JENAER ANTRIEBSTECHNIK

PROFINET®

PROFIBUS®

PROFIdrive

ECOVARIO® 114D / ECOVARIO® 616(D)

AN33: Control via PROFINET
with PROFIdrive Profile
with annex „PROFIBUS with PROFIdrive Profile“

Published Editions:

Edition	Comment
Feb. 2015	First English Edition
August 2016	new LOGO
June 2017	Chapter 5.2 Required encoder settings -> Absolute value encoder modified
February 2018	Chapter 6.8 Modification of the telegram types Chapter 7.3 Added: Base mode parameter access Chapter 8 Added: Error handling
March 2018	Chapter 6.6 Object description and units (linear) added. Graphics for the representation of AC1 inserted. Chapter 6.7.2 Units (linear) added
January 2019	Chapter 4.3 Overview table: Standard telegram 5 added
March 2019	Chapter 7.3 Table standard data types added
January 2020	Annex „PROFIdrive via PROFIBUS“ added

Jenaer Antriebstechnik GmbH
Buchaer Straße 1
07745 Jena
Germany

Contents

1 About this Documentation

This manual describes the procedure for functional integration of servo amplifiers series ECOVARIO® 114D, ECOVARIO 616(D) with PROFIdrive profile into a PROFINET network (Master: Siemens SIMATIC PLC S7-1500) via the Siemens-TIA-Portal (Totally Integrated Automation). Prerequisite is that a physical connection exists between the servo amplifier and the PROFINET master via the interface X22 (ECOVARIO® 114 DR-[I,H]x-xxx-xxx, ECOVARIO® 616 xR-[I,H]x-xxx-xxx).

JAT servo amplifiers with PROFIdrive drive profile can also be integrated into a PROFIBUS-DP network. However, some particularities should be noted, as described in Annex A.

Further information:

- Hardware installation: Installation and Operating Instructions ECOVARIO®114 D, ECOVARIO®616(D)
- Software commissioning and parameter setting: ECO Studio Online Help or Operation Manual
- Documentation of the program package „TIA-Portal“ and of the PLC S7-1500 (Siemens AG)
- Appendix of this application note: „PROFIdrive Parameterverzeichnis“

This manual makes the following demands on qualified personnel:
Transport: Personnel trained in handling electrostatic sensitive devices
Installation: Electrotechnically qualified personnel who know the security directives of electrical engineering and automation
Setup/Commissioning: Qualified personnel with a broad knowlege of the fields of electrical engineering, automation and drives, especially of PLC programming.
For functional integration of the servo amplifiers into a PROFINET network knowledge of the Siemens program package „TIA-Portal“ is necessary.

When handling drive systems the manuals listed above and the safety precautions contained therein have to be observed.

The listed programs are example programs and show the basic procedure for integration of the drives into a PROFINET environment. In customer specific applications the user has to check whether all function and safety relevant conditions are fulfilled.

2 About PROFINET

The field bus system PROFINET is standardized by PROFIBUS/PROFINET Internation (PI). By using switches all Ethernet tree topologies are possible.

The maximum cable length between two nodes is 100 m for Ethernet based field bus systems. The transmission rate is 100 MBit/s.

The PROFINET communication is implemented according to a master/slave principle. For PROFINET IRT (**I**sochrounous **R**eal**T**ime) there is a fix bus cycle for the actualisation between master and slave, similar to PROFIBUS DP.

3 Preconditions

Required Components

For controlling the JAT drives via PROFINET the following components are required (in case of deviations please contact us):

- Servo amplifier ECOVARIO 114 D or ECOVARIO 616(D) with PROFINET interface (Option „IJ“)
- Siemens SIMATIC PLC S7-1500
- Program package „Totally Integrated Automation Portal“ (TIA Portal), Version V12/V13 (Siemens AG)
- Service PC with commissioning tool ECO Studio V 2.11.0.0 or higher
- The latest GSDML file: (e.g. from our homepage www.jat-gmbh.de/download/ecosoftware/progs.html)

In order to establish the PROFINET cabling between the devices please refer to the ECOVARIO 114 D or ECOVARIO 616(D) installation manual and the SIMATIC S7-1500 documentation.

4 Procedure of the integration into the TIA Portal

Before the integration of the drive can be done in the TIA Portal, the drive must be fully configured via ECO Studio for its purpose.
Proceed according to the ECOSTUDIO online help or the ECOSTUDIO Operation Manual.
After hardware commissioning has been finished the further procedure is:

Basic Procedure

1. Start the TIA Portal program package
2. Open an existing project or create a new project, respectively.
3. Map the ECOVARIO 114 D or ECOVARIO 616(D) as a third-party drive by means of its .GSDML file (cf. chapter 4.1)
4. Establish the PROFINET bus connection (cf. chapter 4.2)
5. Establish the communication (cf. chapter 4.3)
6. Establish the isochronous clock (cf. chapter 4.4)
7. By means of the appropriate technology objects you can map your application in the TIA Portal (cf. chapter 5).

Background information

In **chapter 6** you find background information about the PROFIdrive profile and about communication:

- Drive Objects (DO)
- State Machine
- Control word (STW1)
- Status word (ZSW1)
- PROFIdrive basic parameters (PNU)
- Velocity mode
- Positioning mode (PSI, MDI)
- Standard telegrams

Acyclic Reading and Writing

The procedure described in **chapter 7** for the acyclic reading of parameters from the drive and the acyclic writing of parameters into the drive is applied if outside of the technology objects specific parameters have to be written from master to drive or read from it.

4.1 Map the ECOVARIO 114 D / ECOVARIO 616 (D) by means of its GSDML file

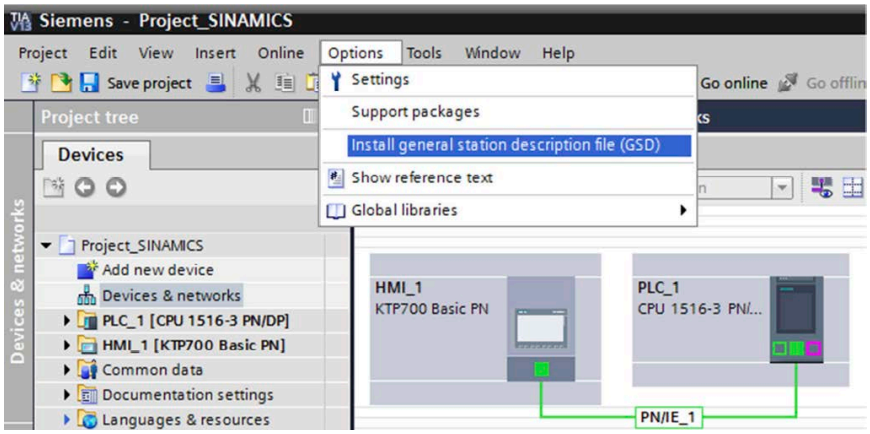
Note: In order to use the drive together with the SIMATIC S7-1500, the configuration in the drive via the commissioning software ECO Studio and the configuration via the Totally Integrated Automation Portal (TIA-Potal) must match.

The device master file (.GSDML) of the required firmware version of the drive can be imported into the Totally Integrated Automation Portal (TIA-Portal) by means of the menu item „***Options*** > ***Install general station description file (GSD)***“.

Select GSDML File

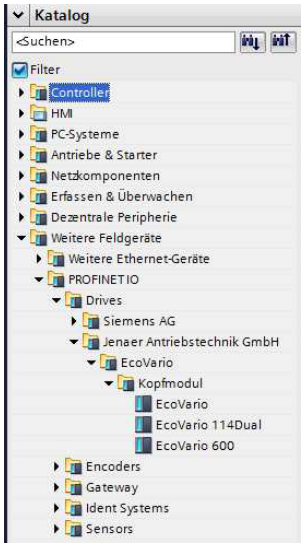
As source select the folder where you have stored the GSDML file including the picture file for the device representation.

GSDML-V2.x-JAT-EcoVario-2014xxxx.XML	GSDML file for servo amplifier ECOVARIO114 D / ECOVARIO 616(D).
GSDML-FFFF-FFFF-EcoVario114D	Corresponding picture file



View in the Hardware Catalog

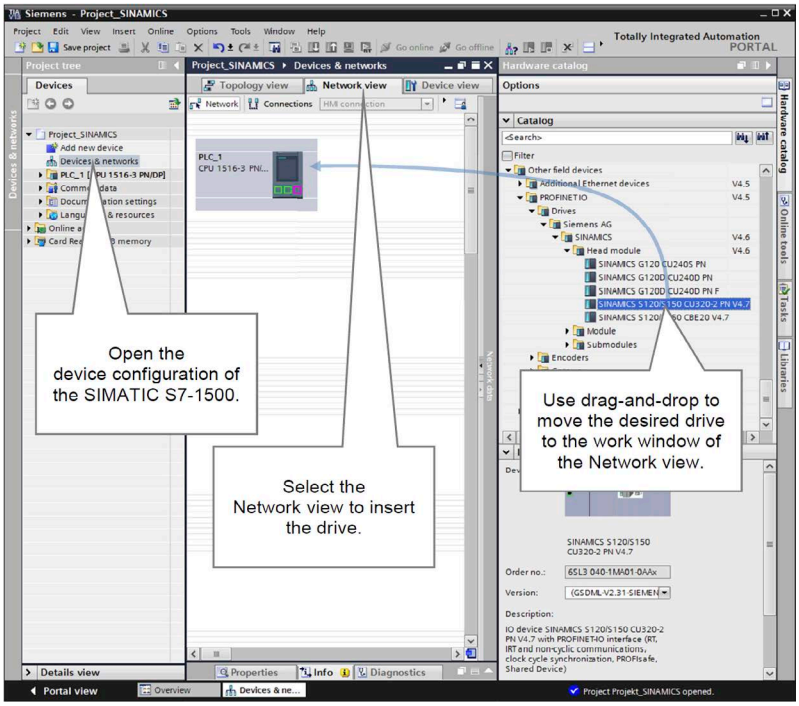
The ECOVARIO 114 D / ECOVARO 616(D) is now listed in the Hardware Catalog in the folder ***Other field devices***.



4.2 Establishing the PROFINET bus connection

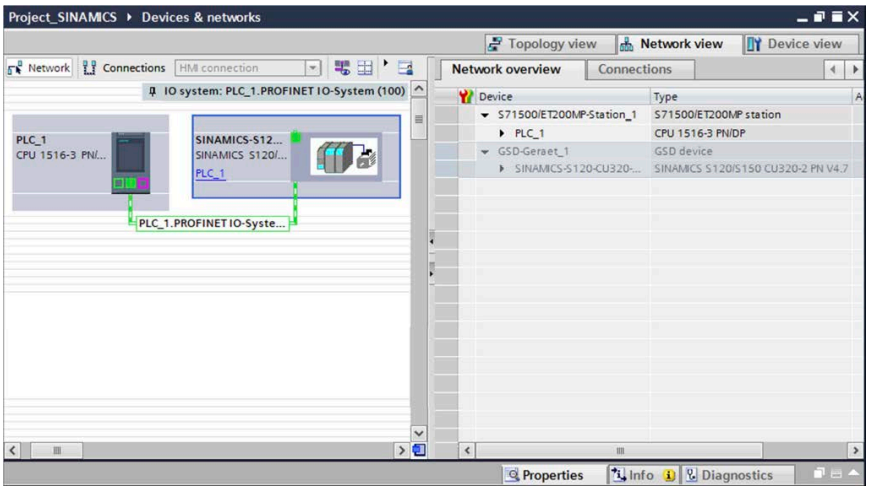
Dragging Ecovario into the working space

Proceed as follows:



Establishing the bus connection

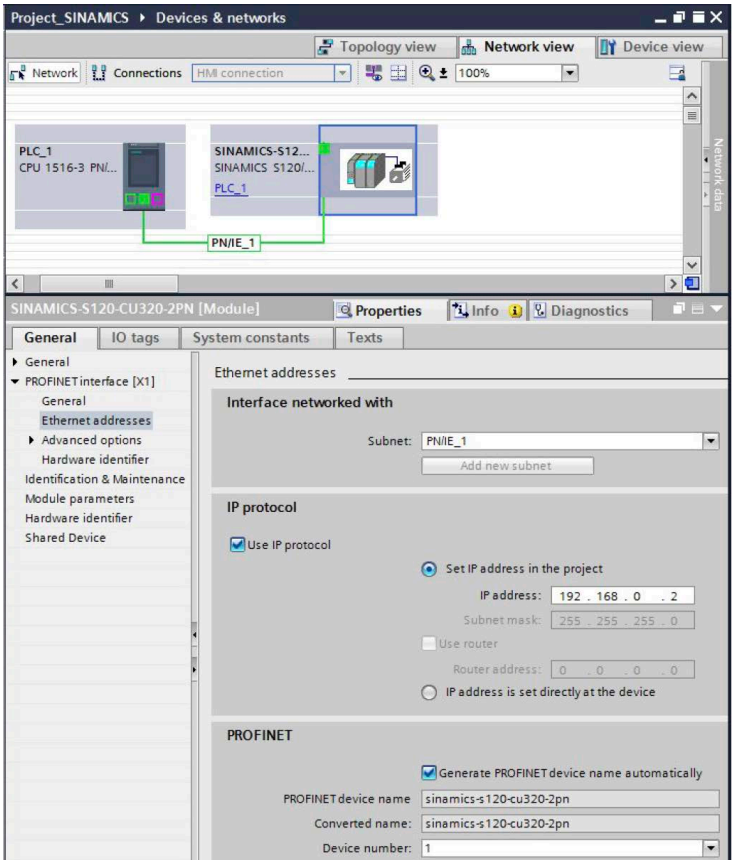
To establish the connection between the SIMATIC S7-1500 and the drive via the PROFINET fieldbus, click on the PROFINET port of the drive in the TIA Portal and connect this port to the PROFINET port of the SIMATIC S7-1500 while keeping the left mouse button pressed. When doing so, make sure that you are in the “Network” function mode in the “Network view”.



Note: Initially, the bus connection defines the communication partner of the SIMATIC S7-1500 or the drive. At this point, the data to be exchanged between the SIMATIC S7-1500 and the drive is not yet configured. The bus connection only creates the option to exchange data and defines the physical interface.

Network View

Finally, to completely establish the connection, you have to assign the IP address of the drive or – if this has not already been done – the IP address of the SIMATIC S7-1500. To do so, click on the image of the module in the ***Network view***. In the ***Properties*** workspace, ***General*** tab, you can now use the ***Ethernet addresses*** menu option to set the address of the SIMATIC S7-1500 and, as shown in the figure, the IP address of the drive.
Here you can also specify or automatically generate the PROFINET device name and, if necessary, change the device number.



4.3 Establishing the communication connection

Telegram selection:
Data exchange between the SIMATIC S7-1500 and the drive takes place using so called telegrams via which the data to be exchanged is defined according to the PROFIdrive standard.
ECOVARIO 114 D and ECOVARIO 616(D) support the following telegrams:

Application Class	Meaning	Standard telegram
1	Standard drive with velocity control	1 and 2
3	Drive with 1-axis positioning control	7 and 9
4	Servo drive with clock synchronous velocity and position control with central or decentral motion control	3 or 5

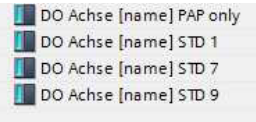
Details and background informationen can be found in chapter 6.

