRevHigh	BOOL	If this input is set to TRUE, the motor starts in reverse direction with high speed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
		present for the first starter/power device in this avatar.	
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xUpstrVPres3	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the third starter/power device in this avatar.	Cyclic data
q_xUpstrVPres4	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the fourth starter/power device in this avatar.	Cyclic data
q_xRunFwdLow	BOOL	If this output is set to TRUE, the motor is running in low speed. If this output is set to FALSE, the motor is stopped or is running in high speed.	Cyclic data
q_xRunFwdHigh	BOOL	If this output is set to TRUE, the motor is running in high speed. If this output is set to FALSE, the motor is stopped or is running in low speed.	Cyclic data
q_xRunRevLow	BOOL	If this output is set to TRUE, the low speed reverser switch is closed.	Cyclic data
q_xRunRevHigh	BOOL	If this output is set to TRUE, the high speed reverser switch is closed.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[52] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 230.	Acyclic data

[52] Safety Integrity Level according to standard IEC 61508.

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	Yes
SILGroup	n/A	Yes
ThermalCapacity	n/A	Yes
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	Yes
TripMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	LongStart	Yes
	Jam	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	Yes
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	Yes

Elements (Continued)

Element	Sub-Element	Supported
TimeToTrip	n/A	Yes
TimeToReset	n/A	Yes
PAStatusReg1	PA0Status	No
	PA1Status	No
	PA2Status	No
	PA3Status	No
	PA4Status	No
	PA5Status	No
	PA6Status	No
	PA7Status	No
	PA8Status	No
	PA9Status	No

Motor Two Speeds, Two Directions – SIL Stop, W. Cat. 3/4

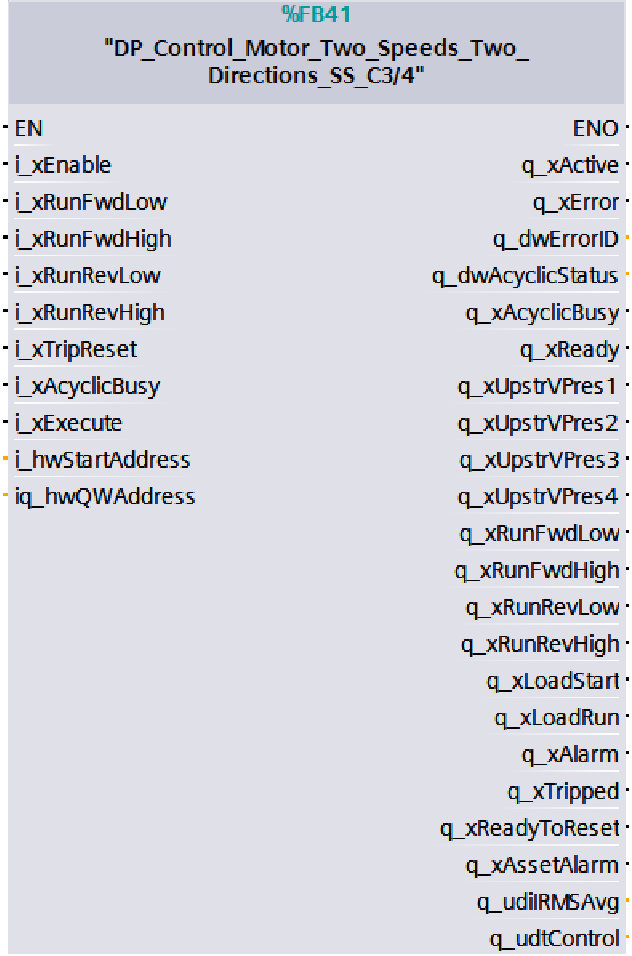
NOTE: Safety Integrity Level according to standard IEC 61508. Wiring Category 3 and Category 4 according to ISO 13849.

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Load avatar “Motor Two Speeds, Two Directions – SIL Stop, W. Cat. 3/4 “.
Functionality	The Motor Two Speeds Two Directions – SIL Stop, W. Cat. 3/4 function block is used to manage a two speed motor in two directions (forward and reverse) with Stop Category 0 or Stop Category 1[53], Wiring Category 3 and Category 4.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Motor_Two_Speeds_Two_Directions_SS_C3/4 (PROFIBUS DP)



[53] Stop categories according to EN/IEC 60204-1.

The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwdLow	BOOL	If this input is set to TRUE, the motor starts in forward direction with low speed.	Cyclic data
i_xRunFwdHigh	BOOL	If this input is set to TRUE, the motor starts in forward direction with high speed.	Cyclic data
i_xRunRevLow	BOOL	If this input is set to TRUE, the motor starts in reverse direction with low speed.	Cyclic data
i_xRunRevHigh	BOOL	If this input is set to TRUE, the motor starts in reverse direction with high speed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data

Outputs (Continued)

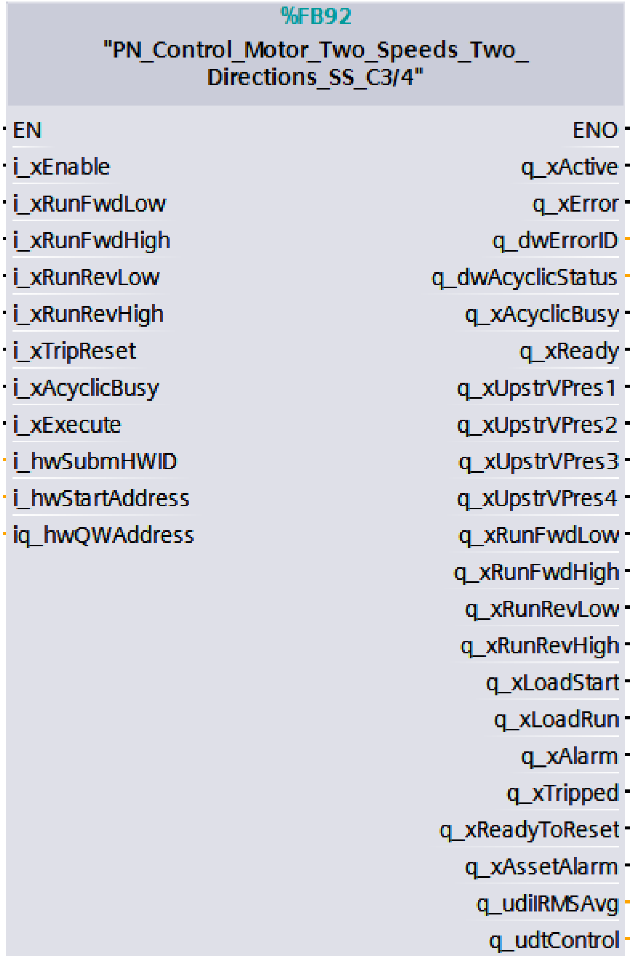
Output	Data Type	Description	Variable Category
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xUpstrVPres3	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the third starter/power device in this avatar.	Cyclic data
q_xUpstrVPres4	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the fourth starter/power device in this avatar.	Cyclic data
q_xRunFwdLow	BOOL	If this output is set to TRUE, the motor is running in low speed. If this output is set to FALSE, the motor is stopped or is running in high speed.	Cyclic data
q_xRunFwdHigh	BOOL	If this output is set to TRUE, the motor is running in high speed. If this output is set to FALSE, the motor is stopped or is running in low speed.	Cyclic data
q_xRunRevLow	BOOL	If this output is set to TRUE, the low speed reverser switch is closed.	Cyclic data
q_xRunRevHigh	BOOL	If this output is set to TRUE, the high speed reverser switch is closed.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[54] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 238.	Acyclic data

[54] Safety Integrity Level according to standard IEC 61508.

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

PN_Control_Motor_Two_Speeds_Two_Directions_SS_C3/4 (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwdLow	BOOL	If this input is set to TRUE, the motor starts in forward direction with low speed.	Cyclic data
i_xRunFwdHigh	BOOL	If this input is set to TRUE, the motor starts in forward direction with high speed.	Cyclic data
i_xRunRevLow	BOOL	If this input is set to TRUE, the motor starts in reverse direction with low speed.	Cyclic data
i_xRunRevHigh	BOOL	If this input is set to TRUE, the motor starts in reverse direction with high speed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xUpstrVPres3	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the third starter/power device in this avatar.	Cyclic data
q_xUpstrVPres4	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the fourth starter/power device in this avatar.	Cyclic data
q_xRunFwdLow	BOOL	If this output is set to TRUE, the motor is running in low speed. If this output is set to FALSE, the motor is stopped or is running in high speed.	Cyclic data
q_xRunFwdHigh	BOOL	If this output is set to TRUE, the motor is running in high speed. If this output is set to FALSE, the motor is stopped or is running in low speed.	Cyclic data
q_xRunRevLow	BOOL	If this output is set to TRUE, the low speed reverser switch is closed.	Cyclic data
q_xRunRevHigh	BOOL	If this output is set to TRUE, the high speed reverser switch is closed.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[55] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 238.	Acyclic data

[55] Safety Integrity Level according to standard IEC 61508.

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	Yes
SILGroup	n/A	Yes
ThermalCapacity	n/A	Yes
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	Yes
TripMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	LongStart	Yes
	Jam	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	Yes
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	Yes

Elements (Continued)

Element	Sub-Element	Supported
TimeToTrip	n/A	Yes
TimeToReset	n/A	Yes
PAStatusReg1	PA0Status	No
	PA1Status	No
	PA2Status	No
	PA3Status	No
	PA4Status	No
	PA5Status	No
	PA6Status	No
	PA7Status	No
	PA8Status	No
	PA9Status	No

Resistor

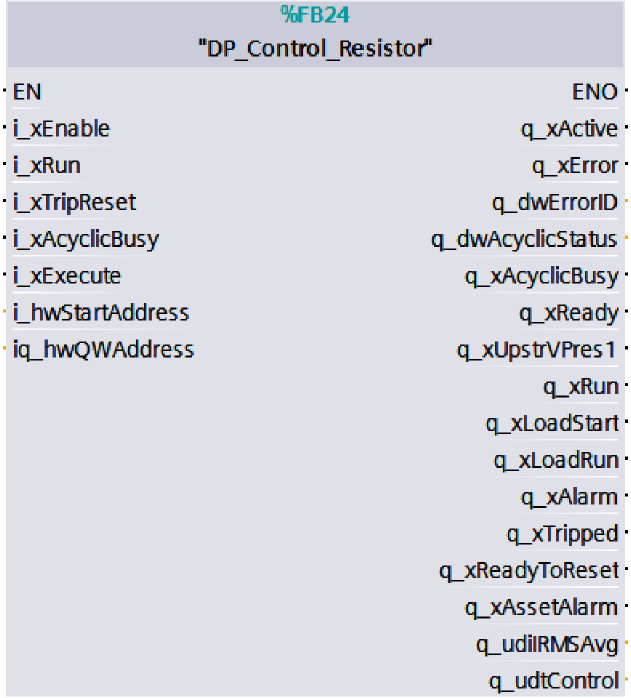
Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Load avatar “Resistor“.
Functionality	The Resistor function block is used to manage a resistive load.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Resistor (PROFIBUS DP)

The following tables give information about the parameters of the function block interface.



%FB24
"DP_Control_Resistor"

EN	ENO
i_xEnable	q_xActive
i_xRun	q_xError
i_xTripReset	q_dwErrorID
i_xAcyclicBusy	q_dwAcyclicStatus
i_xExecute	q_xAcyclicBusy
i_hwStartAddress	q_xReady
iq_hwQWAddress	q_xUpstrVPres1
	q_xRun
	q_xLoadStart
	q_xLoadRun
	q_xAlarm
	q_xTripped
	q_xReadyToReset
	q_xAssetAlarm
	q_udiIRMSAvg
	q_udtControl

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRun	BOOL	If this input is set to TRUE, the avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xRun	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[56] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data

[56] Safety Integrity Level according to standard IEC 61508.