Selection of the Drive Objects (DO)

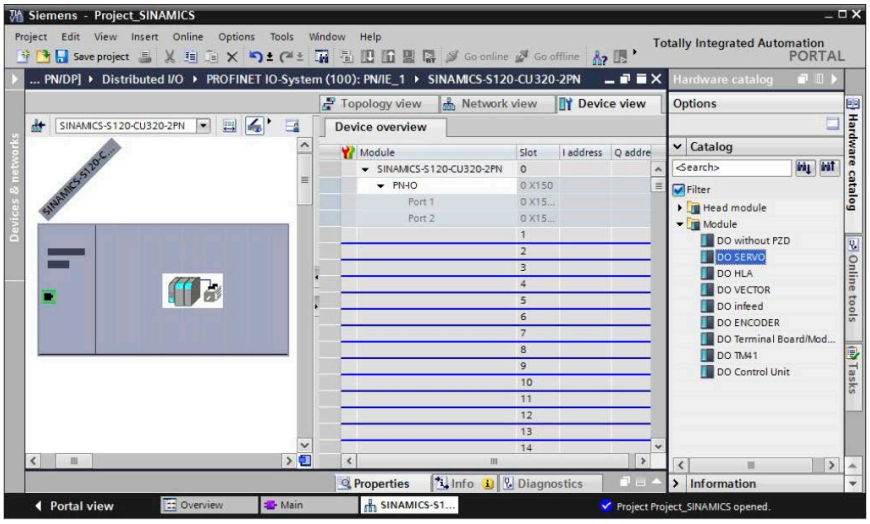
In the Totally Integrated Automation Portal (TIA-Portal) double-click on the desired drive to go to the ***Device view***. If the ***Filter*** function is activated in the hardware catalog, the PROFIdrive telegrams available for this drive are automatically displayed.

Before you can insert the desired PROFIdrive telegrams into the device overview, you have to create a drive object (DO) for each axis of the drive.

Therefore select from the Hardware Catalog the respective Drive Object (DO) which you want to address in the selected drive:



Please note that, without filter (= “Filter” not checked), the respective drive objects (DOs) may be listed multiple times in the hardware catalog. To determine the correct drive object (DO) with switched-off filter, successively click on all drive objects (DOs) listed in the hardware catalog. As soon as the rows for inserting the drive object (DO) are marked in blue, you have determined the correct object and can apply it to the selected rows in the device overview using drag-and-drop.



Once you have created the axes of the drive unit as drive objects (DOs), you can define the desired submodules (telegrams) below the DOs in the hardware catalog. To do so, select the required telegrams and use drag-and-drop to move them to the device overview.

This example shows the integration of telegrams for an ECOVARIO 114 D with two axes where both axes contain a standard telegram 9.

4.4 Setting Isochronous Mode [Source: Siemens AG]

If the axes connected to the SIMATIC S7-1500 are to be used as positioning axes via the position control of the technology object of the SIMATIC S7-1500 or if the DSC (Dynamic Servo Control) function, which moves the position control to the drive for quicker compensation of disturbances, is to be used to increase axis accuracy, the drive must be connected to the automation system via an isochronous bus system.

4.4.1 Basics and definition

IRT Ethernet - Isochronous Realtime Ethernet

IRT is a transmission mode where PROFINET devices are synchronized with extreme accuracy.
A sync master provides the clock, sync slaves synchronize with this clock. Both an IO controller and an IO device can act as a sync master.
Sync master and sync slaves are always nodes of a sync domain. Within the sync domain, bandwidth is reserved for IRT communication. Real-time and non-real-time communication (TCP/IP communication) is possible outside the reserved bandwidth.

Sync Domain

A sync domain is necessary to synchronize PROFINET IO devices. The sync domain ensures that all nodes of this domain can communicate in isochronous mode.
The prerequisite for IRT communication is a synchronization cycle for all PROFINET devices in a sync domain for distributing a common time base. This basic synchronization allows a synchronism of the transmission cycle of the PROFINET devices within a sync domain. The sync master (generally an IO controller) generates the common synchronization clock and specifies the time base with which all other sync slaves (e.g., IO devices) synchronize.
If the sync master fails, the communication of the IRT devices falls back to RT quality.

Setting up PROFINET with IRT

When setting up and operating a PROFINET IO system in IRT mode, please follow the rules below. The aim of these rules is to ensure optimum operation of your PROFINET IO system.

- When using IRT, you must configure the topology. This ensures the exact-calculation of update time, bandwidth and other parameters.
- If you want to use multiple sync domains, configure a sync domain boundary for the port connected to a PROFINET device of another sync domain.
- In a Sync domain In a sync domain, you can only configure one sync master at a time.
- A PROFINET IO system may only belong to a single sync domain.
- If you configure PROFINET devices in a sync domain and want to synchronize them with IRT, the relevant PROFINET devices must support IRT communication.
- Where possible, use the same PROFINET device as the PROFINET IO controller and sync master.
- If only some of the PROFINET devices of a PROFINET IO system are synchronized, place PROFINET devices that are not participating in IRT communication outside the sync domain.

Isochronous Mode

The "Isochronous mode" system property allows acquisition of measured values and process data in a fixed system cycle. Within the same system cycle, the signal is processed until it is available at the output terminal. Therefore, isochronous mode contributes to high control system quality, which results in greater manufacturing accuracy. Isochronous mode drastically reduces possible fluctuations in process reaction times. This processing stable in terms of time can be used for higher machine cycles. Basically, isochronous mode is the choice where measured values need to be acquired synchronously, motions need to be coordinated and process reactions need to be defined and take place simultaneously.

Time Sequence of Isochronous Processing

The following section explains the basic time sequence of all components involved in the synchronization from reading in the input data to outputting the output data:

- Reading in the input data in isochronous mode
- Transporting the input data to the IO controller (CPU) via the PROFINET subnet
- Processing the data in the isochronous application of the CPU
- Transporting the output data to the outputting IO device via the PROFINET subnet
- Outputting the output data in isochronous mode

Bias Time Ti

To ensure that a consistent status of the inputs can be transferred to the IO controller at the start time of a new system cycle, the read action must be moved up by the time Ti. For a specific input module, the time Ti includes at least the signal conditioning and conversion time on the electronic modules and the time for transfer to the interface module on the IO device backplane bus.
In a plant, the values are read in simultaneously because the bias time Ti of all input modules read in isochronous mode is set to the same value and this value is greater than or equal to the longest minimum bias time Ti of all isochronous input modules. With the default setting, STEP 7 ensures that a common bias time Ti is set that is as short as possible.

Delay Time To

To ensure that a consistent status of the outputs can be transferred to the process at the start time of a new system cycle, the output at the terminal does not take place before the time To after the clock beat. For a specific output module, the time To includes at least the transfer time from the IO controller to the IO device (via PROFINET IO) and in the IO device, the transfer of the outputs from the interface module to the electronic module (backplane bus) with the time for digital-to-analog conversion possibly included in this module.
In the plant, these values are written simultaneously because the delay time To of all isochronous output modules is set to the same value. This value must be greater than or equal to the longest minimum delay time To of all isochronous output modules. STEP 7 automatically calculates a common delay time To that is as short as possible.

When establishing isochronous communication via PROFINET, please note the following:

- Isochronous mode is only possible with the interfaces integrated in the CPU. For the SIMATIC S7-1500 with two PROFINET interfaces, only interface X1 can be configured for PROFINET IO and isochronous mode. Isochronous mode is not possible with CPs.
- Full isochronous mode from “terminal” to “terminal” is only possible if all components involved in the chain support the “Isochronous mode” system property. When selecting in the catalog, look for the “Isochronous mode” or “Isochronous processing” item in the information box of the module.
- If you are operating an IO device in isochronous mode (i.e. you have assigned it, for example, the sync slave role), the IO device must include at least one module or submodule operated in isochronous mode.

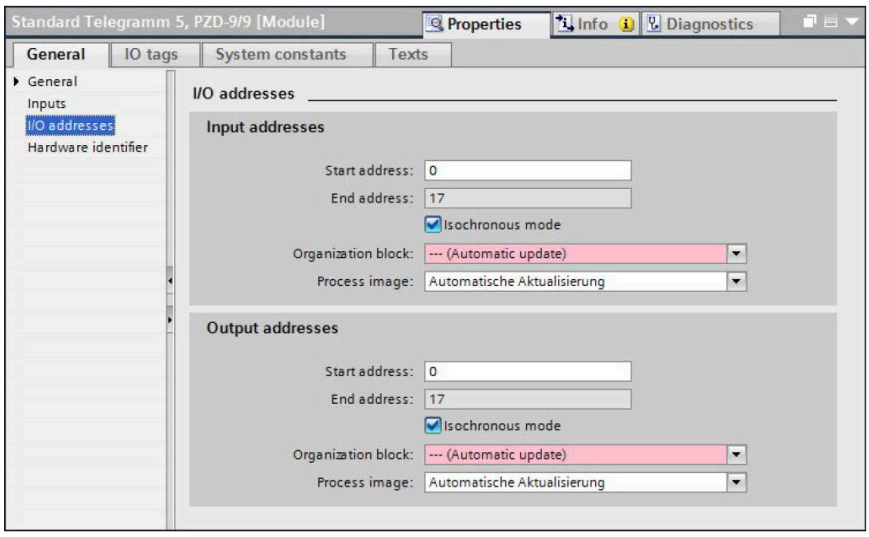
Note: When using PROFINET, the configured PROFINET name of the drive in the Totally Integrated Automation Portal (TIA Portal) must match the configured name in the commissioning software of the drive (ECOSTUDIO). Please note that the name is case sensitive.
If the PROFINET name in the Totally Integrated Automation Portal (TIA Portal) differs from the one in the drive, it may not be possible to establish a connection between the SIMATIC S7-1500 and the drive.

4.4.2 Settings in the TIA Portal

To activate isochronous mode, both the parameters for isochronous mode and the sync domain must be configured in the TIA Portal.

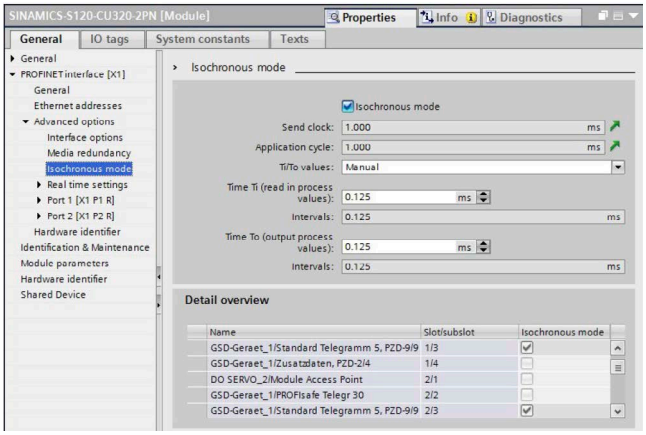
Isochronous Mode

In the device overview of the ECOVARIO on the telegram of the desired axis, select ***Properties > I/O addresses*** and activate the ***Isochronous mode*** function for the input and output addresses of the axis.



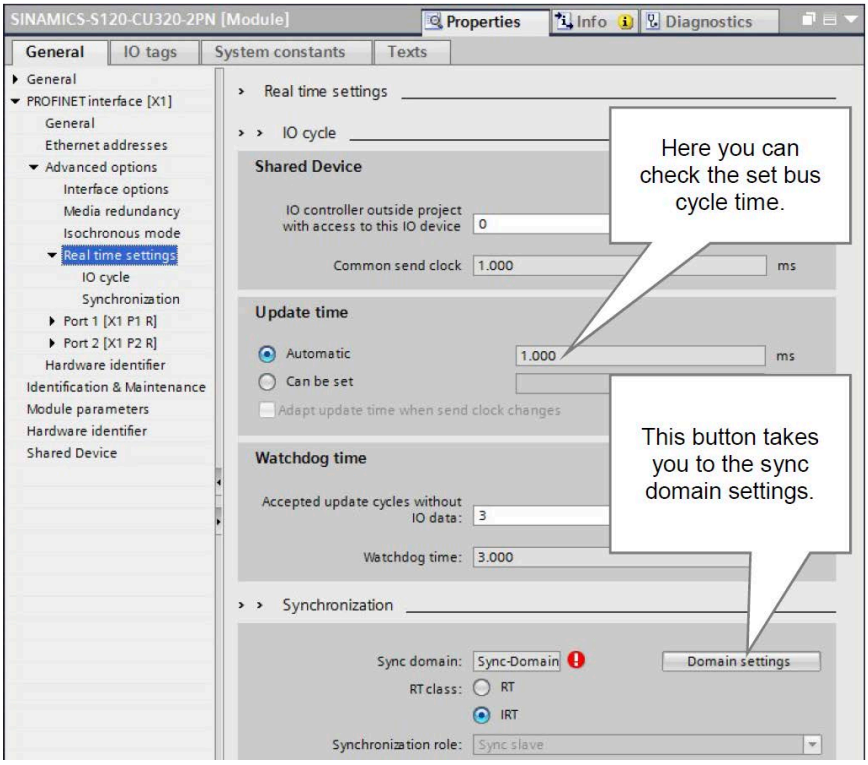
At this point, it is not yet necessary to assign the organization block for isochronous mode. This is done automatically when assigning the axis to a technology object in the SIMATIC S7-1500.

On the ECOVARIO under ***Advanced options*** > ***Isochronous mode*** set the „***Isochrounous mode***“ check box. For the other parameters retain the setting ***Automatic (Update)***



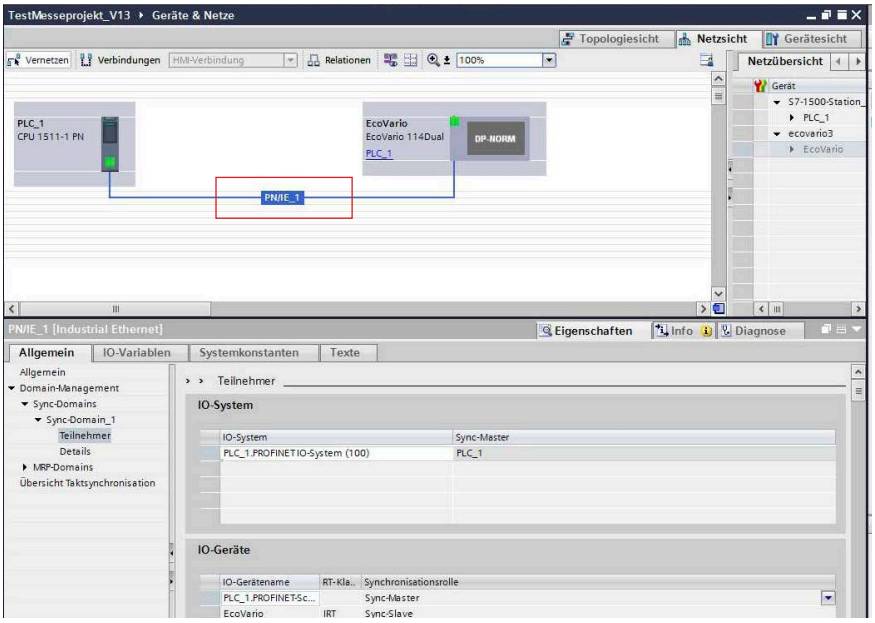
Sync Domain

To go to the sync domain settings, select ***Properties*** > ***Real time settings*** > ***Synchronization***. The ***Domain settings*** button takes you directly to the properties of the PROFINET line.