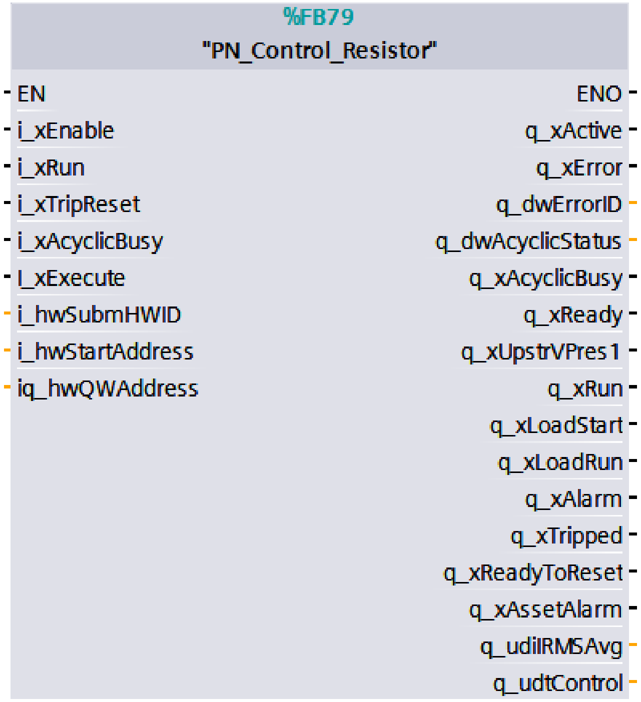
Outputs (Continued)

Output	Data Type	Description	Variable Category
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 244.	Acyclic data

Status and error codes from ConstTeSysIsand

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

PN_Control_Resistor (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRun	BOOL	If this input is set to TRUE, the avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xRun	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[57] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 244.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which

[57] Safety Integrity Level according to standard IEC 61508.

elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	No
SILGroup	n/A	No
ThermalCapacity	n/A	No
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	No
	PhaseUnbalance	Yes
	UnderCurrent	Yes
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	No
TripMsg1	GroundCurrent	Yes
	ThermalOverload	No
	LongStart	No
	Jam	No
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	No
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	No
TimeToTrip	n/A	No
TimeToReset	n/A	No
PAStatusReg1	PA0Status	No
	PA1Status	No
	PA2Status	No
	PA3Status	No
	PA4Status	No
	PA5Status	No
	PA6Status	No
	PA7Status	No
	PA8Status	No
	PA9Status	No

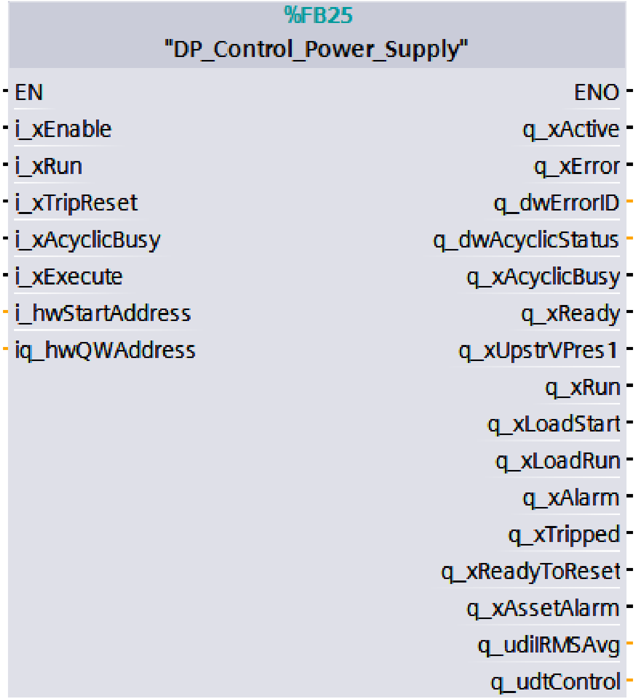
Power Supply

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Load avatar “Power Supply“.
Functionality	The Power Supply function block is used to manage a power supply.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Power_Supply (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRun	BOOL	If this input is set to TRUE, the avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control

Inputs (Continued)

Input	Data Type	Description	Variable Category
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xRun	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[58] Starter within the avatar	Cyclic data

[58] Safety Integrity Level according to standard IEC 61508.

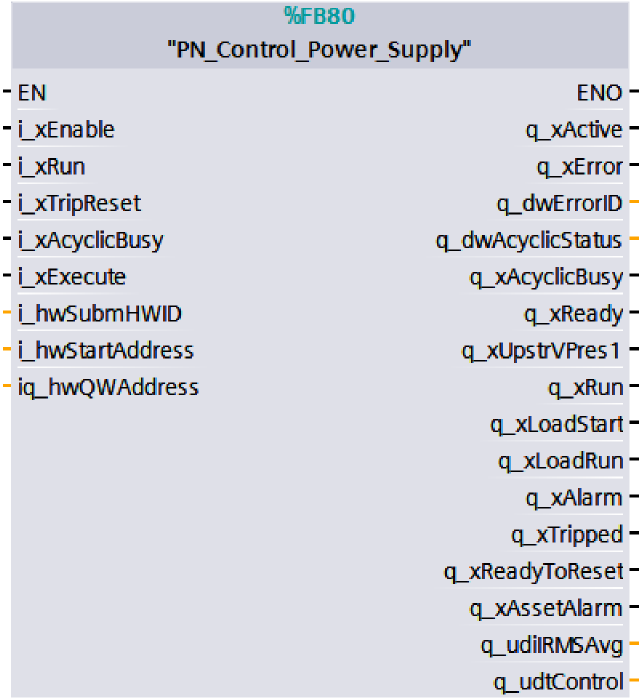
Outputs (Continued)

Output	Data Type	Description	Variable Category
		has reached or exceeded 90% of the expected durability (per the avatar parameter).	
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 251.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

PN_Control_Power_Supply (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRun	BOOL	If this input is set to TRUE, the avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xRun	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[59] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 251.	Acyclic data

[59] Safety Integrity Level according to standard IEC 61508.

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	No
SILGroup	n/A	No
ThermalCapacity	n/A	No
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	No
	PhaseUnbalance	Yes
	UnderCurrent	Yes
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	No
TripMsg1	GroundCurrent	Yes
	ThermalOverload	No
	LongStart	No
	Jam	No
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	No
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	No

Elements (Continued)

Element	Sub-Element	Supported
TimeToTrip	n/A	No
TimeToReset	n/A	No
PAStatusReg1	PA0Status	No
	PA1Status	No
	PA2Status	No
	PA3Status	No
	PA4Status	No
	PA5Status	No
	PA6Status	No
	PA7Status	No
	PA8Status	No
	PA9Status	No

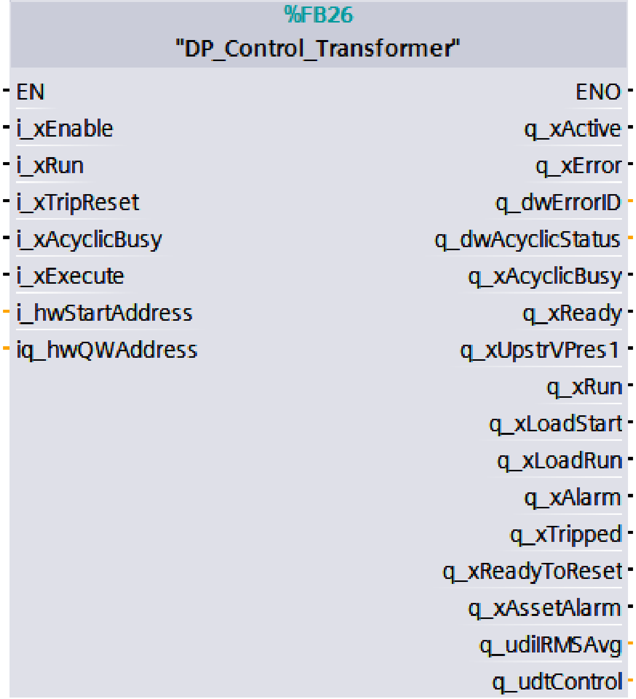
Transformer

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Load avatar “Transformer“.
Functionality	The Transformer function block is used to manage a transformer.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Transformer (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRun	BOOL	If this input is set to TRUE, the avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xRun	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[60] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data

[60] Safety Integrity Level according to standard IEC 61508.

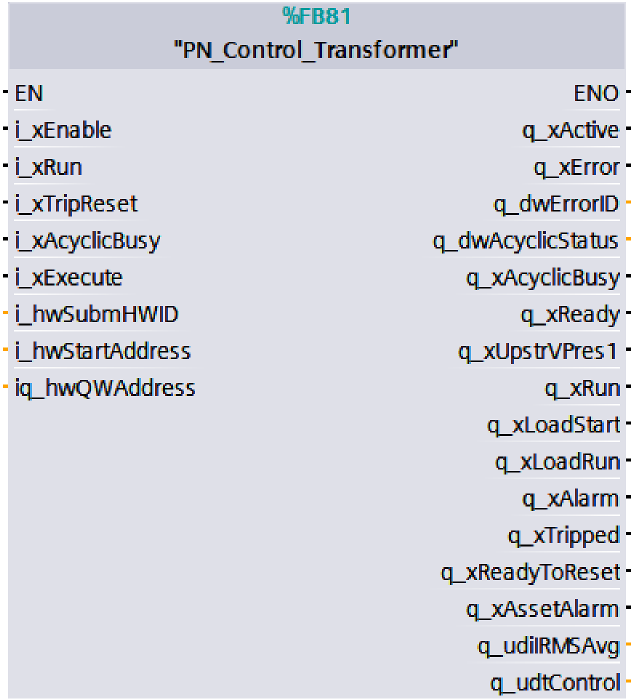
Outputs (Continued)

Output	Data Type	Description	Variable Category
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 257.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

PN_Control_Transformer (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRun	BOOL	If this input is set to TRUE, the avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xRun	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[61] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 257.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which

[61] Safety Integrity Level according to standard IEC 61508.

elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	No
SILGroup	n/A	No
ThermalCapacity	n/A	No
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	No
	PhaseUnbalance	Yes
	UnderCurrent	Yes
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	No
TripMsg1	GroundCurrent	Yes
	ThermalOverload	No
	LongStart	No
	Jam	No
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	No
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	No
TimeToTrip	n/A	No
TimeToReset	n/A	No
PAStatusReg1	PA0Status	No
	PA1Status	No
	PA2Status	No
	PA3Status	No
	PA4Status	No
	PA5Status	No
	PA6Status	No
	PA7Status	No
	PA8Status	No
	PA9Status	No

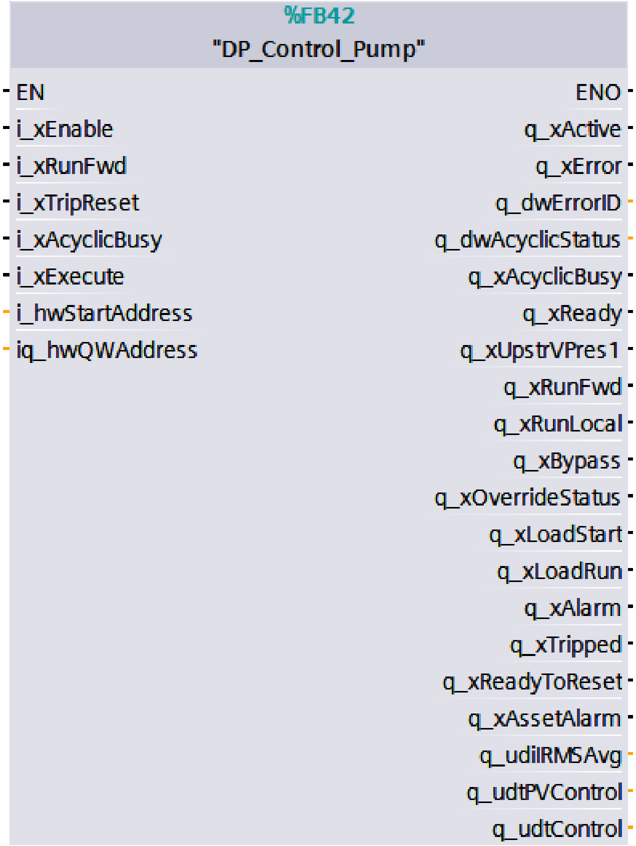
Pump

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Application avatar “Pump“.
Functionality	The Pump function block is used to manage a pump.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Pump (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwd	BOOL	If this input is set to TRUE, the avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control

Inputs (Continued)

Input	Data Type	Description	Variable Category
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xRunFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunLocal	BOOL	If this output is set to TRUE, the avatar Local Run command is set.	Cyclic data
q_xBypass	BOOL	If this output is set to TRUE, the avatar Bypass command is set.	Cyclic data
q_xOverrideStatus	BOOL	If this output is set to TRUE, the avatar Manual mode is set.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data

Outputs (Continued)

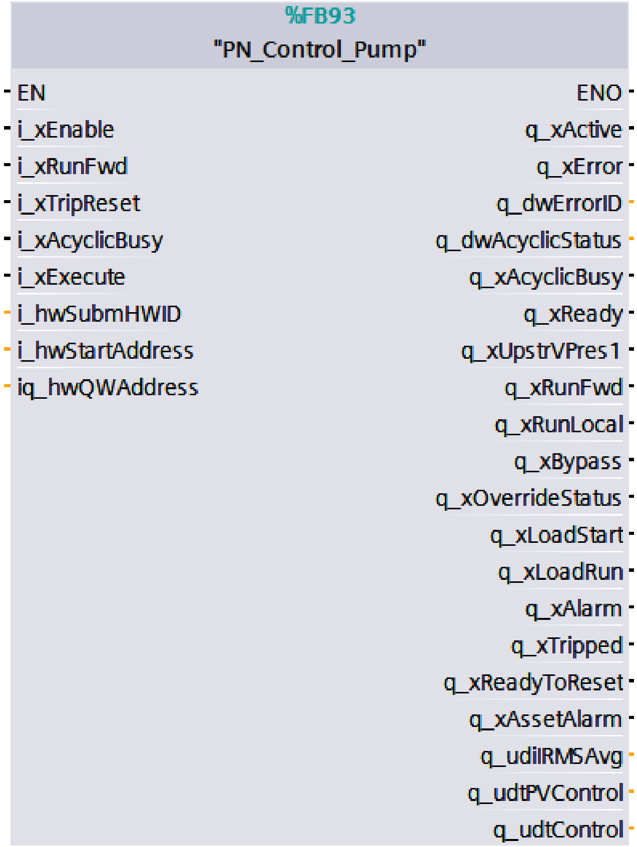
Output	Data Type	Description	Variable Category
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[62] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtPVControl	UDT_ PVControl	This is a structure of cyclic PV status data for Control avatars. For detailed information about the structure of this UDT, see UDT_PVControl, page 58.	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 264.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

[62] Safety Integrity Level according to standard IEC 61508.