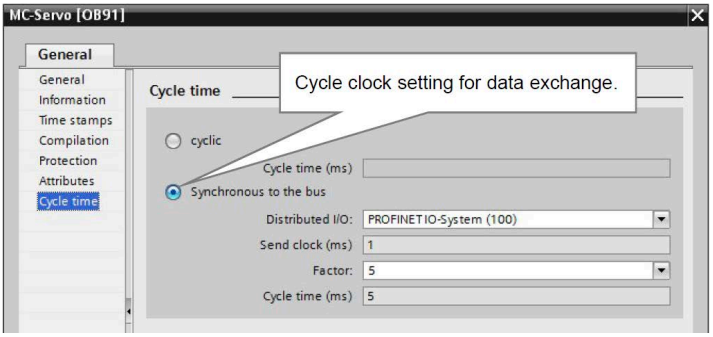
After selecting the bus in the network view you can assign the sync domain nodes the respective roles:

- SIMATIC S7-1500 (via PROFINET) = „Sync Master“.
- ECOVARIO drive system = „Sync Slave“ with RT class = „IRT“



SETTING ON THE TECHNOLOGY OBJECT

If a technology object is created after setting isochronous mode – as referred to in this documentation –, technology OB 91 "MC_Servo" will be created. In the properties of this organization block, the data exchange synchronization must be set in "Cycle time". For isochronous data exchange via PROFINET I/O, select the "Synchronous to the bus" setting.



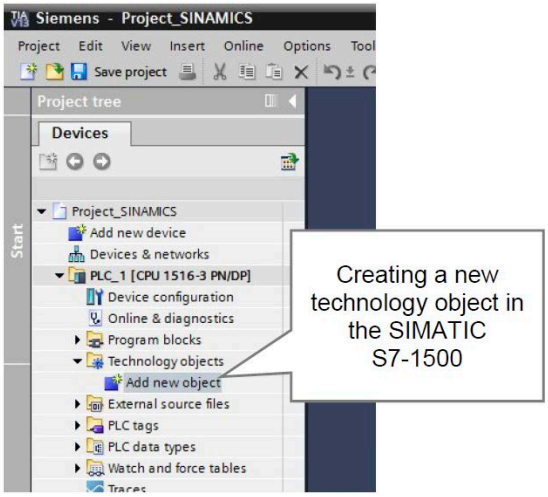
CHECKING THE ISOCHRONOUS MODE SETTINGS ON THE DRIVE

Finally, you can check the settings you have made via the PROFINET line in the ***Overview isochronous mode***. Once a technology object has been created, the organization block of the technology object is automatically entered in the detail overview in ***Application with isochronous mode***.

5 Motion Control Functions of the S7-1500

5.1 Technology objects

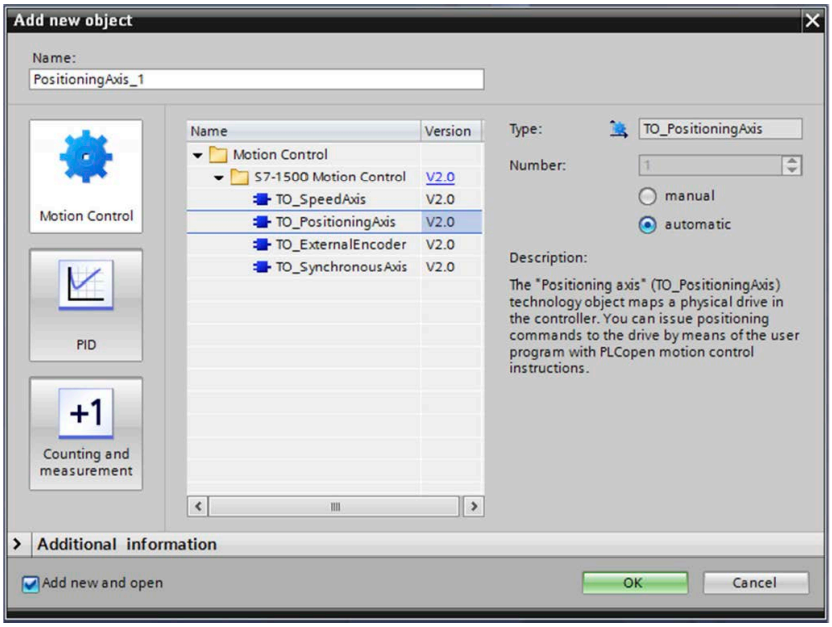
After the axes of the drive unit have been configured and connected to the SIMATIC S7-1500, a new technology object (TO) must now be created in the SIMATIC S7-1500 for each axis to allow easy axis control.



For this purpose, the Totally Integrated Automation Portal (TIA Portal) provides a dialog where different technology objects can be selected in the “Motion Control” area.

For motion control applications, the following technology objects are available:

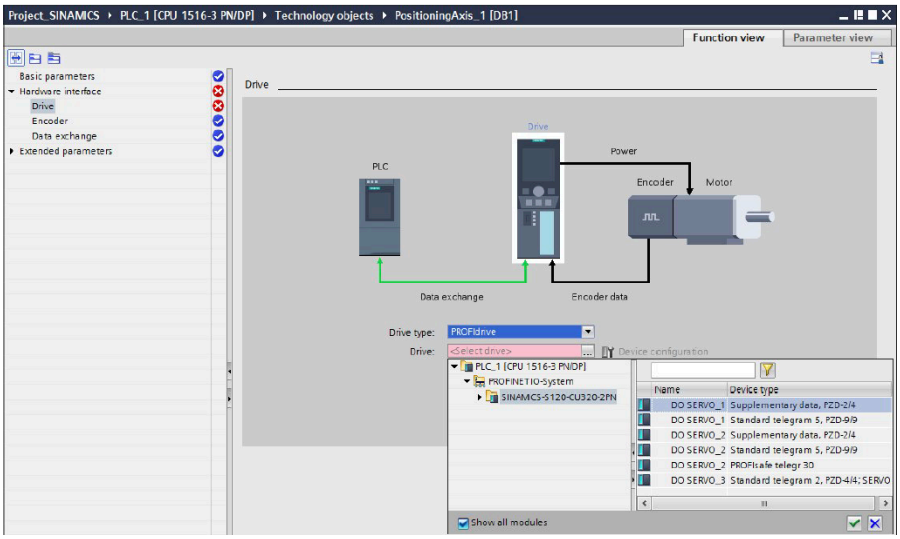
- Axes
  - Speed axes: TO_SpeedAxis
    The “SpeedAxis” technology object is used to select the speed for a drive.
    The axis motion can be controlled via motion control instructions.
  - Positioning axes: TO_PositionAxis
    The “PosAxis” axis technology object is used for position-controlled positioning of a drive.
    Motion control instructions allow you to give the axis positioning jobs via the user program.
- Other technology objects: TO_ExternalEncoder
  - External encoders: TO_ExternalEncoder
    The “ExternalEncoder” technology object detects a position and provides it to the controller.
    The determined position can be evaluated in the user program.



5.2 Selecting the drive on the technology object

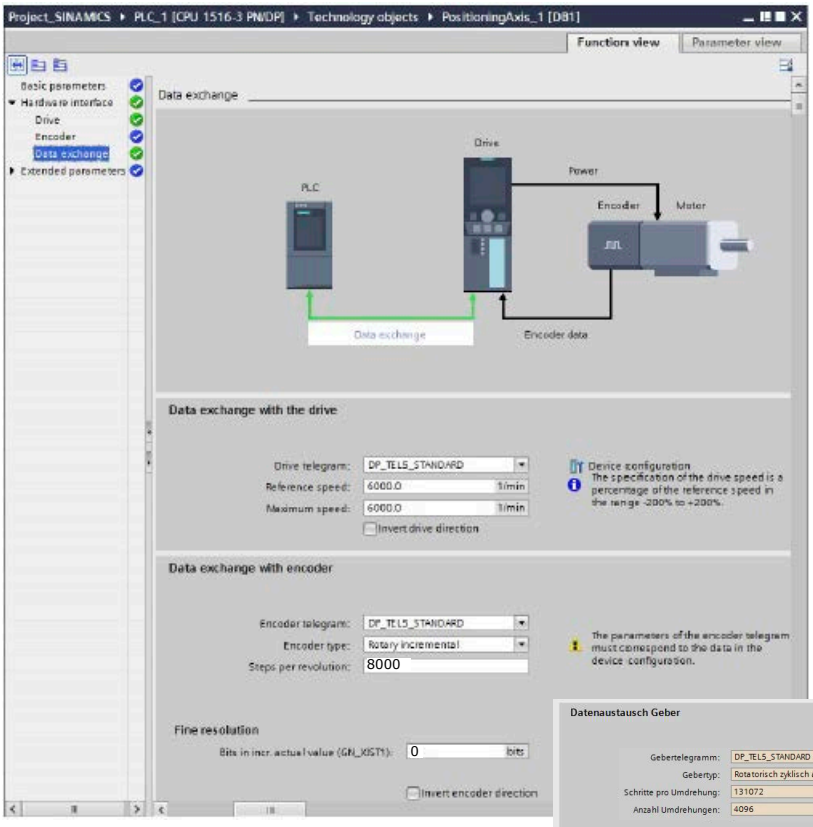
After adding the desired technology object, the appropriate drive unit axis must be connected to the technology object.

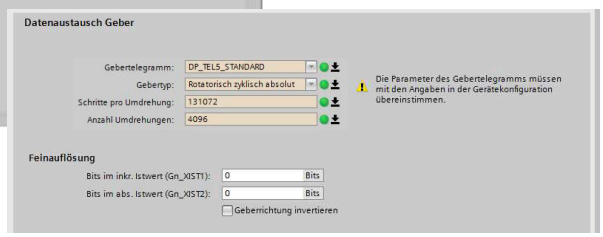
In the technology object, select ***Hardware interface*** > ***Drive***. In the "Drive" selection box, select the appropriate drive object (DO) for PROFINET or the appropriate telegram of the desired axis for PROFIBUS.



Note: If the desired drive object (DO) for PROFINET or the desired telegram of the axis for PROFIBUS has been configured but is not displayed in the selection dialog, activate the ***Show all modules*** function in the selection dialog and try again.

Now the desired axis is connected to the technology object of the SIMATIC S7-1500.





Settings in the ***Data exchange with encoder*** group box for motors of Jenaer Antriebstechnik:
- Incremental encoder 8000 inc/rev (example above)
- Absolute value encoder 17 bit/rev and 12 bit revolutions (example on the right)

Note: If the data is exchanged with the selected axis via PROFIdrive telegram 5, the DSC (Dynamic Servo Control) function to move the position control to the drive is automatically enabled when creating the Positioning Axis technology object.
If you do not want to use DSC (Dynamic Servo Control), explicitly change this function in the technology object in Extended Parameters > Control loop from "Position control and speed control in drive (DSC enabled)" to "Position control and speed control in the PLC". Axis position control is then only performed by the technology object.

5.3 Controlling the axis from the user program

To control the axes created as technology objects, the SIMATIC S7-1500 provides multiple technology functions that allow the user to influence the technology objects quickly and easily.
To influence the technology objects from the user program, the following functions are available:

Table 5-1: Technology functions

Technology function	Can be used for the following technology objects	Description
MC_Power	Positioning axis Speed axis External encoder	"MC_Power" is used to enable or disable a technology object.
MC_Reset	Positioning axis Speed axis External encoder	"MC_Reset" is used to acknowledge all technology alarms that can be acknowledged in the user program. The acknowledgement also resets the "Error" and "Warning" bits in the technology data block. The "***Restart***" input of "MC_Reset" is used to start the reinitialization of technology objects. When restarting the technology object, new configuration data is applied to the technology data block.
MC_Home	Positioning axis External encoder	"MC_Home" is used to create the reference between the position on the technology object and the mechanical position of the associated axis. The actual position value on the technology object is assigned to a reference mark. This reference mark represents a known mechanical position. Homing is performed according to the mode selected on the "***Mode***" parameter and the configuration in "***Technology object > Configuration > Extended parameters > Homing***".
MC_Halt	Positioning axis Speed axis	"MC_Halt" is used to decelerate an axis to a standstill. The "***Jerk***" and "***Deceleration***" parameters are used to define the dynamic response for deceleration.
MC_MoveJog	Positioning axis Speed axis	"MC_MoveJog" is used to move an axis in jog mode. The "***Velocity***", "***Jerk***", "***Acceleration***" and "***Deceleration***" parameters are used to define the dynamic response for the motion. • Positioning axis: A velocity is selected on the "***Velocity***" parameter. • Speed axis: A speed is selected on the "***Velocity***" parameter.
MC_MoveVelocity	Positioning axis Speed axis	"MC_MoveVelocity" is used to move an axis at a constant velocity. The "***Velocity***", "***Jerk***", "***Acceleration***" and "***Deceleration***" parameters are used to define the dynamic response for the motion. • Positioning axis: A velocity is selected on the "***Velocity***" parameter. • Speed axis: A speed is selected on the "***Velocity***" parameter.
MC_MoveRelative	Positioning axis	"MC_MoveRelative" is used to move an axis relative to the position of the start of job processing. The "***Velocity***", "***Jerk***", "***Acceleration***" and "***Deceleration***" parameters are used to define the dynamic response for the motion.
MC_MoveAbsolute	Positioning axis	"MC_MoveAbsolute" is used to move an axis to an absolute position. The "***Velocity***", "***Jerk***", "***Acceleration***" and "***Deceleration***" parameters are used to define the dynamic response for the motion.

Note: For detailed information on the use of the technology functions, please refer to the documentation for the motion control functions of the SIMATIC S7-1500 or the Totally Integrated Automation Portal (TIA Portal) documentation.

6 PROFIdrive via PROFINET and PROFIBUS

6.1 Supported telegrams

The servo amplifiers ECOVARIO 114D and ECOVARIO 616(D) support the following application classes and standard telegrams in PROFIdrive profile via PROFINET and PROFIBUS:

Tabelle 6.2: Supported application classes and standard telegrams

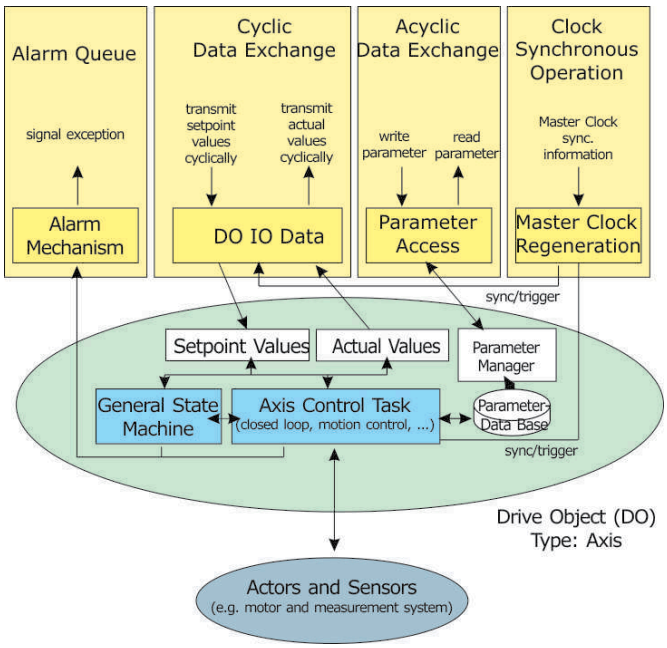
Application class	Description	Standard telegrams	PROFINET	PROFIBUS
1	Standard drive with velocity control	1 and 2	x	x
3	Drive with 1-axis position control	7 and 9	x	x
4	Servo drive with clock synchronous velocity and position control with central or decentral motion control	3 or 5	x	-

6.1 Drive Objects (DO)

The PROFIdrive profile defines as the main element the Drive Objects (DO) that control the motion parameters.