PN_Control_Pump (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwd	BOOL	If this input is set to TRUE, the avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xRunFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunLocal	BOOL	If this output is set to TRUE, the avatar Local Run command is set.	Cyclic data
q_xBypass	BOOL	If this output is set to TRUE, the avatar Bypass command is set.	Cyclic data
q_xOverrideStatus	BOOL	If this output is set to TRUE, the avatar Manual mode is set.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[63] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data

[63] Safety Integrity Level according to standard IEC 61508.

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_udtPVControl	UDT_ PVControl	This is a structure of cyclic PV status data for Control avatars. For detailed information about the structure of this UDT, see UDT_PVControl, page 58.	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 264.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	Yes
SILGroup	n/A	No
ThermalCapacity	n/A	Yes
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	Yes

Elements (Continued)

Element	Sub-Element	Supported
TripMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	LongStart	No
	Jam	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	Yes
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	Yes
TimeToTrip	n/A	Yes
TimeToReset	n/A	Yes
PAStatusReg1	PA0Status	Yes
	PA1Status	Yes
	PA2Status	Yes
	PA3Status	Yes
	PA4Status	Yes
	PA5Status	Yes
	PA6Status	Yes
	PA7Status	Yes
	PA8Status	Yes
	PA9Status	Yes

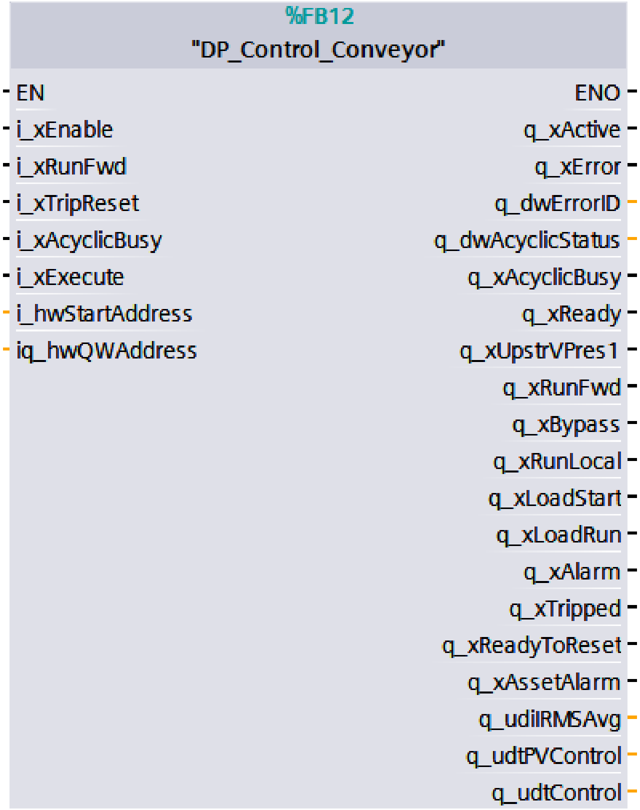
Conveyor, One Direction

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Application avatar “Conveyor, One Direction“.
Functionality	The Conveyor One Direction function block is used to manage a conveyor in one direction.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Conveyor (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwd	BOOL	If this input is set to TRUE, the avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control

Inputs (Continued)

Input	Data Type	Description	Variable Category
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xRunFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunLocal	BOOL	If this output is set to TRUE, the avatar Local Forward command is set.	Cyclic data
q_xBypass	BOOL	If this output is set to TRUE, the avatar Bypass command is set.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data

Outputs (Continued)

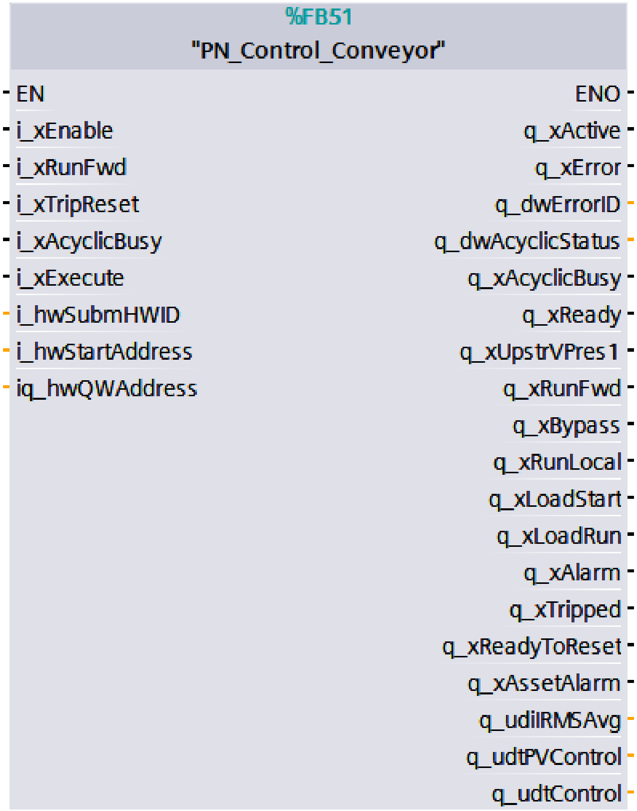
Output	Data Type	Description	Variable Category
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[64] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtPVControl	UDT_ PVControl	This is a structure of cyclic PV status data for Control avatars. For detailed information about the structure of this UDT, see UDT_PVControl, page 58..	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 271.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

[64] Safety Integrity Level according to standard IEC 61508.

PN_Control_Conveyor (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwd	BOOL	If this input is set to TRUE, the avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xRunFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunLocal	BOOL	If this output is set to TRUE, the avatar Local Forward command is set.	Cyclic data
q_xBypass	BOOL	If this output is set to TRUE, the avatar Bypass command is set.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[65] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data

[65] Safety Integrity Level according to standard IEC 61508.

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_udtPVControl	UDT_ PVControl	This is a structure of cyclic PV status data for Control avatars. For detailed information about the structure of this UDT, see UDT_PVControl, page 58.	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 271.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	Yes
SILGroup	n/A	No
ThermalCapacity	n/A	Yes
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	Yes

Elements (Continued)

Element	Sub-Element	Supported
TripMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	LongStart	Yes
	Jam	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	Yes
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	Yes
TimeToTrip	n/A	Yes
TimeToReset	n/A	Yes
PAStatusReg1	PA0Status	Yes
	PA1Status	Yes
	PA2Status	Yes
	PA3Status	Yes
	PA4Status	Yes
	PA5Status	Yes
	PA6Status	Yes
	PA7Status	Yes
	PA8Status	Yes
	PA9Status	Yes

Conveyor, One Direction – SIL Stop, W. Cat. 1/2

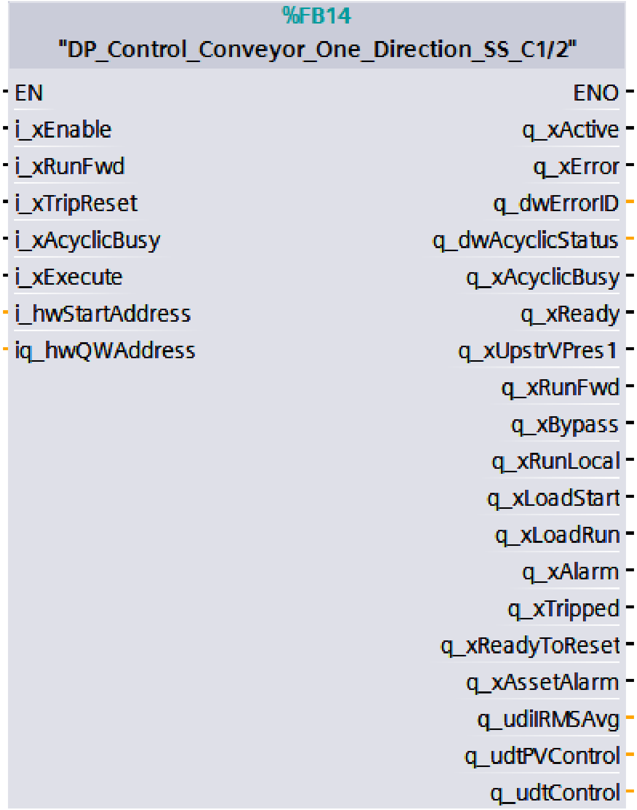
NOTE: Safety Integrity Level according to standard IEC 61508. Wiring Category 1 and Category 2 according to ISO 13849.

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Application avatar “Conveyor, One Direction – SIL Stop, W. Cat. 1/2“.
Functionality	The Conveyor One Direction – SIL Stop, W. Cat. 1/2 function block is used to manage a conveyor in one direction with Stop Category 0 or Stop Category 1[66], Wiring Category 1 and Category 2.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Conveyor_One_Direction_SS_C1/2 (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwd	BOOL	If this input is set to TRUE, the avatar forward switch is closed.	Cyclic data

[66] Stop categories according to EN/IEC 60204-1.

Inputs (Continued)

Input	Data Type	Description	Variable Category
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xRunFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunLocal	BOOL	If this output is set to TRUE, the avatar Local Forward command is set.	Cyclic data
q_xBypass	BOOL	If this output is set to TRUE, the avatar Bypass command is set.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data

Outputs (Continued)

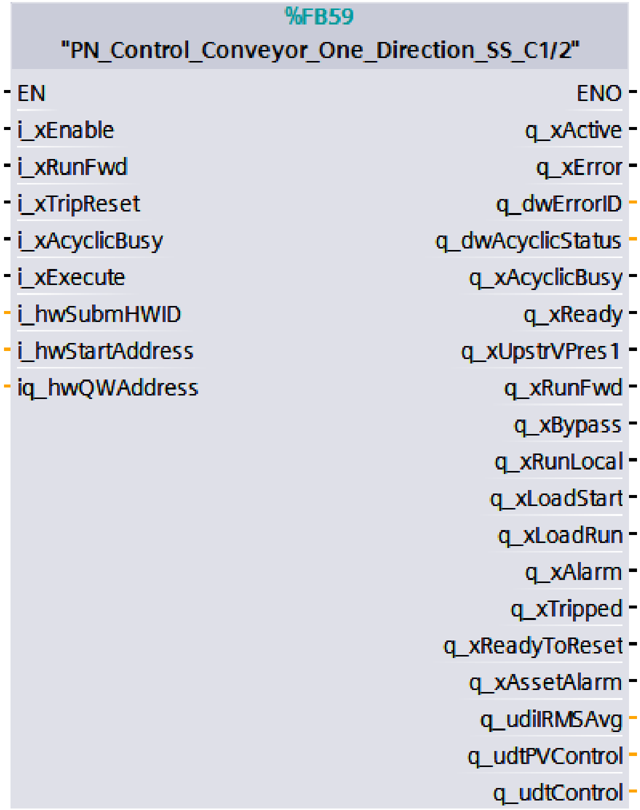
Output	Data Type	Description	Variable Category
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[67] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtPVControl	UDT_ PVControl	This is a structure of cyclic PV status data for Control avatars. For detailed information about the structure of this UDT, see UDT_PVControl, page 58.	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 278.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

[67] Safety Integrity Level according to standard IEC 61508.