The Drive Object (DO) consists of:
- Common State Machine
- Axis Control Task
- Parameter manager with parameter data base.

The read/write access to the DO via PROFINET can be done by:
- Cyclic data exchange
- Acyclic data exchange (cf. chapter 7)
- Alarm Queue
- Iscochronous operation.



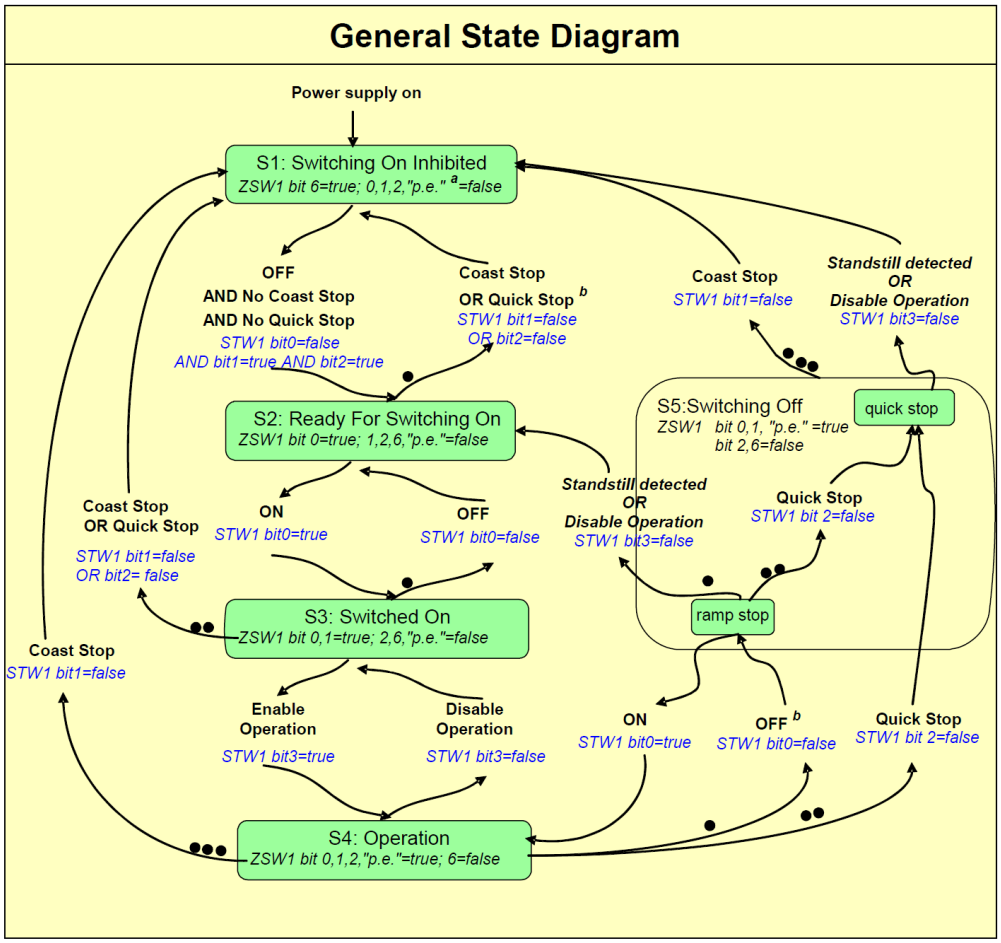
The cyclic data exchange comprises sending/receiving of e.g. target values, actual values, control word, status word, etc. between the master and the DO. These data are transferred in real time.

The acyclic data exchange is used for configuring the drive. Normally, this process is not time-critical. Each DO has its own parameter manager which handles the access.

The Alarm Queue is used for signalling the master exception situations.

The isochronous operation requires PROFINET IRT (conformance class C).

6.2 General State Machine



6.3 Control word (STW1)

The S7 application has to set the bits in the control word 1 (STW1) accordingly in order to get through the PROFIdrive state machine to the Enable Mode (S4: Operation).

Depending on the operating mode some bits of the STW1 have different functions. The following tables show the bit assignment.

Table 6.1: Control word STW1: General

Bit	Description	Action if bit is set (= 1)
0	ON/OFF	Power stage on
1	Coast Stop	Do **not** coast stop the axis
2	Quick Stop	Axis performs **no** Quick Stop
3	Enable Operation	The drive runs-up to the setpoint
7	Fault Acknowledge	The group signal is acknowledged with a positive edge
10	Control by PLC	The drive receives its commands from a higher-level controller (PLC)

In velocity mode:

Table 6.2: Control word STW1: Velocity mode

Bit	Description	Action if bit is set (= 1)
4	Enable ramp generator	Use ramp
5	Unfreeze the ramp generator in the drive (ramp up or down is continued)	Unfreeze. (Bit not set: If frozen, the drive stays at current velocity without continuing to ramp up or down)
6	Enable setpoint	The setpoint of the ramp generator is switched through to the power stage
8	Jog 1 on/off	The drive runs up along the ramp to jogging setpoint 1. Prerequisite: Operation is enabled, drive is in standstill and STW1 bit 4, 5, 6 = 0.
9	Jog 2 on/off	
11 ... 15	reserved	

In positioning mode:

Table 6.3: Control word STW1: Positioning mode

Bit	Description	Action if bit is set (= 1)
4	Do not reject traversing task	A traversing task is activated using the positive signal edge at bit 6
5	No intermediate stop	Traversing task is executed without interruption. (If bit is not set (=0) drive brakes to n = 0 along a ramp, and remains stationary with the holding torque. The traversing task is continued when the bit changes to 1.)
6	Activate traversing task	The positive signal edge enables a traversing task or a new MDI setpoint.
8	Jog 1 on/off	The drive is in variable-speed mode with jogging setpoint 1. Prerequisite: Operation is enabled and no positioning procedure is active.
9	Jog 2 on/off	
11	Start homing	Homing Procedure is started with a change from 0 to 1. Bit 11 in the status word is set to 0 with the start of the Homing Procedure. Prerequisite: Operation is enabled.
11 ... 15	reserved	

6.4 Status word (ZSW1)

Depending on the operating mode several bits of the status word (ZSW1) have different functions. The assignment is shown in the following tables.

Table 6.4: Status word ZSW1: General

Bit	Description	Status if bit is set (= 1)
0	Ready to switch on	Power supply is switched on, power stage off
1	Ready to operate	Refer to STW1, bit 0
2	Operation enabled	Drive follows setpoint values
3	Fehler	Unacknowledged faults or currently not acknowledgeable faults (fault messages) are present (in the fault buffer).
4	No Coast Stop	Bit set: Coast stop **not** activated
5	No Quick Stop	Bit set: Quick stop **not** activated
6	Switching on inhibited	No switch-on of the power stage possible. The drive goes only again in the „Switched On" condition with bit 1 and bit 2 of the control word (STW1) set and afterwards bit 0 set.
7	Warning present	Warning information present in the service/ maintenance parameter; no acknowledgement.
9	Control requested	The higher-level controller (e.g. PLC) is requested to assume control

Table 6.5: Status word ZSW1: Velocity mode

Bit	Description	Status if bit is set (= 1)
8	Velocity error within tolerance range	Actual value is within a tolerance band; dynamic violations are permissible for $t < t_{max}$
10	Target velocity reached or exceeded	Actual value ≥ comparison value (setpoint) depending on object 0x5105 in % (x4000 = 100%)
11 ... 15	reserved	

Table 6.6: Status word ZSW1: Positioning mode

Bit	Description	Remarks
8	Following error within tolerance range	The dynamic comparison of the setpoint position with the actual position value is located within the defined following error window.
10	Target position reached	The position at the end of a traversing task is reached.
11	Home position set	Homing procedure was executed and home position is valid.
12	Traversing task acknowledgment	Using the positive edge, it is acknowledged that a new traversing task or MDI-setpoint was accepted.
13	Drive stopped	Signals that a traversing task has been completed or standstill for intermediate stop.
14 ... 15	reserved	

6.5 Supported PNUs

Table 6.7 lists the PROFIdrive specific basic parameters supported by the ECOVARIO 114D. Parameter access is handled via PAP (Parameter Access Point). As access model the ‚Base Mode Parameter Access Local' (Map Index = 0xB02E) model is used.
The access model is defined in the IEC 61800-7-303 chapter 4.6 „Parameter Access".

Table 6.7: PROFIdrive specific basic parameters

PNU	Name	Data type	Description
915	Configuration setpoint telegram	U16, RW	Reading or modifying the current telegram configuration
916	Configuration actual telegram	U16, RW	Reading or modifying the current telegram configuration
922	Telegram selection	U16, RO	Selected telegram (e.g. 1 = Standard telegram 1)
924	Pulse enabled	U16, RO	Number of the signal and bit position in which „Pulse enabled“ is encoded.
925	Sign of Life	U16, RW	Maximum error number of the „Sign-of-Life“ handling
930	Operating mode	U16, RW	1 = Velocity mode 2 = Positioning mode
944	Fault message counter	U16, RO	
947	Fault number	U16, RO	Latest fault and 7 preceding faults
964	Drive Unit Identification	U16, RO	Manufacturer coding, software version, date, axis number
965	Profile identification	U16, RO	0x0328 = PROFIdrive profile version 4.0
967	Control word 1	U16, RW	
968	Status word 1	U16, RO	
970	Reset (axis-specific)	U16, RO	By entering a 1 the factory settings are active for the selected axis after restart
971	Save (axis-specific)	U16, RO	By entering a 1 all parameters and checksums are stored
975	Drive Object Identification	U16, RO	Manufacturer coding, software version, date, supported application classes, selected axis
976	Reset (device)	U16, RO	By entering a 1 factory settings in the device active after reboot
977	Save (device)	U16, RO	By entering a 1 all parameters and checksums are stored
978	Driveobject IDs	DO-IDs	
980 ... 989	Parameter list	U16, RO	The parameters 980 to 989 map the complete inventory of the device parameters.

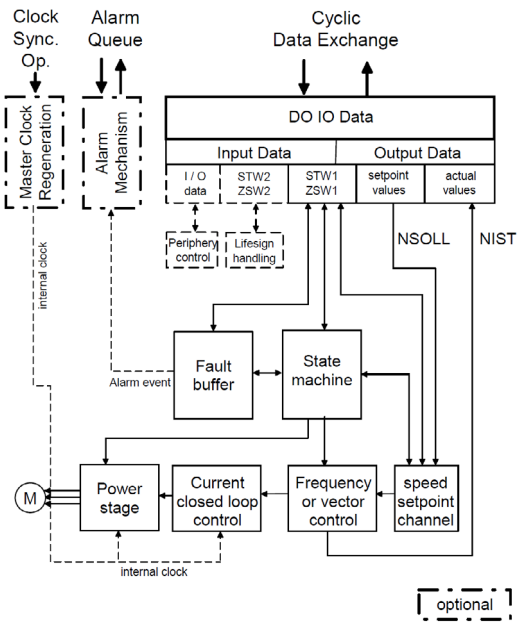
A detailed description of the PROFIdrive specific parameters and all other manufacturer specific parameters can be found in the document „PROFIdrive Parameterverzeichnis“.

6.6 Velocity Mode (Application class 1)

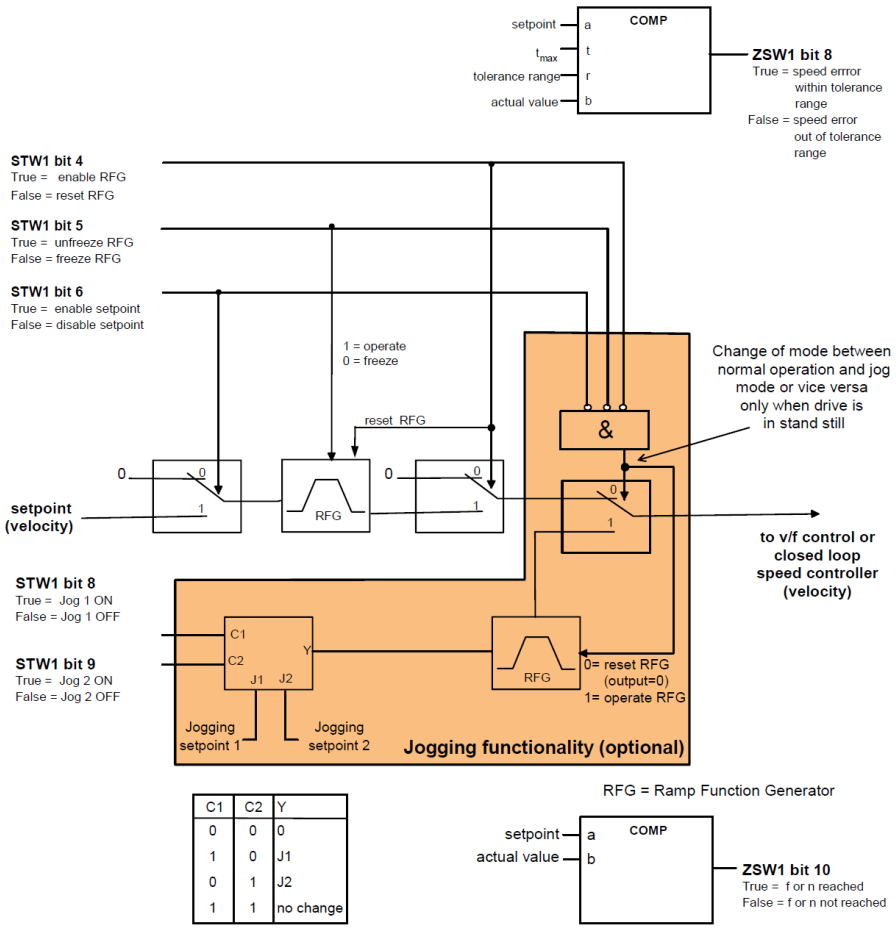
In this mode, the drive is controlled via a velocity set point. The velocity control is done completely in the servo amplifier.

The PROFINET field bus is the transmission medium between the higher-level controller and the servo amplifier. The Cyclic Data Exchange Communication Service is used.

The figure to the right shows the implementation of an Application Class 1 PROFIdrive Axis DO.



The figure below shows the functionality of the setpoint channel block for use in Application Class 1 and the effect of the STW1 control bits on the speed setpoint channel. The jogging functionality is optional.



C1	C2	Y
0	0	0
1	0	J1
0	1	J2
1	1	no change

Definition of the telegram (**Standard telegram 1**):

Notes:

➲ N_SOLL/N_IST WORD in % - x4000 = 100%
Object 0x5100 Reference velocity (default: 1000 rpm or 1000 mm/s)

➲ ON = 047F, OFF = 040F

IO Data No.	Setpoint	Actual value
1	STW1	ZSW1
2	NSOLL_A	NIST_A

Definition of the telegram (**Standard telegram 2**):

Notes:

➲ N_SOLL/N_IST DWORD in % - x4000 0000 = 100%
Object 0x5100 Reference velocity (default: 1000 rpm or 1000 mm/s)

➲ ON = 047F, OFF = 040F

IO Data No.	Setpoint	Actual value
1	STW1	ZSW1
2	NSOLL_B	NIST_B
3		
4	STW2	ZSW2

6.7 Positioning Mode (Application class 3)

In application class 3 the Drive Object (DO) provides a closed position control loop with its own position interpolation. The drive acts here as a one-axis positioning drive.

In the PROFIdrive profile two different submodes are possible, which allow the controlling device to access motion task parameters via I/O messaging:
- ➲ Submode „Programm mode“ (PSI), telegram 7
- ➲ Submode „Manual data input“ (MDI), telegram 9

Note: For telegram 9 a JAT-specific function block is provided.

6.7.1 Submode „Program mode“

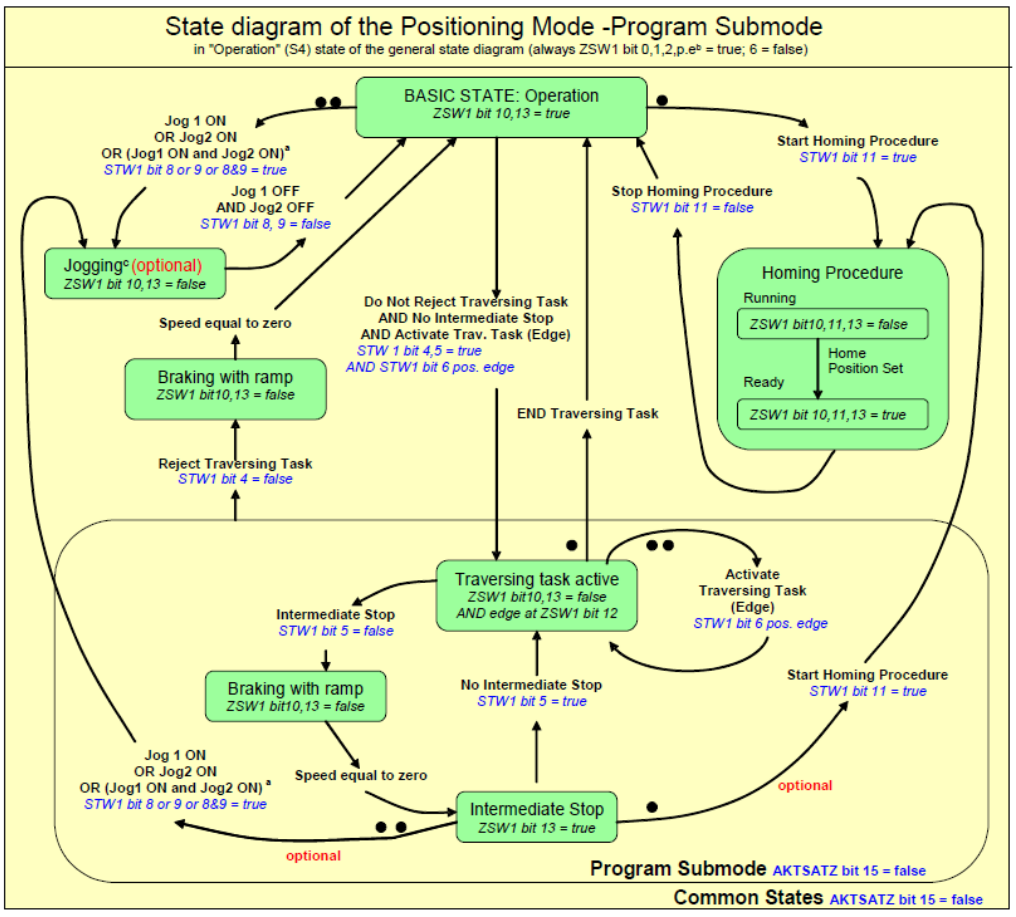
The Program Mode is used to start/switch to a specific predefined motion task via I/O messaging. Telegram 7 is used for this purpose.

The start of a motion program or the change to a new program while a program is still running is done by selecting the first motion record of the new program via SATZANW (while SATZANW Bit15 = 0) and a positive edge of STW1 Bit6. With AKTSATZ the number of the currently processed motion task can be read.

Telegram definition:

Note: ON = 047F, OFF = 040F

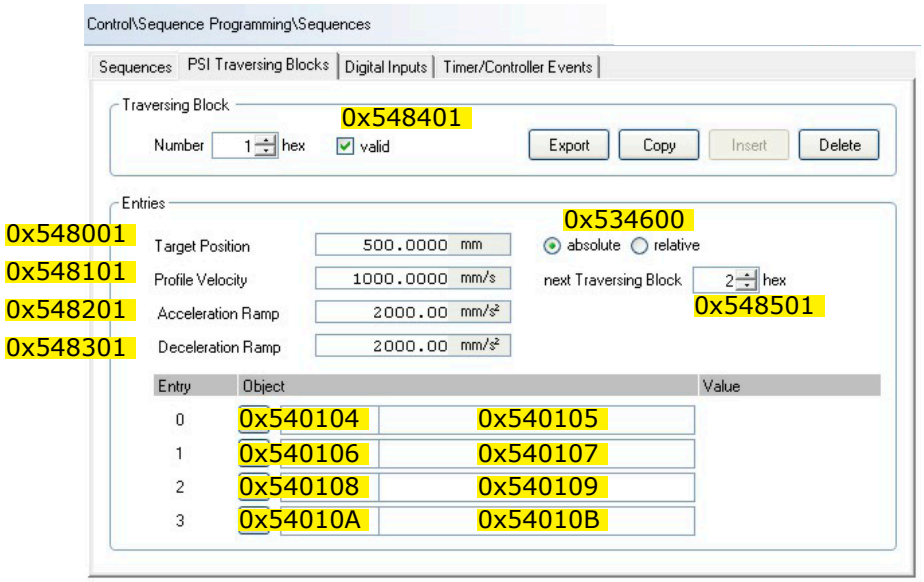
IO Data No.	Setpoint	Actual value
1	STW1	ZSW1
2	SATZANW	AKTSATZ



In the servo amplifier up to 128 motion tasks can be specified. They contain the target position (absolute or relative), profile velocity, acceleration and deceleration ramp and concatenation information to the next motion task. Furthermore, four additional selectable objects can be set within a motion task.

Normally, the motion tasks are set by the controller, (e.g. Siemens S7-1500). However, it is also possible to set them by means of ECO Studio, e.g. for commissioning purposes.

In the ECO Studio user interface the motion tasks can be set in the *Expert Mode* under ***Control\Sequence programming\Sequences*** in the ***PSI Traversing blocks*** tab:



Below the parameters used for motion task 1 are listed as an example (with correspondances above in the ECO Studio window). The digits highlighted in yellow contain the number of the motion task (hexadecimal, i.e. for motion task 10 a „0A“ would be inserted here instead of the „01“).
0x540100 – 0x540103 are permanently pre-set by 0x548501 and 0x534600.

User Defined Variables

...	548001	PSI Zielposition 001	18000,0000	°
...	548101	PSI Profilgeschwindigkeit 001	5,0000	U/s
...	548201	PSI Beschleunigung 001	10,00	U/s²
...	548301	PSI Bremsbeschleunigung 001	10,00	U/s²
...	548401	PSI Sequenz 001 gültig	1	dec
...	540100	PSI 01 Mapping 01	53460010	hex
...	540101	PSI 01 Wert 01	1	dec
...	540102	PSI 01 Mapping 02	54850110	hex
...	540103	PSI 01 Wert 02	1	dec
...	540104	PSI 01 Mapping 03	00000000	hex
...	540105	PSI 01 Wert 03	0	dec
...	540106	PSI 01 Mapping 04	00000000	hex
...	540107	PSI 01 Wert 04	0	dec
...	540108	PSI 01 Mapping 05	00000000	hex
...	540109	PSI 01 Wert 05	0	dec
...	54010A	PSI 01 Mapping 06	00000000	hex
...	54010B	PSI 01 Wert 06	0	dec
...	~			

6.7.2 Submode „Manual Data Input“ (MDI)

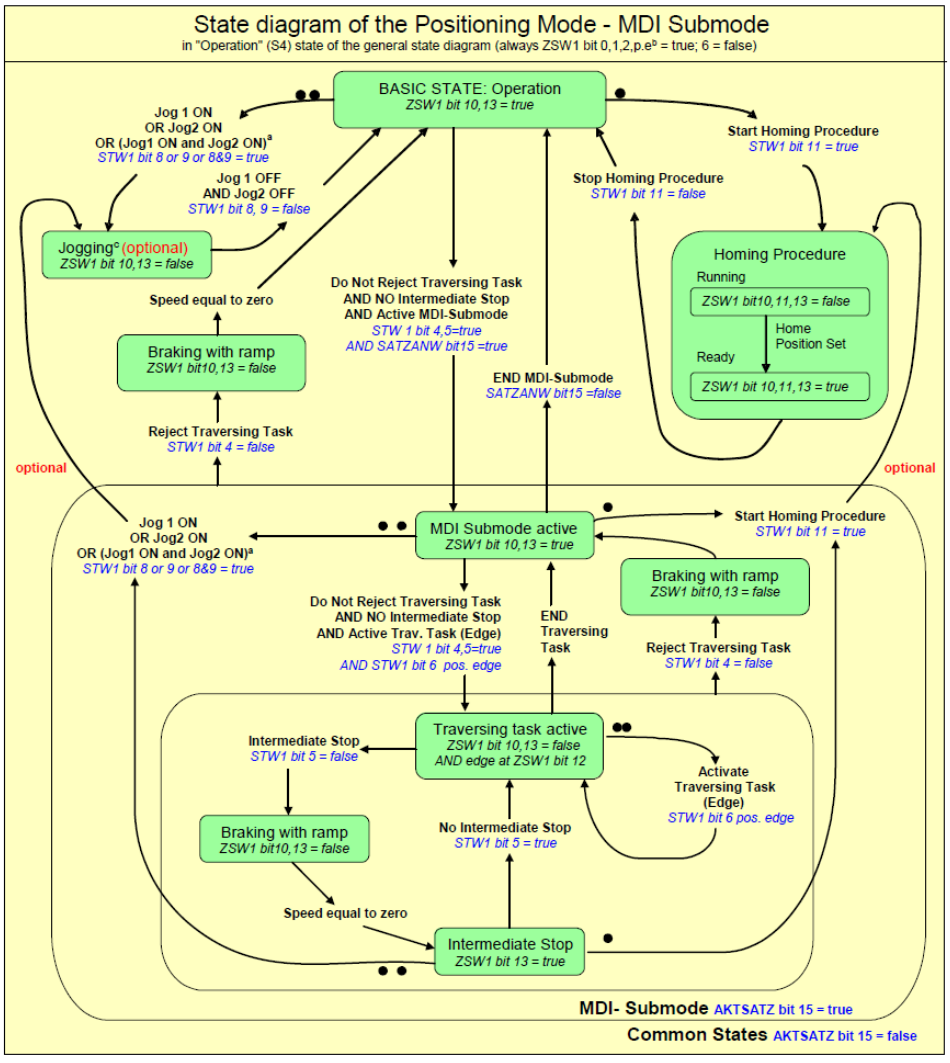
The „Manual Data Input“ (MDI) is used to start a motion task which is configured directly by the received I7O data. For this, standard telegram 9 is used, which defines parameters like acceleration (MDI_ACC), deceleration (MDI_DEC), velocity (MDI_VEL) and target position (MDI_TAR_POS).

Notes:

- for MDI Mode SATZANW has to be set to x8000
- MDI_VELOCITY in %
  - x4000 0000 = 100%
  Object 0x5100 Reference velocity (default: 1000 rpm) or 1000 mm/s
- MDI_ACC and MDI_DEC in %
  - x4000 = 100%
  - Object 0x5101 Reference acceleration (default: 100 rev/s² or 100 mm/s²)
- MDI_MOD 0= rel. or 1 = abs.
- ON = 047F / OFF = 040F

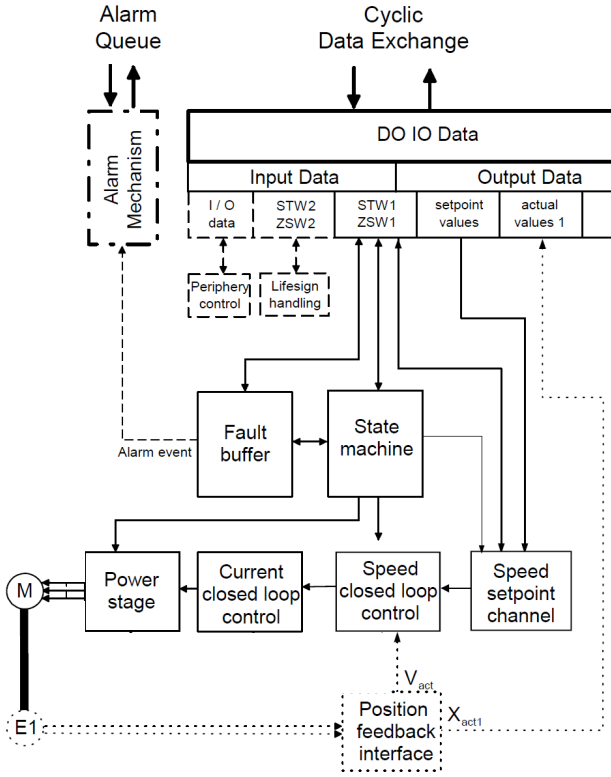
Definition of the telegram:

IO Data No.	Setpoint	Actual value
1	STW1	ZSW1
2	SATZANW	AKTSATZ
3	STW2	ZSW2
4	MDI_TARPOS	XIST_A
5		
6	MDI_VELOCITY	
7		
8	MDI_ACC	
9	MDI_DEC	
10	MDI_MOD	

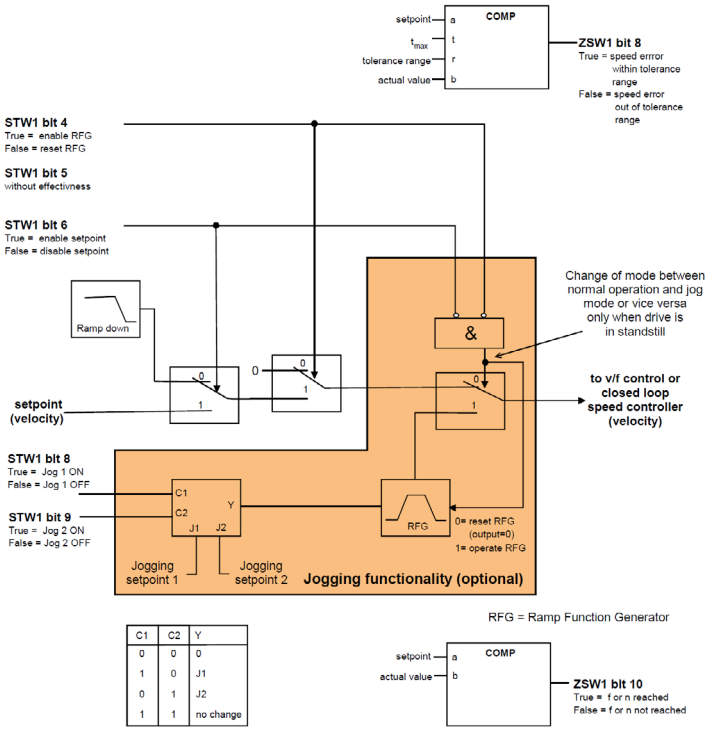


6.8 Motion Control in Application Class 4

Application Class 4 typically is used for positioning by use of servo drives with central interpolation. Therefore the position closed loop control located on the control is closed via the PROFIdrive Interface. For this Application Class, the Position Feedback Interface and the Clock Synchronous Operation is necessary. The following figure shows the implementation of an Application Class 4 PROFIdrive Axis DO.