PN_Control_Conveyor_One_Direction_SS_C1/2 (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwd	BOOL	If this input is set to TRUE, the avatar forward switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xRunFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunLocal	BOOL	If this output is set to TRUE, the avatar Local Forward command is set.	Cyclic data
q_xBypass	BOOL	If this output is set to TRUE, the avatar Bypass command is set.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[68] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data

[68] Safety Integrity Level according to standard IEC 61508.

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_udtPVControl	UDT_ PVControl	This is a structure of cyclic PV status data for Control avatars. For detailed information about the structure of this UDT, see UDT_PVControl, page 58.	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 278.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	Yes
SILGroup	n/A	Yes
ThermalCapacity	n/A	Yes
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	Yes

Elements (Continued)

Element	Sub-Element	Supported
TripMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	LongStart	Yes
	Jam	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	Yes
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	Yes
TimeToTrip	n/A	Yes
TimeToReset	n/A	Yes
PAStatusReg1	PA0Status	Yes
	PA1Status	Yes
	PA2Status	Yes
	PA3Status	Yes
	PA4Status	Yes
	PA5Status	Yes
	PA6Status	Yes
	PA7Status	Yes
	PA8Status	Yes
	PA9Status	Yes

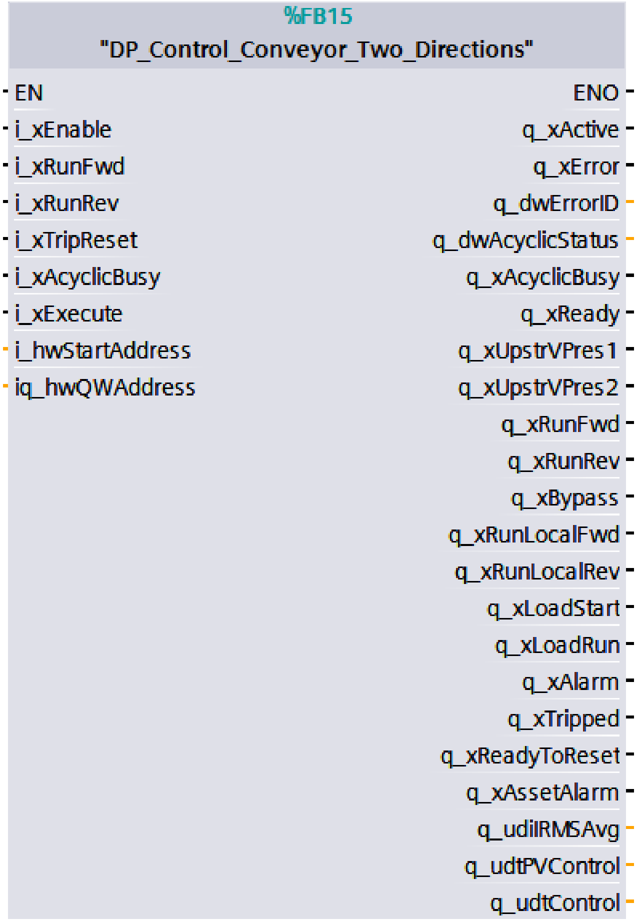
Conveyor, Two Directions

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Application avatar “Conveyor, Two Directions“.
Functionality	The Conveyor Two Directions function block is used to manage a conveyor in two directions (forward and reverse).

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Conveyor_Two_Directions (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwd	BOOL	If this input is set to TRUE, the avatar forward switch is closed.	Cyclic data
i_xRunRev	BOOL	If this input is TRUE, the avatar reverse switch is closed.	Cyclic data

Inputs (Continued)

Input	Data Type	Description	Variable Category
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xRunFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunRev	BOOL	If this output is set to TRUE, the avatar reverse switch is closed.	Cyclic data
q_xRunLocalFwd	BOOL	If this output is set to TRUE, the avatar Local Forward command is set.	Cyclic data
q_xRunLocalRev	BOOL	If this output is set to TRUE, the avatar Local Reverse command is set.	Cyclic data
q_xBypass	BOOL	If this output is set to TRUE, the avatar Bypass command is set.	Cyclic data

Outputs (Continued)

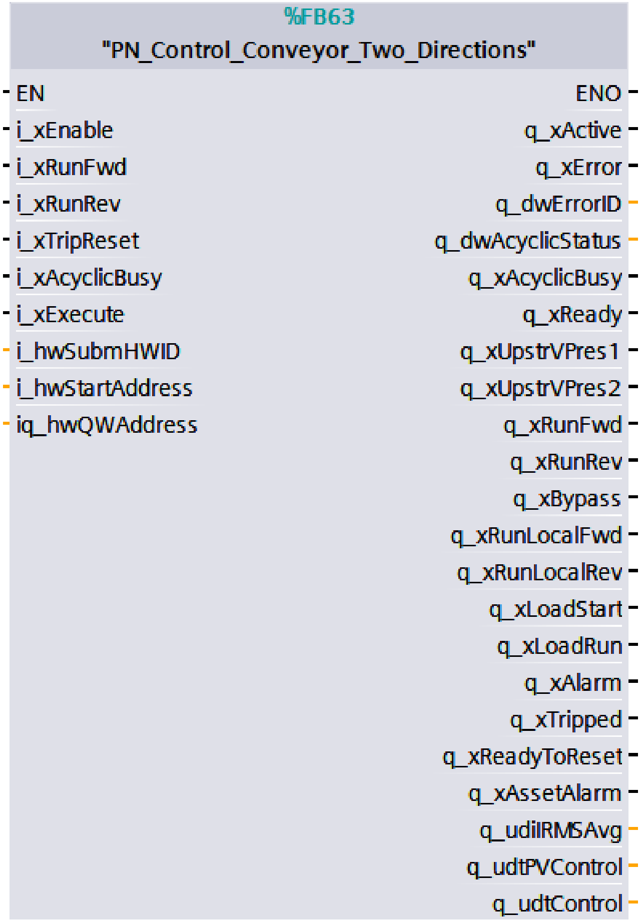
Output	Data Type	Description	Variable Category
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[69] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtPVControl	UDT_ PVControl	This is a structure of cyclic PV status data for Control avatars. For detailed information about the structure of this UDT, see UDT_PVControl, page 58.	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 285.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

[69] Safety Integrity Level according to standard IEC 61508.

PN_Control_Conveyor_Two_Directions (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwd	BOOL	If this input is set to TRUE, the avatar forward switch is closed.	Cyclic data
i_xRunRev	BOOL	If this input is TRUE, the avatar reverse switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xRunFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunRev	BOOL	If this output is set to TRUE, the avatar reverse switch is closed.	Cyclic data
q_xRunLocalFwd	BOOL	If this output is set to TRUE, the avatar Local Forward command is set.	Cyclic data
q_xRunLocalRev	BOOL	If this output is set to TRUE, the avatar Local Reverse command is set.	Cyclic data
q_xBypass	BOOL	If this output is set to TRUE, the avatar Bypass command is set.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[70] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtPVControl	UDT_ PVControl	This is a structure of cyclic PV status data for Control avatars. For detailed information about the structure of this UDT, see UDT_PVControl, page 58.	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 285.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Element	Sub-Element	Supported
MotorTemperature	n/A	Yes
SILGroup	n/A	No
ThermalCapacity	n/A	Yes

[70] Safety Integrity Level according to standard IEC 61508.

Element	Sub-Element	Supported
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	Yes
TripMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	LongStart	Yes
	Jam	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	Yes
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	Yes
TimeToTrip	n/A	Yes
TimeToReset	n/A	Yes
PAStatusReg1	PA0Status	Yes
	PA1Status	Yes
	PA2Status	Yes
	PA3Status	Yes
	PA4Status	Yes
	PA5Status	Yes
	PA6Status	Yes
	PA7Status	Yes
	PA8Status	Yes
	PA9Status	Yes

Conveyor, Two Directions – SIL Stop, W. Cat. 1/2

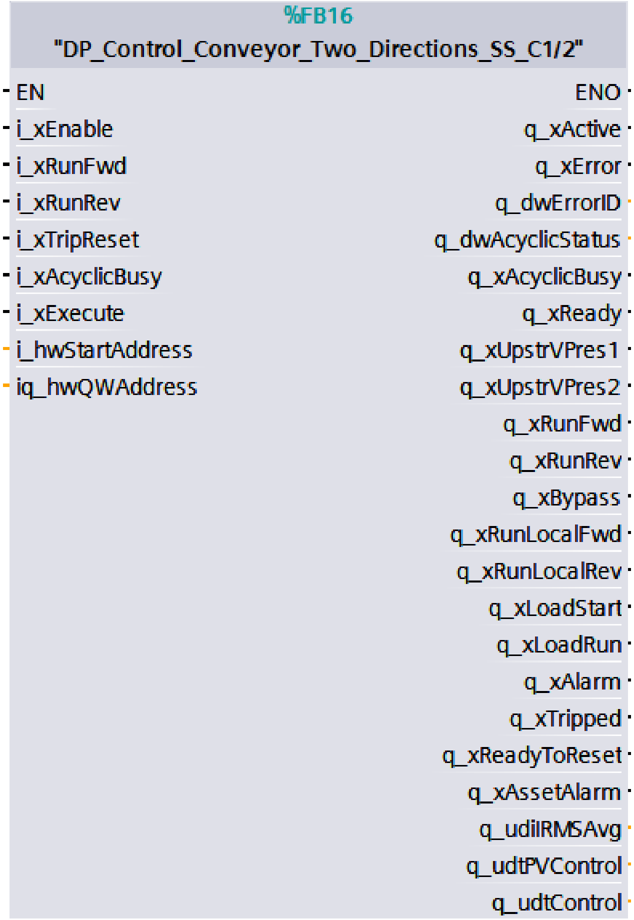
NOTE: Safety Integrity Level according to standard IEC 61508. Wiring Category 1 and Category 2 according to ISO 13849.

Function block profile

Function block type	Avatar Control function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	Application avatar “Conveyor, Two Directions – SIL Stop, W. Cat. 1/2“.
Functionality	The Conveyor Two Directions – SIL Stop, W. Cat. 1/2 function block is used to manage a conveyor in two directions (forward and reverse) with Stop Category 0 or Stop Category 1[71], Wiring Category 1 and Category 2.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Control_Conveyor_Two_Directions_SS_C1/2 (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

[71] Stop categories according to EN/IEC 60204-1.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwd	BOOL	If this input is set to TRUE, the avatar forward switch is closed.	Cyclic data
i_xRunRev	BOOL	If this input is TRUE, the avatar reverse switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xRunFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunRev	BOOL	If this output is set to TRUE, the avatar reverse switch is closed.	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_xRunLocalFwd	BOOL	If this output is set to TRUE, the avatar Local Forward command is set.	Cyclic data
q_xRunLocalRev	BOOL	If this output is set to TRUE, the avatar Local Reverse command is set.	Cyclic data
q_xBypass	BOOL	If this output is set to TRUE, the avatar Bypass command is set.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[72] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtPVControl	UDT_ PVControl	This is a structure of cyclic PV status data for Control avatars. For detailed information about the structure of this UDT, see UDT_PVControl, page 58.	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 293.	Acyclic data

Status and error codes from ConstTeSysIsland

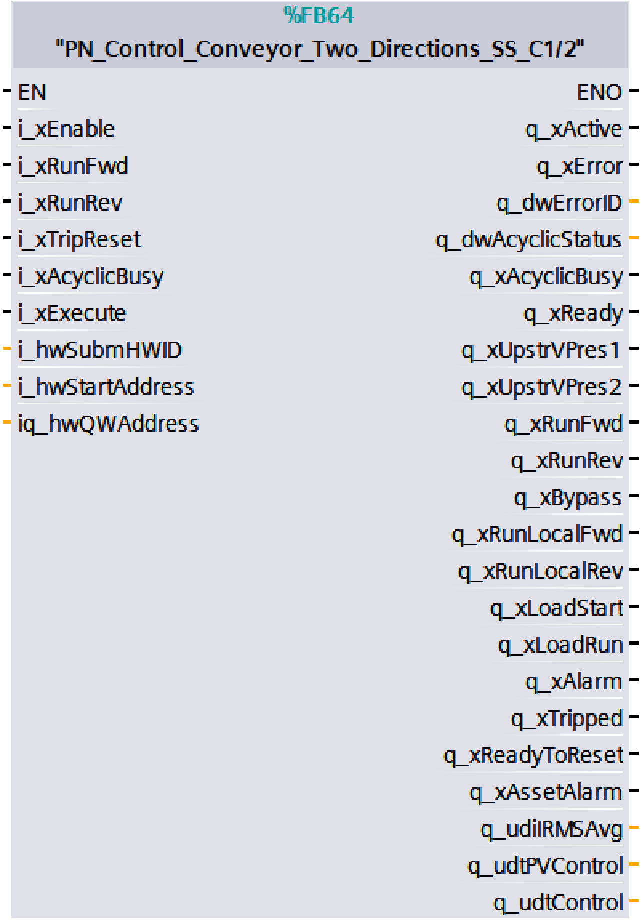
Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes

[72] Safety Integrity Level according to standard IEC 61508.

Status and error codes from ConstTeSysIsland (Continued)

Member name	Used by the function block
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

PN_Control_Conveyor_Two_Directions_SS_C1/2 (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunFwd	BOOL	If this input is set to TRUE, the avatar forward switch is closed.	Cyclic data
i_xRunRev	BOOL	If this input is TRUE, the avatar reverse switch is closed.	Cyclic data
i_xTripReset	BOOL	If this input is set to TRUE, detected trips whose trip reset conditions have been met will reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control

Inputs (Continued)

Input	Data Type	Description	Variable Category
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xReady	BOOL	Indicates TRUE if the function block is ready to receive an Execute command.	Cyclic data
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xRunFwd	BOOL	If this output is set to TRUE, the avatar forward switch is closed.	Cyclic data
q_xRunRev	BOOL	If this output is set to TRUE, the avatar reverse switch is closed.	Cyclic data
q_xRunLocalFwd	BOOL	If this output is set to TRUE, the avatar Local Forward command is set.	Cyclic data
q_xRunLocalRev	BOOL	If this output is set to TRUE, the avatar Local Reverse command is set.	Cyclic data
q_xBypass	BOOL	If this output is set to TRUE, the avatar Bypass command is set.	Cyclic data
q_xLoadStart	BOOL	If this output is set to TRUE, the motor is in start phase.	Cyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_xLoadRun	BOOL	If this output is set to TRUE, a Run or Close command has been executed, and current is flowing in the poles (equivalent to motor running but also for non-motor avatars).	Cyclic data
q_xAlarm	BOOL	If this output is set to TRUE, a protection alarm has been detected by the avatar.	Cyclic data
q_xTripped	BOOL	If this output is set to TRUE, a protection trip event has been detected by the avatar.	Cyclic data
q_xReadyToReset	BOOL	If this output is set to TRUE, the avatar meets the trip reset conditions and can be reset with a Trip Reset command.	Cyclic data
q_xAssetAlarm	BOOL	If this output is set to TRUE, a power device or SIL[73] Starter within the avatar has reached or exceeded 90% of the expected durability (per the avatar parameter).	Cyclic data
q_udiIRMSAvg	UDINT	Indicates the average of the most recent phase current Irms values. (Unit: A)	Cyclic data
q_udtPVControl	UDT_ PVControl	This is a structure of cyclic PV status data for Control avatars. For detailed information about the structure of this UDT, see UDT_PVControl, page 58.	Cyclic data
q_udtControl	UDT_Control	This is a structure of acyclic status data for Control avatars, which includes information about alarm and trip messages and maintenance information about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52. For an overview of which members of the UDT are supported by this avatar, see Supported Members of UDT_Control, page 293.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

[73] Safety Integrity Level according to standard IEC 61508.

Supported Members of UDT_Control

Not all members of the user defined type “UDT_Control” (see UDT_Control, page 52) are supported by all avatars. The following table gives an overview of which elements are available for this avatar. Unsupported values will always use the default value.

NOTE: Safety Integrity Level according to standard IEC 61508.

Elements

Element	Sub-Element	Supported
MotorTemperature	n/A	Yes
SILGroup	n/A	Yes
ThermalCapacity	n/A	Yes
AlarmMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
AlarmMsg2	OverCurrent	Yes
	MotorOverheat	Yes
TripMsg1	GroundCurrent	Yes
	ThermalOverload	Yes
	LongStart	Yes
	Jam	Yes
	PhaseUnbalance	Yes
	UnderCurrent	Yes
	Stall	Yes
TripMsg2	PhConfig	Yes
	OverCurrent	Yes
	PhaseLoss	Yes
	PhaseReversal	Yes
	MotorOverheat	Yes
TimeToTrip	n/A	Yes
TimeToReset	n/A	Yes
PAStatusReg1	PA0Status	Yes
	PA1Status	Yes
	PA2Status	Yes
	PA3Status	Yes
	PA4Status	Yes
	PA5Status	Yes
	PA6Status	Yes
	PA7Status	Yes
	PA8Status	Yes
	PA9Status	Yes

Generic Avatar Function Blocks

What's in This Chapter

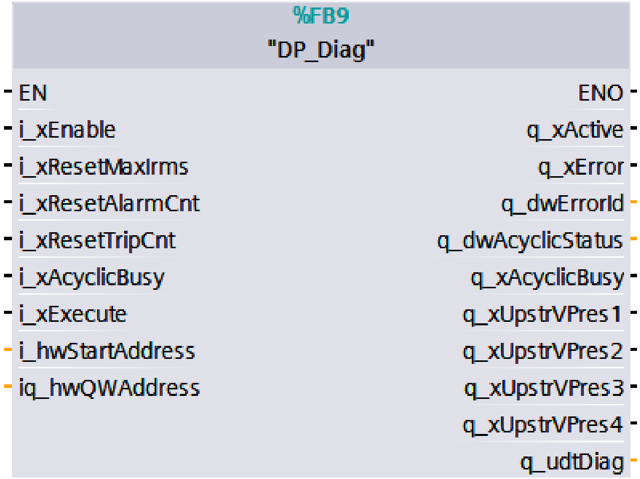
Avatar Diagnostic

Function block profile

Function block type	Avatar Generic function block.
Type of data access	Cyclic (read/write) and acyclic (read).
Intended use	All Load, Application and Device avatars. For the Device avatars “Analog I/O” and “Digital I/O” the data can be retrieved but will only contain default values as these avatars do not support this functionality.
Functionality	The Avatar Diagnostic function block returns the status of the diagnostic data of the specified avatar.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Diag (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xResetMaxIRMS	BOOL	If this input is set to TRUE, the maximum average Irms current value and the timestamp are reset.	Cyclic data
i_xResetAlarmCnt	BOOL	If this input is set to TRUE, all counters are reset for this avatar.	Cyclic data
i_xResetTripCnt	BOOL	If this input is set to TRUE, all trip counters are reset for this avatar.	Cyclic data

Inputs (Continued)

Input	Data Type	Description	Variable Category
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

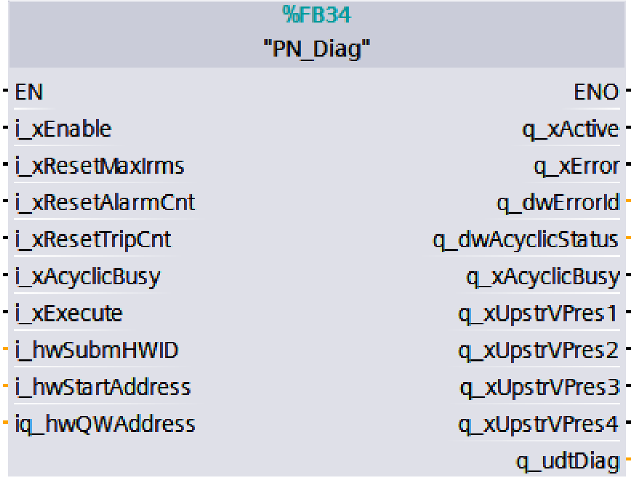
Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xUpstrVPres3	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the third starter/power device in this avatar.	Cyclic data
q_xUpstrVPres4	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the fourth starter/power device in this avatar.	Cyclic data
q_udtDiag	UDT_ Diagnostic	This is a structure of acyclic diagnostic data for Control avatars, which includes information about Irms values and trip record about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

PN_Diag (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xResetMaxIRMS	BOOL	If this input is set to TRUE, the maximum average Irms current value and the timestamp are reset.	Cyclic data
i_xResetAlarmCnt	BOOL	If this input is set to TRUE, all alarm counters are reset for this avatar.	Cyclic data
i_xResetTripCnt	BOOL	If this input is set to TRUE, all trip counters are reset for this avatar.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control

Inputs (Continued)

Input	Data Type	Description	Variable Category
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input submodule address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_xUpstrVPres1	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the first starter/power device in this avatar.	Cyclic data
q_xUpstrVPres2	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the second starter/power device in this avatar.	Cyclic data
q_xUpstrVPres3	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the third starter/power device in this avatar.	Cyclic data
q_xUpstrVPres4	BOOL	If this output is set to TRUE, the avatar has detected that upstream main power is present for the fourth starter/power device in this avatar.	Cyclic data
q_udtDiag	UDT_ Diagnostic	This is a structure of acyclic diagnostic data for Control avatars, which includes information about Irms values and trip record about the avatar. For detailed information about the structure of this UDT, see UDT_Control, page 52.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	Yes
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

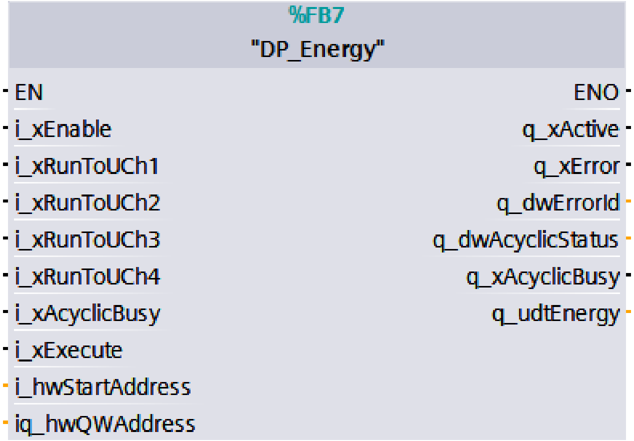
Avatar Energy Management

Function block profile

Function block type	Avatar Generic function block.
Type of data access	Cyclic (write) and acyclic (read).
Intended use	All Load, Application and Device avatars. For the Device avatars “Analog I/O” and “Digital I/O” the data can be retrieved but will only contain default values as these avatars do not support this functionality.
Functionality	The Avatar Energy function block returns the status of the energy data of the specified avatar.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this avatar.

DP_Energy (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunToUCh1	BOOL	If this input is set to TRUE, the Record Time of Use Channel 1 command is started.	Cyclic data
i_xRunToUCh2	BOOL	If this input is set to TRUE, the Record Time of Use Channel 2 command is started.	Cyclic data
i_xRunToUCh3	BOOL	If this input is set to TRUE, the Record Time of Use Channel 3 command is started.	Cyclic data
i_xRunToUCh4	BOOL	If this input is set to TRUE, the Record Time of Use Channel 4 command is started.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control

Inputs (Continued)

Input	Data Type	Description	Variable Category
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwStartAddress	Variant	Must correspond to the first address of the cyclic input address range of the related avatar to be able to execute cyclic read communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the output address of the related avatar to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_udtEnergy	UDT_Energy	This is a structure of acyclic energy data for Control avatars, which includes information about active and reactive energy of the avatar. For detailed information about the structure of this UDT, see UDT_Energy, page 57.	Acyclic data

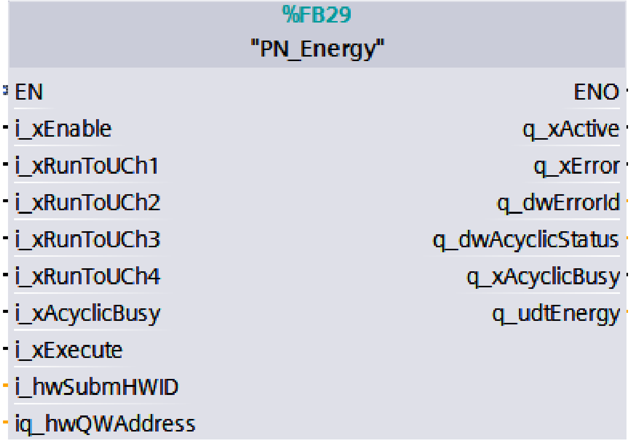
Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	Yes
Cdw_ErrInvalidHwid	No
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	No
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes

Status and error codes from ConstTeSysIsland (Continued)

Member name	Used by the function block
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

PN_Energy (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xRunToUCh1	BOOL	If this input is set to TRUE, the Record Time of Use Channel 1 command is started.	Cyclic data
i_xRunToUCh2	BOOL	If this input is set to TRUE, the Record Time of Use Channel 2 command is started.	Cyclic data
i_xRunToUCh3	BOOL	If this input is set to TRUE, the Record Time of Use Channel 3 command is started.	Cyclic data
i_xRunToUCh4	BOOL	If this input is set to TRUE, the Record Time of Use Channel 4 command is started.	Cyclic data
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the acyclic submodule of the related avatar to be able to execute the acyclic communication.	Communication parameter

In/Out

InOut	Data Type	Description	Variable Category
iq_hwQWAddress	WORD	Must correspond to the address of the cyclic output submodule to be able to execute cyclic write communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_udtEnergy	UDT_Energy	This is a structure of acyclic energy data for Control avatars, which includes information about active and reactive energy of the avatar. For detailed information about the structure of this UDT, see UDT_Energy, page 57.	Acyclic data

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	No
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	No
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

Device Asset Management Function Blocks

What's in This Chapter

All Device Asset Management function blocks share the same set of detected error codes as stated in the following table:

Status and error codes from ConstTeSysIsland

Member name	Used by the function block
Cdw_ErrNoMsg	Yes
Cdw_ErrInternal	Yes
Cdw_ErrInvalidCycInAddr	No
Cdw_ErrInvalidHwid	Yes
Cdw_ErrInvalidNumOfAvatars	No
Cdw_ErrReadCycIn	No
Cdw_ErrReadAcycData1	Yes
Cdw_ErrReadAcycData2	No
Cdw_ErrWriteAcycData	No
Cdw_StsReadAcycData1Compl	Yes
Cdw_StsReadAcycData2Compl	No
Cdw_StsWriteAcycDataCompl	No

SIL Interface Module, Asset Management

NOTE: Safety Integrity Level according to standard IEC 61508.