In Application Class 4, there is typically no need for a Ramp Function Generator (RFG) in the speed setpoint channel. Therefore, a more simple speed setpoint channel without RFG functionality may be used (optional). The figure on the right shows the functionality of this reduced speed setpoint channel block for use in Application Class 4 and the effect of the STW1 control bits on the speed setpoint channel.
With this speed setpoint channel, STW1 bit 5 has no effect. The jog mode functionality is optional.
If the drive supports parameter PNU930 (operating mode) than the drive shows the functionality of the reduced speed setpoint channel by answering a parameter read request on PNU930 with a value == 3.



For Motion Control in application class 4 the standard telegrams 3 or 5 can be used.

Definition of the telegram (**Standard telegram 3**):
Contents: n-set interface, 32 bit, with one sensor

IO Data No.	Setpoint	Actual value
1	STW1	ZSW1
2	NSOLL_B	NIST_B
3		
4	STW2	ZSW2
5	G1_STW	G1_ZSW
6		G1_XIST1
7		
8		G1_XIST2
9		

Definition of the telegram (**Standard telegram 5**):
The standard telegram 5 is derived from standard telegram 3 and is intended for use with the Dynamic Servo Control (DSC). Via DSC the positioning accuracy of the axes can be increased, because the position control is shifted into the drive for faster interference compensation. The drive must be connected to the automation system via an isochronous bus system.
Contents: n-set interface, 32 bit, with one sensor, additionally position deviation (XERR) and position controller gain (KPC) in the setpoint value for DSC.

IO Data No.	Setpoint	Actual value
1	STW1	ZSW1
2	NSOLL_B	NIST_B
3		
4	STW2	ZSW2
5	G1_STW	G1_ZSW
6	XERR	G1_XIST1
7		
8	KPC	G1_XIST2
9		

7 Acyclic Reading and Writing of Parameters

7.1 General

The procedure described here for the acyclic reading of parameters from the drive and the acyclic writing of parameters into the drive is applied if outside of the technology objects specific parameters have to be written from master to drive or read from it.

The acyclic data exchange is used for events that do not keep repeating. Examples of acyclic data traffic are sending parameterization and configuration data when starting a peripheral device (from the central controller to the peripheral device) or sending a diagnostic message from the peripheral device to the central controller in operation.

Detailed information can be found in IEC 61800-7-203 (chapter 6.2.3 Base Mode Parameter Access).
The application note „Acyclic reading and writing parameters of the frequency inverters via PROFINET and PROFIBUS, Version 1.3, Entry ID: 29157692“ (Siemens AG) also provides further information on this topic.

Note:
Only the following Record Data Object is supported:

Parameter Access Service	Index
Base Mode Parameter Access - Locale	0xB02E

7.2 Sequence

For communication the system function blocks (SFB) „RDREC“ and „WRREC“ (instances of SFB52, SFB53) are used, which are provided in the ***Instructions*** > ***Advanced Instructions*** palettes in the TIA Portal.
The sequence for the common parameter access is as follows:
➲ Request DB (RECORD) is sent to the servo drive via WRREC
➲ Response DB (RECORD) is filled with the answer via RDREC.

Example for Read Parameter single (DB in „not optimized block access“, otherwise the DB length has to be set manually):

```
// Example for single parameter query (control via VAT RDWRREC_ single_multi)
// 1. WRREC with Request DB
// 2. RDREC to Response DB
//

IF „S1_SwitchRDWRREC_single“ = 1 AND #ENABLEo = 0 AND #zsta = 0 THEN
    #zsta := 1;
END_IF;

CASE #zsta OF
```

```
 1:  // init

     #ENABLEo := 1;

     #HW_ID:="PAP";//from „Display all variables"´-> System constants

     //Determine length Request_DB_single !DB in NOT optimized block access
     #Z1:= ATTR_DB(ATTRIB=>#Attribut1,
                   DB_LENGTH=>#LEN_Request_DB_single,
                   DB_NUMBER:=17,//Request_DB_single[DB17]
                   REQ:=1);

     //Determine length Response_DB_single !DB in NOT optimized block access
     #Z2 := ATTR_DB(ATTRIB => #Attribut2,
                    DB_LENGTH => #LEN_Response_DB_single,
                    DB_NUMBER := 18,//Response_DB_single[DB18]
                    REQ := 1);
    #zsta := 2;

 2:  // start WRREC

    IF #ENABLEo = 1 AND #BUSY_WR = 0 AND #BUSY_RD = 0 AND #REQ_RD = 0 THEN
        #REQ_WR := 1;
        #zsta := 3;
    END_IF;

 3:  // Funktion WRREC

   „WRREC_DB_single"(REQ := #REQ_WR,
                     ID := #HW_ID,
                     INDEX := W#16#B02E,// Base Moder Parameter Access - Local
                     LEN := UDINT_TO_UINT(IN:=#LEN_Request_DB_single),// Length Request DB in byte (10byte)
                     DONE => #DONE_WR,
                     BUSY => #BUSY_WR,
                     ERROR => #ERROR_WR,
                     STATUS => #SATUS_WR,
                     RECORD := „S1_Request_DB_single".Record_Request);// DB Request RECORD is a Struct
   #zsta := 31;

31: // wait for Done

    IF #BUSY_WR = 1 THEN
        #zsta := 3;
    ELSIF #BUSY_WR = 0 AND #DONE_WR = 1 THEN
        #zsta := 4;
        #REQ_WR := 0;
    ELSIF #ERROR_WR = 1 THEN
        #zsta := 6;
    END_IF;

 4:  // start RDREC

    IF #ENABLEo = 1 AND #BUSY_RD = 0 AND #BUSY_WR = 0 AND #REQ_WR = 0 THEN
        #REQ_RD := 1;
        #zsta := 5;
    END_IF;
```

```
  5:  // Function RDREC

      „RDREC_DB_single“(REQ := #REQ_RD,
                        ID := #HW_ID,
                        INDEX := W#16#B02E, // Base Moder Parameter Access - Local
                        MLEN := UDINT_TO_UINT(IN:=#LEN_Response_DB_single),// Length Response DB in byte (8byte)
                        VALID => #VALID_RD,
                        BUSY => #BUSY_RD,
                        ERROR => #ERROR_RD,
                        STATUS => #STATUS_RD,
                        RECORD := „SL_Response_DB_single“.record_Antwort);// DB Request RECORD is a Struct
      #zsta := 51;

  51: // wait for done

      IF #BUSY_RD = 1 THEN
          #zsta := 5;
      ELSIF #BUSY_RD = 0 AND #VALID_RD = 1 THEN
          #REQ_RD := 0;
          #zsta := 6;
      ELSIF #ERROR_RD = 1 THEN
          #zsta := 6;
      END_IF;

  6:  // ready

      #zsta := 0;
      #REQ_RD := 0;
      #REQ_WR := 0;
END_CASE;

#ENABLEo := „SL_SwitchRDWRREC_single“;
```

7.3 Data format of the „Base Mode Parameter Access“

In the following table the telegram format of the parameter access for a parameter request and a parameter answer is shown.

Base Mode Parameter Request	Byte n	Byte n+1	Byte address n
Request header	Request reference	Request identification	0
	Axis No	No. of Parameters (n)	2
1. Parameter address	Attribute	No. of elements	4
	Parameter Number (PNU)		
	Sub index		
i. Parameter address	.....		4+6*(n-1)
1. Parameter value (only for „Change parameter“)	Format	No. of values	4+6*n
	Values		
	.....		
			4 + 6 × n +...+ (Format_n × Qty_n)

Base mode parameter request[1]

1) Table from „iec61800-7-203-PROFINETDriveProfile.pdf“

Base Mode Parameter Answer	Byte n	Byte n+1	Byteadresse n
Response header	Request reference mirrored	Response identification	0
	Axis-No./DO-ID mirrored	No. of Parameters (n)	2
1. Parameter value (only after „Request“ inquiry)	Format	No. of values	4
	Values or Error Values		
	.....		
	.....		
n. Parameter value	Format	No. of values	
	Values or Error Values		
	.....		
			4 +... + (Format_n × Qty_n)

Base mode parameter response[2]

The payload is described in the following:

- **Request reference:**
  The request reference is set by the master and mirrored back by the slave in the answer telegram. Based on this reference, the master has the option of assigning each response telegram to a job telegram. A master changes the request reference with each new job.
- **Request ID**
  This identifier basically has the task of describing the type of parameter handling. Currently two different identifiers are defined:
  - Request Parameter
  - Change Parameter
  Further details on the identifier are available in the table „Payload“.
- **Response ID**
  This identifier contains information about the output of a job. If a job has been executed correctly, the Response Id is equal to the Request ID. If an order could not be executed, an identifier listed in the table „Payload“ results.
- **Axis No.**
  This value can be used to specifically address individual axes in a multi-axis system (Axis No. ⇨ 0 single axis). In the ***Base Mode Parameter Access Local*** without function.
- **No. of Parameters**
  Number of parameters processed in a job.
- **Attribute**
  Describes the individual access to a parameter structure. For example, if you want to access the actual numeric value or the parameter description text. Further information is contained in the table „payload“.
- **Number of Elements**
  When accessing an array or string, this area contains the field size or string length.
- **Parameter Number**
  Contains the parameter number (PNU).
- **Subindex**
  Addresses the first array element of a parameter or the beginning of a string. Furthermore, description texts and text arrays can be addressed with it.
- **Format**
  Specifies the data format of the transmitted value.
- **Number of values**
  Number of the following values
- **Values**
  Parameter values

2) Table from „iec61800-7-203-PROFINETDriveProfile.pdf“

Field name	Data type	Value	Meaning	Comment
Request reference	Unsigned8	0x00 0x01..0xFF	Reserved	
Request ID	Unsigned8	0x00 0x01 0x02 0x03..0x03F 0x40..0x7F 0x80..0xFF	Reserved Request parameter Change Parameter Reserved Manufacturer-specific Reserved	Parameter Write Request
Response ID	Unsigned8	0x00 0x01 0x02 0x03..0x3F 0x40..0x7F 0x80 0x81 0x82 0x83..0xBF 0xC0..0xFF	Reserved Request parameter (+) Change Parameter (+) Reserved Manufacturer-specific Reserved Request parameter (-) Change Parameter (-) Reserved Manufacturer-specific	Parameter Read Re-sponse (+) positive answer (-) negative answer
Axis No	Unsigned8	0x00 0x01..0xFE 0xFF	Device Representative Axis-Number 1..254 Reserved	0 = single axis
No. of Parameters	Unsigned8	0x00 0x01..0x27 0x28..0xFF	Reserved Quantity 1..39 Reserved	Limited by telegram length
Attribute	Unsigned8	0x00 0x10 0x20 0x30 0x40..0x70 0x80..0xF0	Reserved Value Description Text Reserved Manufacturer-specific	
No. of Elements	Unsigned8	0x00 0x01..0xEA 0xEB..0xFF	Special Function Quantity 1..234 Reserved	Limited by telegram length
Parameter Number	Unsigned16	0x0000 0x0001… 0xFFFF	Reserved Number 1..65535	
Subindex	Unsigned16	0x0000… 0xFFFF	Number 1..65535	
Format	Unsigned8	0x00 0x01..0x36 0x37..0x3F 0x40 0x41 0x42 0x43 0x44 0x45..0xFF	Reserved Data Types (cf. table „Standard data types“ on page 37) Reserved Zero Byte Word Double Word Error Reserved	
No. of Values	Unsigned8	0x00..0xEA 0xEB..0xFF	Quantity 0..234 Reserved	Limited by telegram length
Error Number	Unsigned16	0x0000… 0x00FF	Error Numbers (cf. table„error numbers“ on page 38)	

Payload

Data types used in Profile PROFIdrive	Reference to definition	Coding (decimal)
Boolean	Boolean (see IEC 61158-5-10)	1
Integer8	Integer8 (see IEC 61158-5-10)	2
Integer16	Integer16 (see IEC 61158-5-10)	3
Integer32	Integer32 (see IEC 61158-5-10)	4
Integer64	Integer64 (see IEC 61158-5-10)	55
Unsigned8	Unsigned8 (see IEC 61158-5-10)	5
Unsigned16	Unsigned16 (see IEC 61158-5-10)	6
Unsigned32	Unsigned32 (see IEC 61158-5-10)	7
Unsigned64	Unsigned64 (see IEC 61158-5-10)	56
FloatingPoint	Float32 (see IEC 61158-5-10)	8
FloatingPoint64	Float64 (see IEC 61158-5-10)	15
VisibleString	VisibleString (see IEC 61158-5-10)	9
OctetString	OctetString (see IEC 61158-5-10)	10
UNICODEString	UNICODEString (see IEC 61158-5-10)	39
TimeOfDay (with date indication)	TimeOfDay (see IEC 61158-5-10)	12
TimeDifference (do not use) *	TimeDifference (see IEC 61158-5-10)	13
Date	BinaryDate (see IEC 61158-5-10)	50
TimeOfDay without date indication	TimeOfDay (see IEC 61158-5-10)	52
TimeDifference with date indication	TimeDifference (see IEC 61158-5-10)	53
TimeDifference without date indication	TimeDifference (see IEC 61158-5-10)	54
*) Data type TimeDifference should not be used for new device implementations. For device implementation use data types TimeDifference with date indication and TimeDifference without date indication instead. For compatibility reasons all three data types of TimeDifference shall be supported by a controller implementation.		

Standard data types

Error number	Meaning	Generated if	Additional Information
0x00	Parameter number not allowed	Access to non-existent parameters	0
0x01	Parameter value cannot be changed	Change request to a parameter value that can not be changed	Sub index
0x02	Value range of the parameter exceeded or fallen below	Change request with value outside the value limits	Sub index
0x03	Bad parameter sub index	Acess to a sub index of an array parameter which is not available. Must not appear for not-array parameters	Sub index
0x04	Parameter is not an array	Access with sub index to non-indexed parameters	0
0x05	Wrong parameter data type	Change request with value that does not match the data type of the parameter	0
0x06	Change access with nonzero value that is not allowed	Change request with a value other than 0, where this is not allowed	Sub index
0x07	Change access to a description item that can not be changed	Change access to a description item that can not be changed	Sub index
0x09	No description text available	Access to non-existent description (parameter value exists)	0
0x0B	No priority	Change request without rights to change parameters	0
0x0F	No text array available	Access to a text array that is not available (parameter value is present)	0
0x11	Job can not be executed due to system health	Access is temporarily unavailable for reasons that are not specified in detail	0
0x14	Value not allowed	Change request with a value that is within the value limits, but it is not allowed for other long-term reasons (parameter with defined individual values)	Sub index
0x15	Answer telegram is too long	The length of the current response exceeds the maximum transmittable length of the telegram	-
0x16	Parameter address not allowed	Illegal value or value not supported for the attribute, the number of elements, parameter number or subindex or a combination	-
0x17	Illegal format	Write request: Illegal format of the request or the format of the parameter data is not supported	0
0x18	Number of parameter values are not consistent	Write request: Number of parameter data values does not match the number of elements in the parameter address	0
0x19	Job to an axis that does not exist	Access to an axis / DO that does not exist	0
0x20	Parameter text element cannot be modified	Change request for a parameter text element which cannot be changed	Sub index
0x21	Service is not supported	Invalid request identifier (ID = 0x80)	-
0x66	Number of parameter elements too large	No. of Elements too large	-
0x65 - 0xFF	Manufacturer specific		-

Error numbers

Note on error handling in case of system error message of RDREC:

If one of the above error numbers contains additional information (see table column), the response DB (Record) will increase by 2 bytes per parameter contained. This case must be taken into account when dimensioning the response DB (record) (dimension large enough).