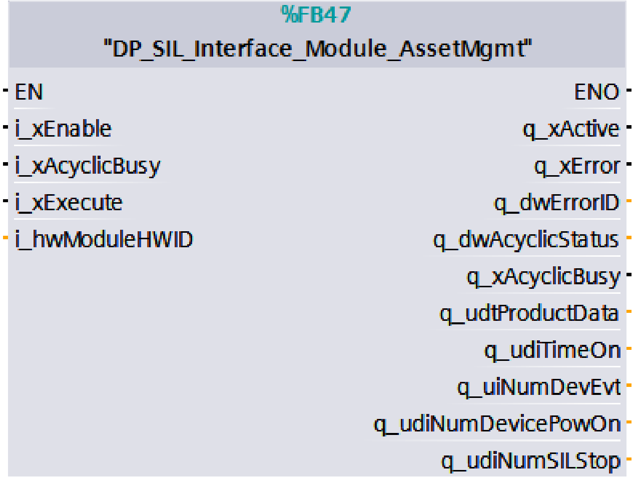
Function block profile

Function block type	Device function block.
Type of data access	Acyclic (read).
Intended use	SIL Interface Module devices.
Functionality	The SIL Interface Module, Asset Management function block returns the status of the acyclic asset management data of the specified device.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this device.

DP_SIL_Interface_Module_AssetMgmt (PROFIBUS DP)

NOTE: Safety Integrity Level according to standard IEC 61508.



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwModuleHWID	HW_IO	Must correspond to the hardware identifier of the of the related device to be able to execute the acyclic communication.	Communication parameter

Outputs

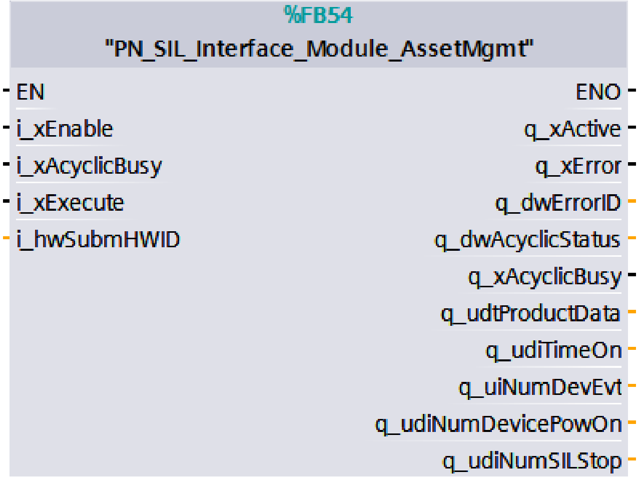
Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_udtProductData	UDT_ ProductData	Contains all relevant product data of the device.	Acyclic data
q_udiTimeOn	UDINT	The length of time that the device has been switched on.	Acyclic data

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_uiNumDevEvt	UINT	The number of detected device events.	Acyclic data
q_udiNumDevicePowOn	UDINT	The number of device power cycles.	Acyclic data
q_udiNumSILStop	UDINT	The number of SIL Stops of the device.	Acyclic data

PN_SIL_Interface_Module_AssetMgmt (PROFINET IO)

NOTE: Safety Integrity Level according to standard IEC 61508.



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the of the related device to be able to execute the acyclic communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus	Function block status

Outputs (Continued)

Output	Data Type	Description	Variable Category
		communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_udtProductData	UDT_ ProductData	Contains all relevant product data of the device.	Acyclic data
q_udiTimeOn	UDINT	The length of time that the device has been switched on.	Acyclic data
q_uiNumDevEvt	UINT	The number of detected device events.	Acyclic data
q_udiNumDevicePowOn	UDINT	The number of device power cycles.	Acyclic data
q_udiNumSILStop	UDINT	The number of SIL Stops of the device.	Acyclic data

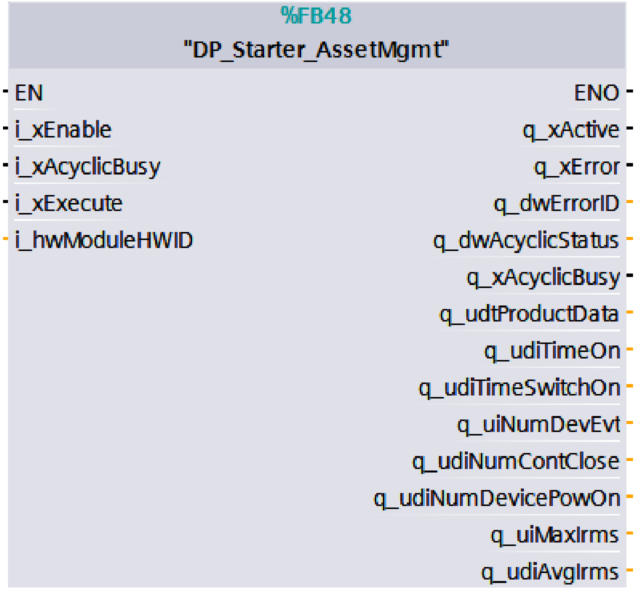
Starter, Asset Management

Function block profile

Function block type	Device function block.
Type of data access	Acyclic (read).
Intended use	Standard or SIL[74] Starter devices.
Functionality	The Starter, Asset Management function block returns the status of the acyclic asset management data of the specified device.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this device.

DP_Starter_AssetMgmt (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwModuleHWID	HW_IO	Must correspond to the hardware identifier of the of the related device to be able to execute the acyclic communication.	Communication parameter

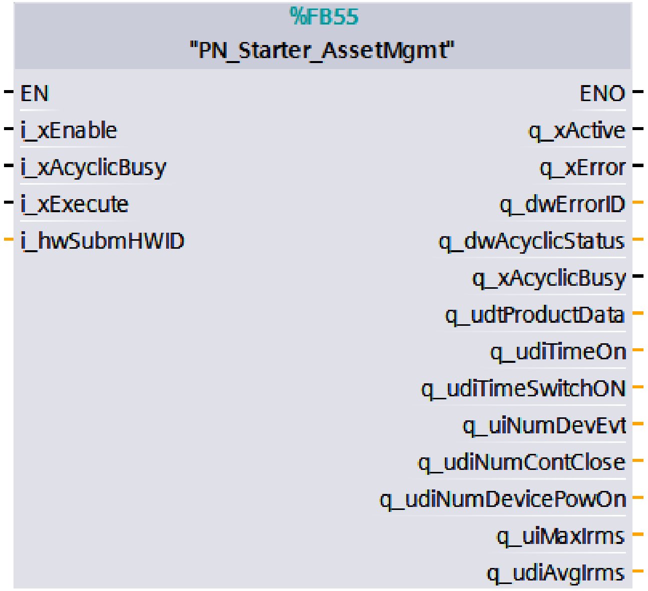
[74] Safety Integrity Level according to standard IEC 61508.

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_udtProductData	UDT_ ProductData	Contains all relevant product data of the device.	Acyclic data
q_udiTimeOn	UDINT	The length of time that the device has been switched on.	Acyclic data
q_udiTimeSwitchOn	UDINT	The length of time that the contactor has been switched on.	Acyclic data
q_uiNumDevEvt	UINT	The number of detected device events.	Acyclic data
q_udiNumContClose	UDINT	The number of closing cycles of the contactor.	Acyclic data
q_udiNumDevicePowOn	UDINT	The number of device power cycles.	Acyclic data
q_uiMaxIrms	UINT	The maximum measured current value during device lifetime.	Acyclic data
q_udiAvgIrms	UDINT	The calculated average current value.	Acyclic data

PN_Starter_AssetMgmt (PROFINET IO)

The following tables give information about the parameters of the function block interface.



Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the of the related device to be able to execute the acyclic communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_udtProductData	UDT_ ProductData	Contains all relevant product data of the device.	Acyclic data
q_udiTimeOn	UDINT	The length of time that the device has been switched on.	Acyclic data
q_udiTimeSwitchOn	UDINT	The length of time that the contactor has been switched on.	Acyclic data
q_uiNumDevEvt	UINT	The number of detected device events.	Acyclic data
q_udiNumContClose	UDINT	The number closing cycles of the contactor.	Acyclic data
q_udiNumDevicePowOn	UDINT	The number of device power cycles.	Acyclic data
q_uiMaxIrms	UINT	The maximum measured current value during device lifetime.	Acyclic data
q_udiAvgIrms	UDINT	The calculated average current value.	Acyclic data

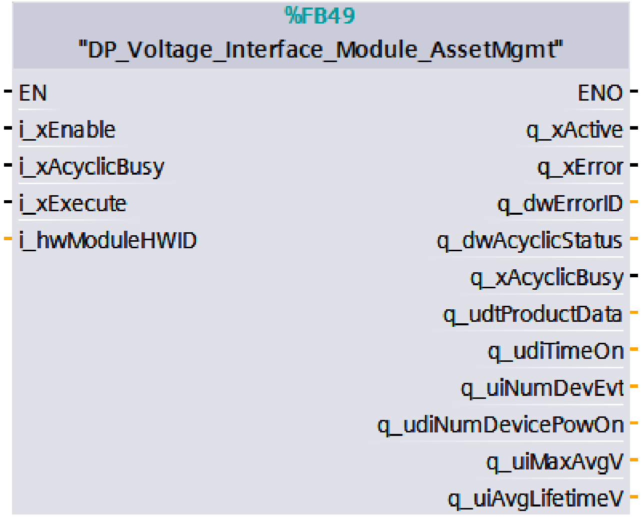
Voltage Interface Module, Asset Management

Function block profile

Function block type	Device function block.
Type of data access	Acyclic (read).
Intended use	Voltage Interface Module devices.
Functionality	The Voltage Interface Module, Asset Management function block returns the status of the acyclic asset management data of the specified device.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this device.