If the response DB is not large enough, an error message is generated by the system function, e.g. error code DE80-A800.

7.4 Reading a single parameter value

In the example the value of parameter 965 is read (profile identification, cf. table 6.7).

Request DB (RECORD)

Auftrag_DB_single

	Name	Datentyp	Offset	Startwert	Remanenz	Erreichbar aus HMI	Sichtbar in HMI	Einstellwert	Kommentar
1	Static								
2	Record_Auftrag	Struct	0.0						
3	Anforderungsreferenzt	Byte	0.0	16#03					Nummer der Aufträge
4	Anforderungs_ID	Byte	1.0	16#01					Paramter lesen = 1; Parameter schreiben = 2
5	Achse	Byte	2.0	16#01					Adressierung einer Achse / Multi-Achsenantrieb
6	Anzahl_der_Parameter	Byte	3.0	16#01					lesen der Parameter r0965
7	Attribut_01	Byte	4.0	B#16#10					Typ des Objektes: 10 Wert, 20 Beschreibung, 30 Text
8	Anzahl_der_Elemente_01	Byte	5.0	16#00					Anzahl der Indizes 1
9	Paramter_Nummer_01	Word	6.0	965					Parameter r0965[1] (965 dez= 3C5 hex) nur 1
10	Subindex_01	Word	8.0	0					Subindex

Response DB (RECORD)

Antwort_DB_single

	Name	Datentyp	Offset	Startwert	Remanenz	Erreichbar a..	Sichtbar i...	Einstellwert	Kommentar
1	Static								
2	record_Antwort	Struct	0.0						
3	Anf_referenz_gespiegelt	Byte	0.0	16#0					Nummer des Auftrag gespiegelt
4	Antwort_ID	Byte	1.0	16#0					Parameter lesen
5	Achse_gespiegelt	Byte	2.0	16#0					Adressierung der Achse gespiegelt
6	Anz_Parameter	Byte	3.0	16#0					Anforderung auf die Anzahlt der Parameter
7	Format_Parameter_1	Byte	4.0	16#0					Anforderung auf das Format des Parameter 1
8	Anz_Parameterwerte_1	Byte	5.0	16#0					Anforderung auf die Anzahl der Werte des Parameters 1
9	Param1_Wert_01	Word	6.0	16#0					Wert aus Index 0

7.5 Reading several parameter values

In the example the values of parameters 965 (profile identification), 2714 (switching threshold full-step operation), 930 (operating mode) und 5100 (normalization velocity) are read

Request DB (RECORD)

Auftrag_DB_multi

	Name	Datentyp	Offset	Startwert	Remanenz	Erreichbar a..	Sichtbar i...	Einstellwert	Kommentar
1	Static								
2	Record_Auftrag	Struct	0.0						
3	Anforderungsreferenzt	Byte	0.0	16#1					Nummer der Aufträge
4	Anforderungs_ID	Byte	1.0	16#1					Paramter lesen = 1; Parameter schreiben = 2
5	Achse	Byte	2.0	16#1					Adressierung einer Achse / Multi-Achsenantrieb
6	Anzahl_der_Parameter	Byte	3.0	16#4					lesen der Parameter 965[1], 930[61], 2714[1] u. 5100
7	Attribut_01	Byte	4.0	16#10					Typ des Objektes: 10 Wert, 20 Beschreibung, 30 Text
8	Anzahl_der_Elemente_01	Byte	5.0	16#0					Anzahl der Indizes 1
9	Paramter_Nummer_01	Word	6.0	16#3c5					Parameter 965[1] - Profilidentifikation (965 dez= 3C5 hex) nur 1
10	Subindex_01	Word	8.0	16#0					Subindex
11	Attribut_02	Byte	10.0	16#10					Typ des Objektes: 10 Wert, 20 Beschreibung, 30 Text
12	Anzahl_der_Elemente_02	Byte	11.0	16#0					Anzahl der Indizes 1
13	Parameter_Nummer_02	Word	12.0	16#2714					Parameter 2714[1] -SingleStepVelo (=CANObjekt 270207) nur 1
14	Subindex_02	Word	14.0	16#0					Subindex
15	Attribut_03	Byte	16.0	16#10					Typ des Objektes: 10 Wert, 20 Beschreibung, 30 Text
16	Anzahl_der_Elemente_03	Byte	17.0	16#0					Anzahl der Indizes 1
17	Parameter_Nummer_03	Word	18.0	16#3A2					Parameter 930[1] - Betriebsart (930 dez= 3A2 hex) nur 1
18	Subindex_03	Word	20.0	16#0					Subindex
19	Attribut_04	Byte	22.0	16#10					Typ des Objektes: 10 Wert, 20 Beschreibung, 30 Text
20	Anzahl_der_Elemente_04	Byte	23.0	16#0					Anzahl der Indizes 1
21	Parameter_Nummer_04	Word	24.0	16#5100					Parameter 5100[1] - Velo_Norm_Para nur 1
22	Subindex_04	Word	26.0	16#0					Subindex

Response DB (RECORD)

Antwort_DB_multi

	Name	Datentyp	Offset	Startwert	Remanenz	Erreichbar a..	Sichtbar i...	Einstellwert	Kommentar
1	Static								
2	record_Antwort	Struct	0.0						
3	Anf_referenz_gespiegelt	Byte	0.0	16#0					Nummer des Auftrag gespiegelt
4	Antwort_ID	Byte	1.0	16#0					Parameter lesen
5	Achse_gespiegelt	Byte	2.0	16#0					Adressierung der Achse gespiegelt
6	Anz_Parameter	Byte	3.0	16#0					Anforderung auf die Anzahlt der Parameter
7	Format_Parameter_1	Byte	4.0	16#0					Anforderung auf das Format des Parameter 1
8	Anz_Parameterwerte_1	Byte	5.0	16#0					Anforderung auf die Anzahl der Werte des Parameters 1
9	Param1_Wert_01	Word	6.0	16#0					Wert aus Index 0
10	Format_Parameter_2	Byte	8.0	16#0					Anforderung auf das Format des Parameter 2
11	Anz_Parameterwerte_2	Byte	9.0	16#0					Anforderung auf die Anzahl der Werte des Parameters 2
12	Param2_Wert_01	DWord	10.0	16#0					Wert aus Index 0
13	Format_Parameter_3	Byte	14.0	16#0					Anforderung auf das Format des Parameter 3
14	Anz_Parameterwerte_3	Byte	15.0	16#0					Anforderung auf die Anzahl der Werte des Parameters 3
15	Param3_Wert_01	Word	16.0	16#0					Wert aus Index 0
16	Format_Parameter_4	Byte	18.0	16#0					Anforderung auf das Format des Parameter 4
17	Anz_Parameterwerte_4	Byte	19.0	16#0					Anforderung auf die Anzahl der Werte des Parameters 4
18	Param4_Wert_01	Word	20.0	16#0					Wert aus Index 0

7.6 Writing a single parameter value

In the example the value of parameter 5100 (normalization speed) is set to 0x3E8 (decimal: 1000) rpm.

Request DB (RECORD)

Auftrag_DB_single_write

	Name	Datentyp	Offset	Startwert	Remanenz	Erreichbar a..	Sichtbar i...	Einstellwert	Kommentar
1	Static								
2	Record_Auftrag	Struct	0.0						
3	Anforderungsreferenzt	Byte	0.0	16#04					Nummer der Aufträge
4	Anforderungs_ID	Byte	1.0	16#02					Paramter lesen = 1; Parameter schreiben = 2
5	Achse	Byte	2.0	16#01					Adressierung einer Achse / Multi-Achsenantrieb
6	Anzahl_der_Parameter	Byte	3.0	16#01					schreiben 5100
7	Attribut_01	Byte	4.0	16#10					Typ des Objektes: 10 Wert, 20 Beschreibung, 30 Text
8	Anzahl_der_Elemente_01	Byte	5.0	16#00					Anzahl der Indizes 1
9	Paramter_Nummer_01	Word	6.0	16#5100					Parameter 5100[1] Normierungsgeschwindigkeit
10	Subindex_01	Word	8.0	16#00					Subindex
11	Data_type	Byte	10.0	16#03					Daten Typ (x03 = integer 16)
12	No_of_values	Byte	11.0	16#01					Number of values = Number of elements
13	Value	Word	12.0	16#03e8					Wert x3E8 auf 5100

Response DB (RECORD)

The Response-DB contains feedback information about the write process, no values are read back.

7.7 Writing several parameter values

In the example the values of the PSI motion task are written into the drive:

- ➲ Set parameter 5480 (PSI target position) to 0
- ➲ Set parameter 5481 (PSI profile velocity) to 0x000FA000
- ➲ Set parameter 5482 (PSI acceleration) to 00001588
- ➲ Set parameter 5483 (PSI deceleration)
- ➲ Set parameter 5400 (PSI mapping)
- ➲ Set parameter 5454 (PSI sequence valid).

Request DB (RECORD)

Auftrag_DB_multi_write

	Name	Datentyp	Offset	Startwert	Remanenz	Erreichbar a...	Sichtbar i...	Einstellwert	Kommentar
1	Static								
2	Record_Auftrag	Struct	0.0						
3	Anforderungsreferenzt	Byte	0.0	16#02					Nummer der Aufträge
4	Anforderungs_ID	Byte	1.0	16#02					Paramter lesen = 1; Parameter schreiben = 2
5	Achse	Byte	2.0	16#01					Adressierung einer Achse / Multi-Achsenantrieb
6	Anzahl_der_Parameter	Byte	3.0	16#06					6 Parameter
7	Attribut_1	Byte	4.0	B#16#10					Typ des Objektes: 10 Wert, 20 Beschreibung, 30 Text
8	Anzahl_der_Elemente_1	Byte	5.0	16#01					Anzahl der Indizes 1
9	Paramter_Nummer_1	Word	6.0	16#5480					Parameter 5480[1]PSI Zielposition 000
10	Subindex_1	Word	8.0	16#00					Subindex
11	Attribut_2	Byte	10.0	B#16#10					Typ des Objektes: 10 Wert, 20 Beschreibung, 30 Text
12	Anzahl_der_Elemente_2	Byte	11.0	16#01					Anzahl der Indizes 1
13	Paramter_Nummer_2	Word	12.0	16#5481					Patameter 5481[1] PSI Profilgeschwindigkeit 000
14	Subindex_2	Word	14.0	16#00					Subindex
15	Attribut_3	Byte	16.0	B#16#10					Typ des Objektes: 10 Wert, 20 Beschreibung, 30 Text
16	Anzahl_der_Elemente_3	Byte	17.0	16#01					Anzahl der Indizes 1
17	Paramter_Nummer_3	Word	18.0	16#5482					Parameter 5482[1] PSI Beschleunigung 000
18	Subindex_3	Word	20.0	16#00					Subindex
19	Attribut_4	Byte	22.0	B#16#10					Typ des Objektes: 10 Wert, 20 Beschreibung, 30 Text
20	Anzahl_der_Elemente_4	Byte	23.0	16#01					Anzahl der Indizes 1
21	Paramter_Nummer_4	Word	24.0	16#5483					Parameter 5483[1] PSI Bremsbeschleunigung 000
22	Subindex_4	Word	26.0	16#00					Subindex
23	Attribut_5	Byte	28.0	B#16#10					Typ des Objektes: 10 Wert, 20 Beschreibung, 30 Text
24	Anzahl_der_Elemente_5	Byte	29.0	16#04					Anzahl der Indizes 4
25	Paramter_Nummer_5	Word	30.0	16#5400					Parameter 5400[1] - PSI_Mapping 000
26	Subindex_5	Word	32.0	16#00					Subindex
27	Attribut_6	Byte	34.0	16#10					Typ des Objektes: 10 Wert, 20 Beschreibung, 30 Text
28	Anzahl_der_Elemente_6	Byte	35.0	16#01					Anzahl der Indizes 1
29	Paramter_Nummer_6	Word	36.0	16#5484					Parameter 5484[1] - PSI Sequenz 000 gültig
30	Subindex_6	Word	38.0	16#00					Subindex
31	Data_type_1	Byte	40.0	16#07					Daten Typ (x03 = integer 16 x04 = integer 32)
32	No_of_values_1	Byte	41.0	16#01					Number of values = Number of elements
33	Value_1	DWord	42.0	16#00000016					Wert 0 auf 5480
34	Data_type_2	Byte	46.0	16#07					Daten Typ (x03 = integer 16 x04 = integer 32)
35	No_of_values_2	Byte	47.0	16#01					Number of values = Number of elements
36	Value_2	DWord	48.0	16#000FA000					Wert 000FA000 auf 5481
37	Data_type_3	Byte	52.0	16#07					Daten Typ (x03 = integer 16 x04 = integer 32)
38	No_of_values_3	Byte	53.0	16#01					Number of values = Number of elements
39	Value_3	DWord	54.0	16#00001388					Wert 00001388 auf 5482

Response DB (RECORD)

The Response DB contains feedback information about the write process, no values are read back.

Notes:

Parameter Description:
- ➲ Analogous to Read Parameter Single
- ➲ Attribute has to be x20

Parameter Text:
- ➲ Analogous to Read Parameter Single
- ➲ Attribute has to be x30

9 Error handling

9.1 Warnings

Warnings are messages that automatically acknowledge after the cause has been eliminated. They serve as a warning level, so that appropriate countermeasures can be taken in good time to avoid a fault condition.

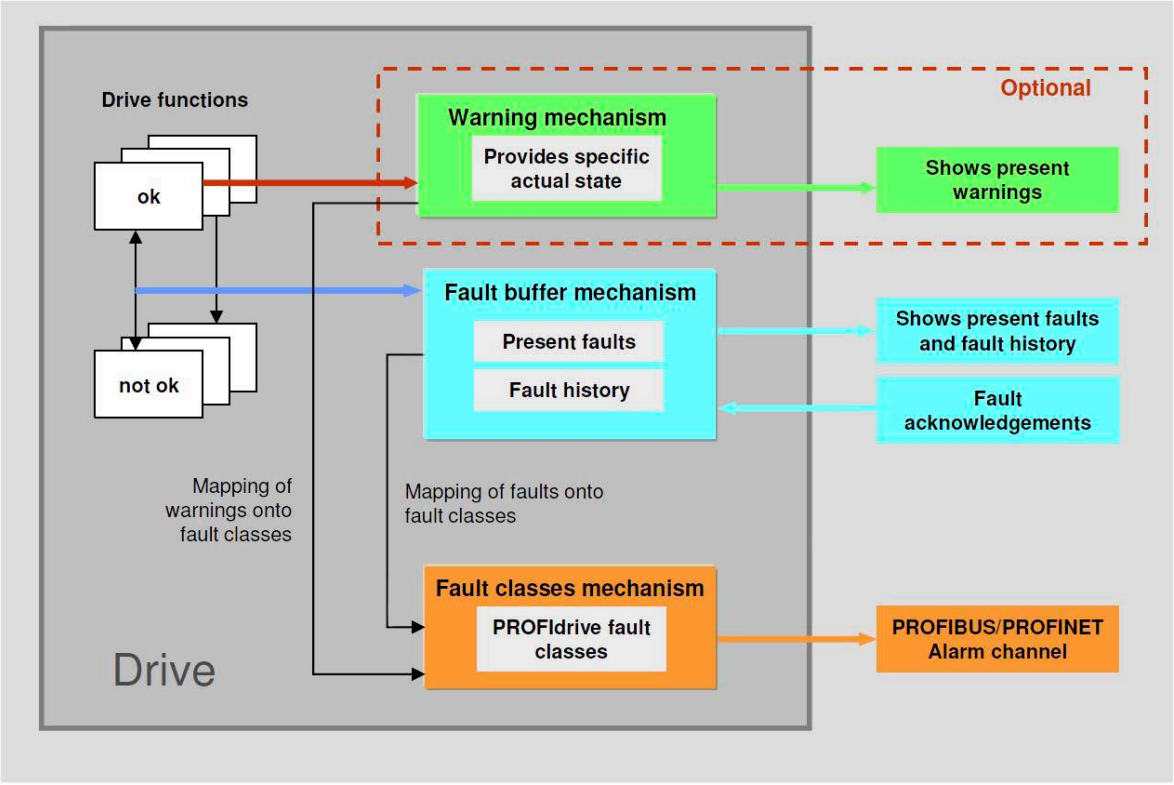
In principle, several warnings can be present at the same time (for example, „Increased temperature of the motor winding“ and „DC link voltage too low“). Warnings, unlike errors, do not stop the drive.