DP_Voltage_Interface_Module_AssetMgmt (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwModuleHWID	HW_IO	Must correspond to the hardware identifier of the of the related device to be able to execute the acyclic communication.	Communication parameter

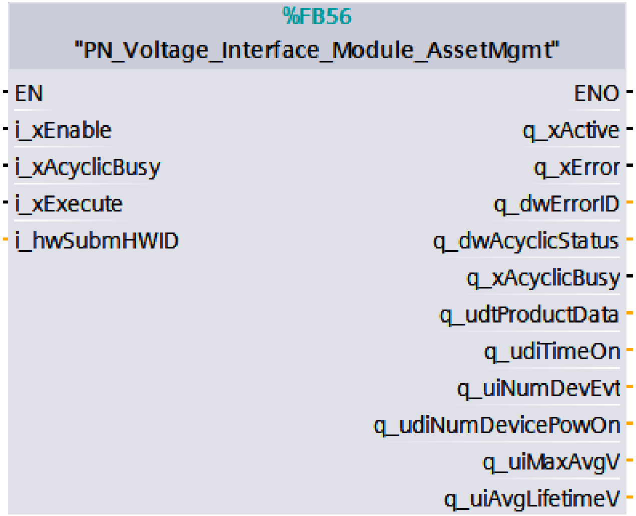
Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information	Function block status

Outputs (Continued)

Output	Data Type	Description	Variable Category
		about the detected error, see the value of the Error ID.	
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_udtProductData	UDT_ ProductData	Contains all relevant product data of the device.	Acyclic data
q_udiTimeOn	UDINT	The length of time that the device has been switched on.	Acyclic data
q_uiNumDevEvt	UINT	The number of detected device events.	Acyclic data
q_udiNumDevicePowOn	UDINT	The number of device power cycles.	Acyclic data
q_uiMaxAvgV	UINT	The maximum average voltage value during device lifetime.	Acyclic data
q_uiAvgLifetimeV	UINT	The calculated average voltage value, since last reset.	Acyclic data

PN_Voltage_Interface_Module_AssetMgmt (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control

Inputs (Continued)

Input	Data Type	Description	Variable Category
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the of the related device to be able to execute the acyclic communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_udtProductData	UDT_ ProductData	Contains all relevant product data of the device.	Acyclic data
q_udiTimeOn	UDINT	The length of time that the device has been switched on.	Acyclic data
q_uiNumDevEvt	UINT	The number of detected device events.	Acyclic data
q_udiNumDevicePowOn	UDINT	The number of device power cycles.	Acyclic data
q_uiMaxAvgV	UINT	The maximum average voltage value during device lifetime.	Acyclic data
q_uiAvgLifetimeV	UINT	The calculated average voltage value, since last reset.	Acyclic data

I/O Module, Asset Management

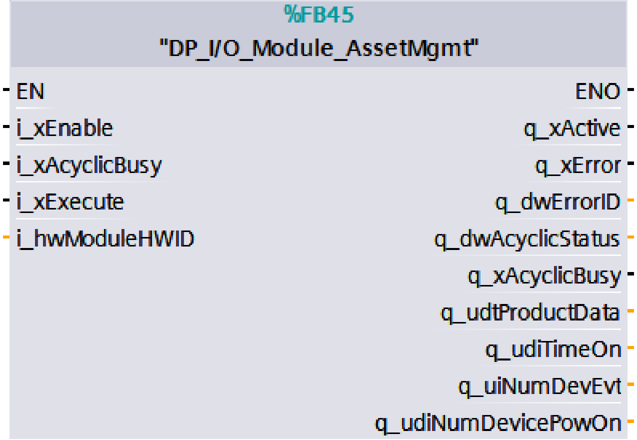
Function block profile

Function block type	Device function block.
Type of data access	Acyclic (read).
Intended use	I/O Module devices.
Functionality	The I/O Module, Asset Management function block returns the status of the acyclic asset management data of the specified device.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this device.

DP_I/O_Module_AssetMgmt (PROFIBUS DP)

The following tables give information about the parameters of the function block interface.



Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwModuleHWID	HW_IO	Must correspond to the hardware identifier of the of the related device to be able to execute the acyclic communication.	Communication parameter

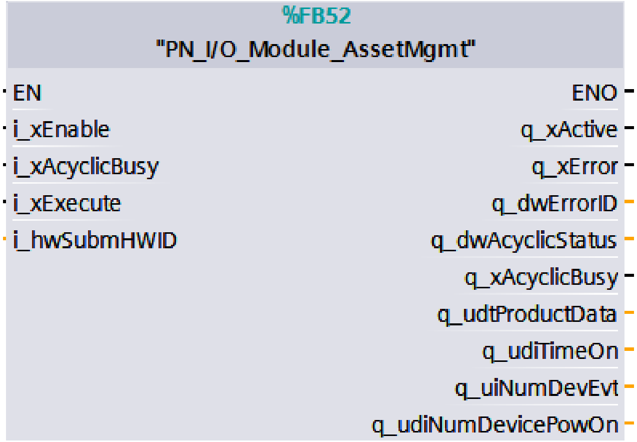
Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status

Outputs (Continued)

Output	Data Type	Description	Variable Category
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_udtProductData	UDT_ ProductData	Contains all relevant product data of the device.	Acyclic data
q_udiTimeOn	UDINT	The length of time that the device has been switched on.	Acyclic data
q_uiNumDevEvt	UINT	The number of detected device events.	Acyclic data
q_udiNumDevicePowOn	UDINT	The number of device power cycles.	Acyclic data

PN_I/O_Module_AssetMgmt (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the of the related device to be able to execute the acyclic communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_udtProductData	UDT_ ProductData	Contains all relevant product data of the device.	Acyclic data
q_udiTimeOn	UDINT	The length of time that the device has been switched on.	Acyclic data
q_uiNumDevEvt	UINT	The number of detected device events.	Acyclic data
q_udiNumDevicePowOn	UDINT	The number of device power cycles.	Acyclic data

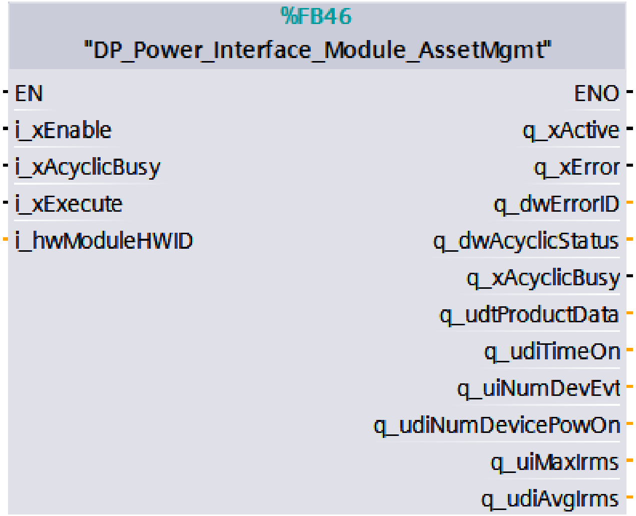
Power Interface Module, Asset Management

Function block profile

Function block type	Device function block.
Type of data access	Acyclic (read).
Intended use	Power Interface Module device.
Functionality	The Power Interface Module, Asset Management function block returns the status of the acyclic asset management data of the specified device.

The following sections provide detailed information about the interface and the use of the parameters of the PROFINET IO and PROFIBUS DP function blocks for this device.

DP_Power_Interface_Module_AssetMgmt (PROFIBUS DP)



The following tables give information about the parameters of the function block interface.

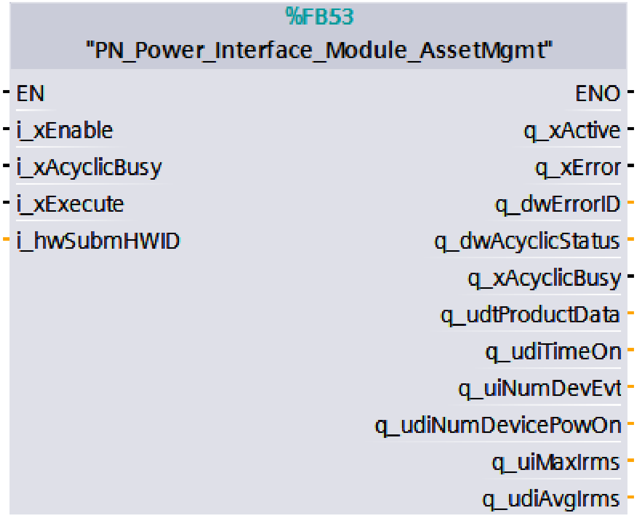
Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwModuleHWID	HW_IO	Must correspond to the hardware identifier of the of the related device to be able to execute the acyclic communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_udtProductData	UDT_ ProductData	Contains all relevant product data of the device.	Acyclic data
q_udiTimeOn	UDINT	The length of time that the device has been switched on.	Acyclic data
q_uiNumDevEvt	UINT	The number of detected device events.	Acyclic data
q_udiNumDevicePowOn	UDINT	The number of device power cycles.	Acyclic data
q_uiMaxIrms	UINT	The maximum measured current value during device lifetime.	Acyclic data
q_udiAvgIrms	UDINT	The calculated average current value.	Acyclic data

PN_Power_Interface_Module_AssetMgmt (PROFINET IO)



The following tables give information about the parameters of the function block interface.

Inputs

Input	Data Type	Description	Variable Category
i_xEnable	BOOL	Activates the function block processing. Cyclic data will be read/written to the hardware IO interface.	Function block control
i_xAcyclicBusy	BOOL	Can be used to block the acyclic execution if acyclic communication of another function block is active.	Function block control
i_xExecute	BOOL	Starts an acyclic request on a rising edge.	Function block control
i_hwSubmHWID	HW_IO	Must correspond to the hardware identifier of the of the related device to be able to execute the acyclic communication.	Communication parameter

Outputs

Output	Data Type	Description	Variable Category
q_xActive	BOOL	If this output is set to TRUE, the Avatar function block is enabled, or an acyclic request is busy.	Function block status
q_xError	BOOL	If this output is set to TRUE, a detected error occurred while executing the Avatar function block. For detailed information about the detected error, see the value of the Error ID.	Function block status
q_dwErrorID	DWORD	The Error ID shows detailed information about the detected error that occurred. Possible values are shown in the table below.	Function block status
q_dwAcyclicStatus	DWORD	The Acyclic status provides detailed information about the acyclic fieldbus communication, including detected error details. For detailed information about the status values, see Function Block Library, page 49.	Function block status
q_xAcyclicBusy	BOOL	If this output is set to TRUE, an acyclic request is still busy.	Function block status
q_udtProductData	UDT_ ProductData	Contains all relevant product data of the device.	Acyclic data
q_udiTimeOn	UDINT	The length of time that the device has been switched on.	Acyclic data
q_uiNumDevEvt	UINT	The number of detected device events.	Acyclic data
q_udiNumDevicePowOn	UDINT	The number of device power cycles.	Acyclic data
q_uiMaxIrms	UINT	The maximum measured current value during device lifetime.	Acyclic data
q_udiAvgIrms	UDINT	The calculated average current value.	Acyclic data

Appendix

What’s in This Part

Frequently Asked Questions (FAQs)

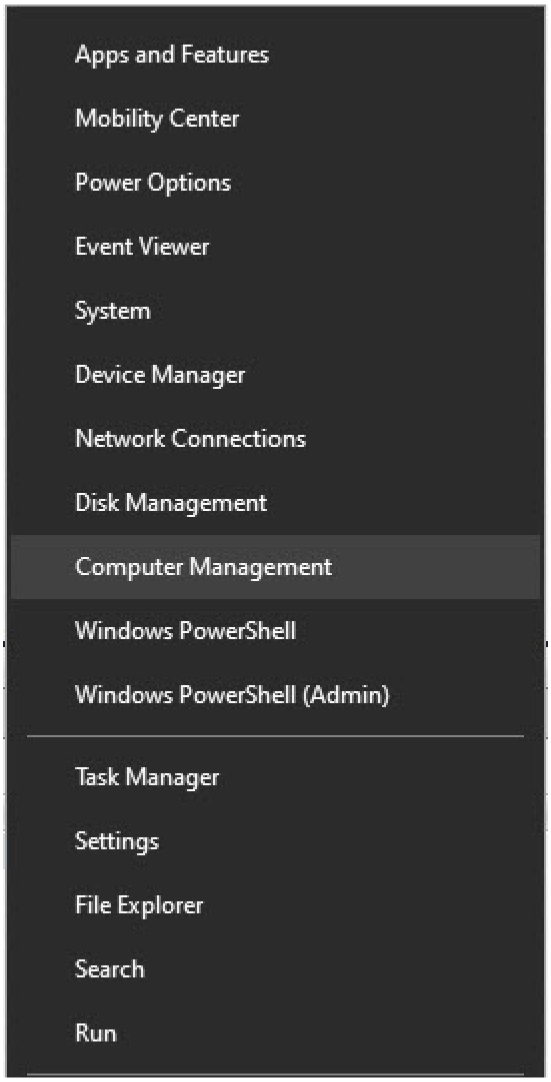
What's in This Chapter

How to Add a User to Siemens® TIA Openness

If you receive an error when trying to import the CAx file that you need to be a user of the Siemens® TIA Openness group, do the following:

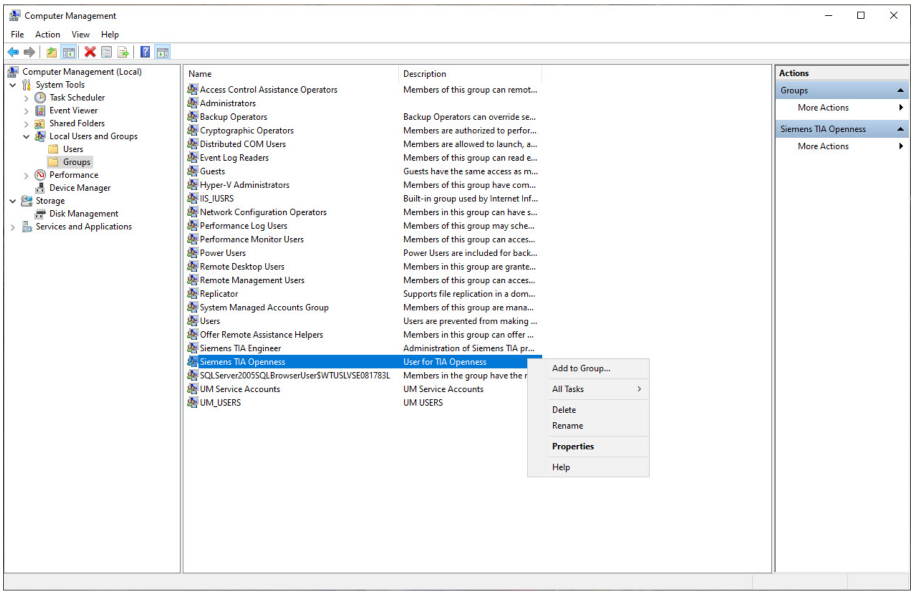
1. Right click on the Windows **Start**, then select **Computer Management.**.

Windows Start



2. In the Computer Management window, click **Local Users and Groups → Groups**. Right click on **Siemens TIA Openness**, then select **Add to Group...**.

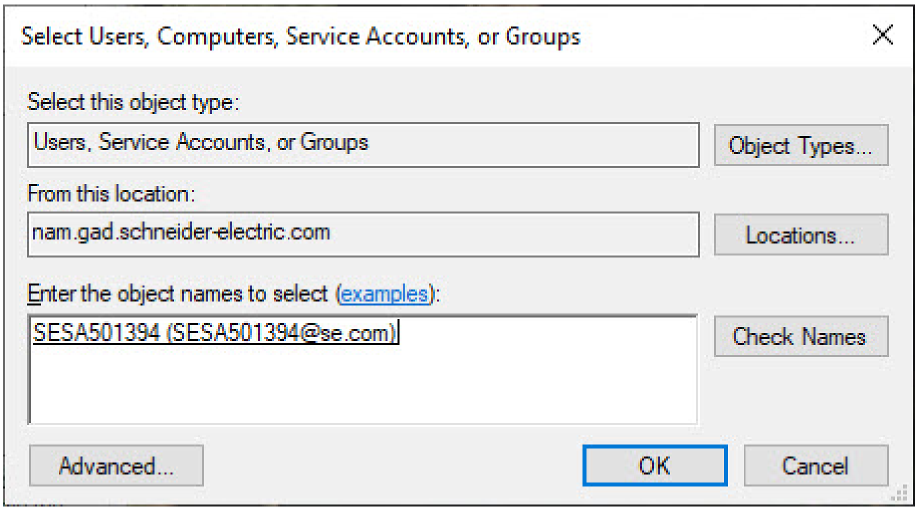
Computer Management



Name	Description
Access Control Assistance Operators	Members of this group can remot...
Administrators	
Backup Operators	Backup Operators can override se...
Cryptographic Operators	Members are authorized to perfor...
Distributed COM Users	Members are allowed to launch, a...
Event Log Readers	Members of this group can read e...
Guests	Guests have the same access as m...
Hyper-V Administrators	Members of this group have com...
IIS_IUSRS	Built-in group used by Internet Inf...
Network Configuration Operators	Members in this group can have s...
Performance Log Users	Members of this group may sche...
Performance Monitor Users	Members of this group can acces...
Power Users	Power Users are included for back...
Remote Desktop Users	Members in this group are grante...
Remote Management Users	Members of this group can acces...
Replicator	Supports file replication in a dom...
System Managed Accounts Group	Members of this group are mana...
Users	Users are prevented from making ...
Offer Remote Assistance Helpers	Members in this group can offer ...
Siemens TIA Engineer	Administration of Siemens TIA pr...
Siemens TIA Openness	User for TIA Openness
SQLServer2005SQLBrowserUser$WTUSLVSE081783L	Members in the group have the r
UM Service Accounts	UM Service Accounts
UM_USERS	UM USERS

3. In the Siemens TIA Openness Properties window, click **Add....** Type the object name to select (username) in the **Enter the object name to select** field, then click **Check Names** to locate the user name.

Add User



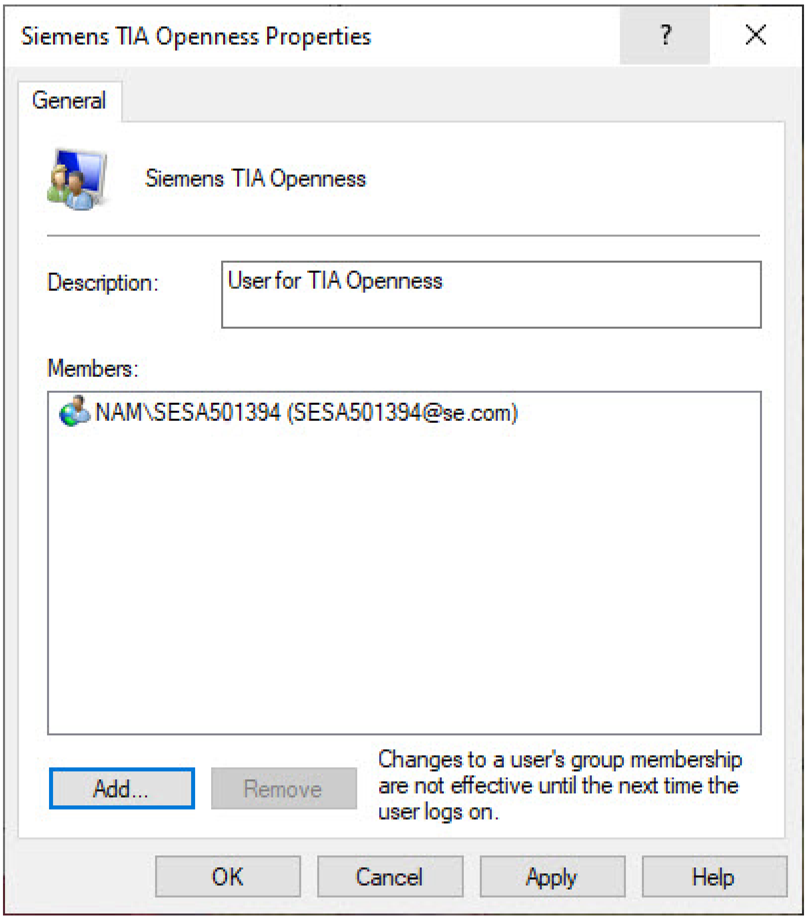
Select this object type:
Users, Service Accounts, or Groups

From this location:
nam.gad.schneider-electric.com

Enter the object names to select (examples):
SESA501394 (SESA501394@se.com)

4. After the user is found, click OK.

Siemens TIA Openness Properties



Siemens TIA Openness Properties

Siemens TIA Openness

Description: User for TIA Openness

Members:

NAM\SESA501394 (SESA501394@se.com)

Changes to a user's group membership are not effective until the next time the user logs on.

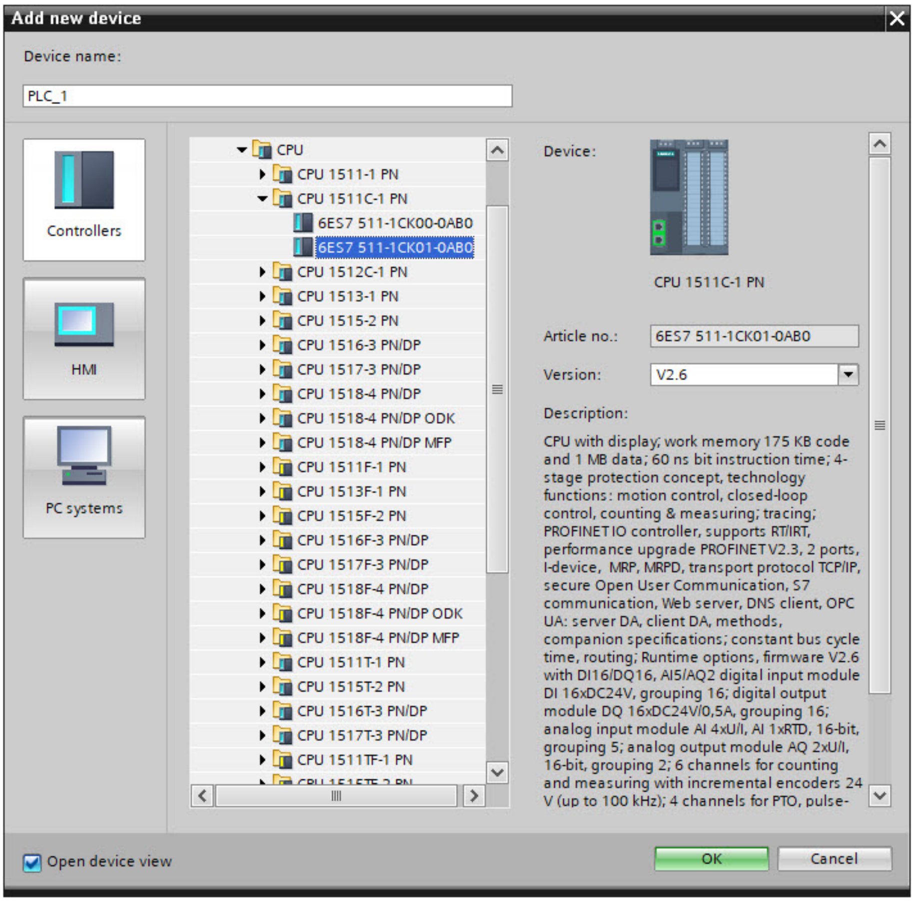
5. The user is added as a member for Siemens TIA Openness. Click **Apply**, then click **OK**.
6. Close the TIA Portal software application.
7. Log off your computer and then log back onto the computer.
8. Reopen your project, then import the CAx file.

How to Add a New PLC

To add a new PLC to your project, follow these steps:

1. Open an existing project.
2. Click **Add new device**. Click **Controllers** and navigate to your PLC, then select **OK**.

Add New Device



Add new device

Device name: PLC_1

Device: CPU 1511C-1 PN

Article no.: 6ES7 511-1CK01-0AB0

Version: V2.6

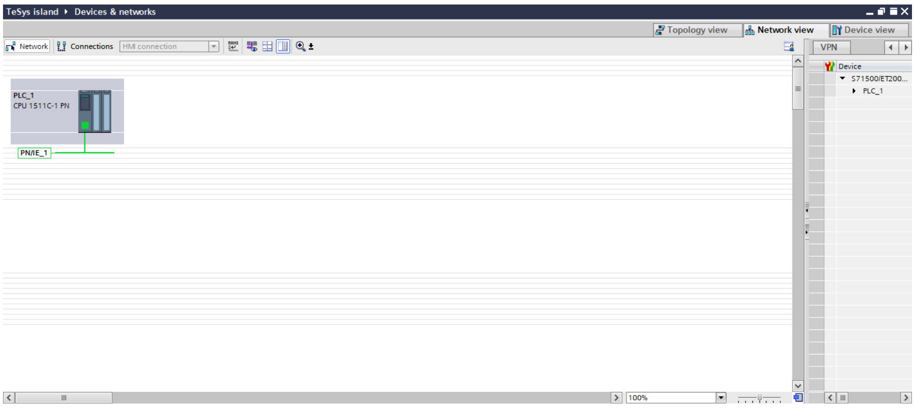
Description:

CPU with display; work memory 175 KB code and 1 MB data; 60 ns bit instruction time; 4-stage protection concept, technology functions: motion control, closed-loop control, counting & measuring; tracing; PROFINET IO controller, supports RT/IRT, performance upgrade PROFINET V2.3, 2 ports, I-device, MRP, MRPD, transport protocol TCP/IP, secure Open User Communication, S7 communication, Web server, DNS client, OPC UA: server DA, client DA, methods, companion specifications; constant bus cycle time, routing; Runtime options, firmware V2.6 with DI16/DQ16, AI5/AQ2 digital input module DI 16xDC24V, grouping 16; digital output module DQ 16xDC24V/0,5A, grouping 16; analog input module AI 4xU/I, AI 1xRTD, 16-bit, grouping 5; analog output module AQ 2xU/I, 16-bit, grouping 2; 6 channels for counting and measuring with incremental encoders 24 V (up to 100 kHz); 4 channels for PTO, pulse-

☑ Open device view

3. The new PLC appears in the Topology view. Select the **Network View** to view and assign the TeSys™ island device.

Network View



Schneider Electric
35 rue Joseph Monier
92500 Rueil Malmaison
France

+ 33 (0) 1 41 29 70 00

www.se.com

As standards, specifications, and design change from time to time, please ask for confirmation of the information given in this publication.

DOCA0272EN-00