For the warning mechanism, parameters are defined in the profile, each of which represents a so-called warning word. Each occurring warning within a drive or a drive axis is mapped to one bit of the warning word.

9.2 Faults

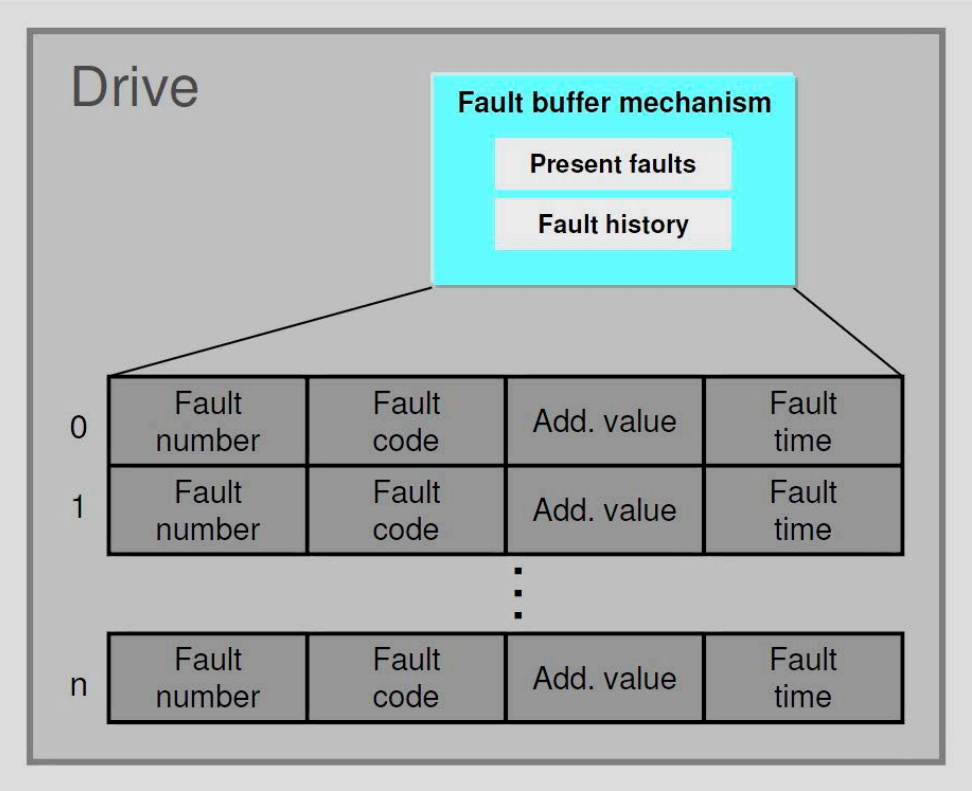
A fault condition in the drive (e.g., overtemperature) always triggers a device specific response, i. As a rule, the drive is switched off. At the same time, one or more fault messages characterizing the fault state are entered in the fault buffer (see figure below).

An entry in the PROFIdrive fault buffer consists of the device-specific fault number, an optional application-specific fault code as well as optionally an associated value or a fault time. The device-specific parts fault number and fault value enable a very detailed device-specific diagnosis.



PROFIdrive diagnosis function

A fault must always be explicitly acknowledged by the user with a command after eliminating the cause of the fault. The acknowledged fault is not deleted, but archived in the fault buffer, which allows subsequent traceability of the faults. The size of the fault buffer can be determined device-specifically.



Mapping of the fault buffers to profile parameters

Annex A: Notes on the integration of the JAT servo amplifiers into a PROFIBUS-DP network

JAT servo amplifiers with PPROFIdrive profile can also be integrated into a **PROFIBUS-DP** network. Supported telegrams are 1, 2, 7 and 9. However, some particularities should be noted which are described in this chapter.

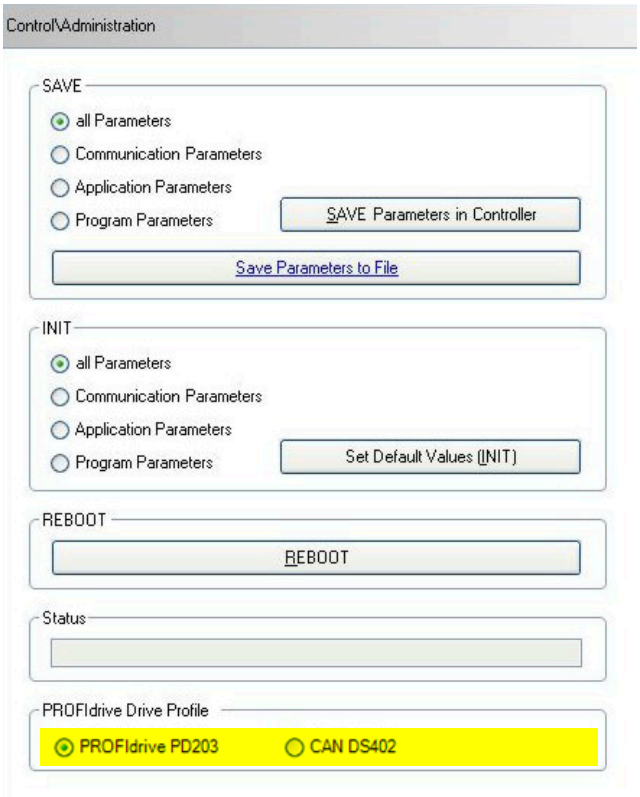
A.1 To chapter 3 „Requirements“

JAT servo amplifers with PROFIBUS-DP interface which support the PROFIdrive profile:

- ECOVARIO® 114DR-[G,F][J,L]-xxx-xxx from firmware release 4.55 on
- ECOVARIO®616[A,D]R-[G,F]J-xxx-xxx from firmware release 4.55 on

Another prerequisite for integration is that the servo amplifier has been switched to the PROFIdrive profile via a software tool ECOSTUDIO®:

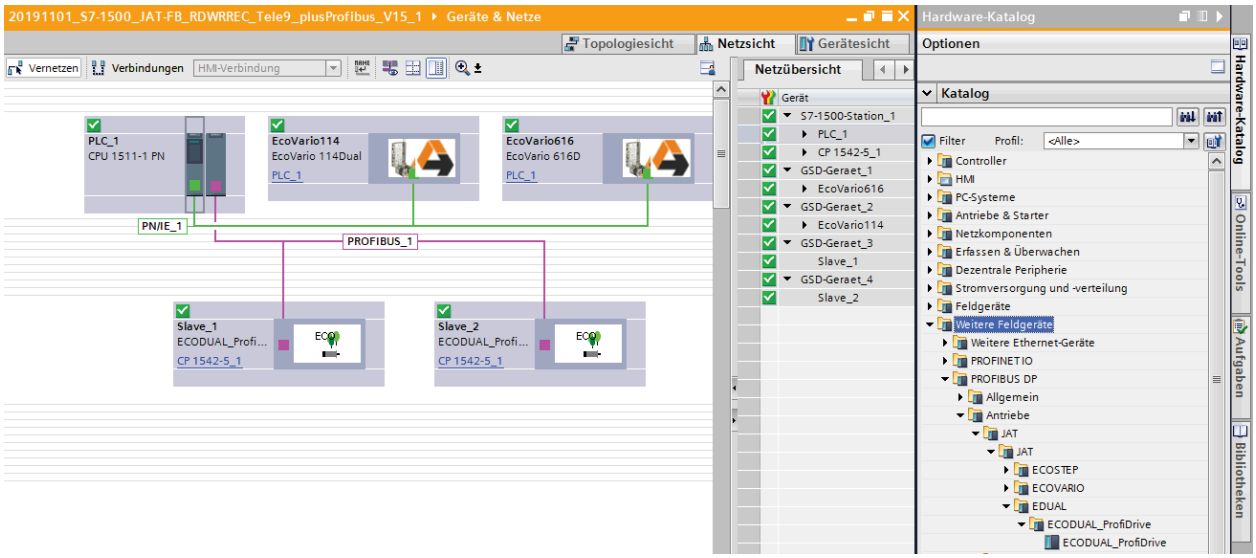
- Switching the **drive profile** from PROFIBUS/CAN servo amplifier (DS402) to PROFIBUS/**Profidrive (PD203)** in ECOSTUDIO® under **Control/Administration**:



A.2 To chapter 4.1 „Map the ECOVARIO 114 D / ECOVARIO 616 (D) by means of its GSD file“

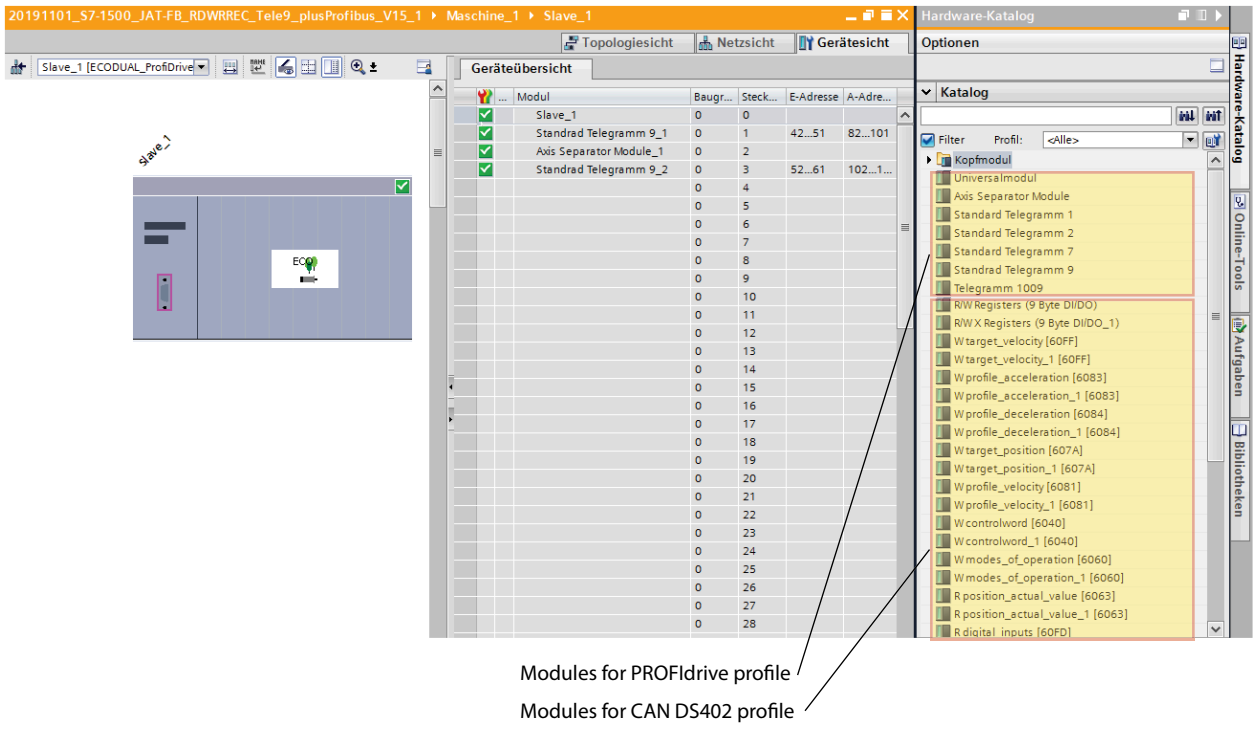
After installation of the **GSD file** the servo amplifiers are available in the hardware catalog and can be inserted into the network view of TIA.

Representation for PROFIBUS devices (ECODUAL_Profidrive):



Only the telegrams in the catalog are supported: 1, 2, 7 and 9.

Example: Selection telegram 9.
The **Axis Separator Module** must be inserted between the two standard telegram modules. All other modules shown are relevant for PROFIBUS devices according to the drive profile CAN DS402, cf. chapter A.1 „Switching the drive profile“.



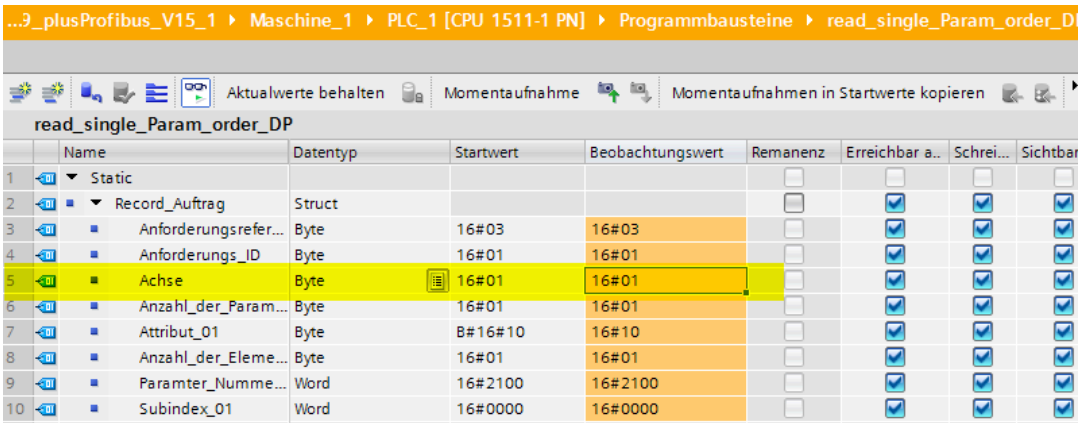
A.3 To chapter 7.2 „Parameter access“

For PROFIBUS devices only:
The system constant for ~Head-Slot has to be selected:

➲ HW_ID system constant => value „SlaveX~Head”

Switching between the axes is done with the switch „Axis“ in the Request DB (RECORD) (cf. chapter 7.4):

➲ Axis => value = 01 for axis A
➲ Axis => value = 02 for axis B



	Name	Datentyp	Startwert	Beobachtungswert	Remanenz	Erreichbar a..	Schrei...	Sichtbar
1	Static				☐	☐	☐	☐
2	Record_Auftrag	Struct			☐	☑	☑	☑
3	Anforderungsrefer...	Byte	16#03	16#03	☐	☑	☑	☑
4	Anforderungs_ID	Byte	16#01	16#01	☐	☑	☑	☑
5	Achse	Byte	16#01	16#01	☐	☑	☑	☑
6	Anzahl_der_Param...	Byte	16#01	16#01	☐	☑	☑	☑
7	Attribut_01	Byte	B#16#10	16#10	☐	☑	☑	☑
8	Anzahl_der_Eleme...	Byte	16#01	16#01	☐	☑	☑	☑
9	Paramter_Numme...	Word	16#2100	16#2100	☐	☑	☑	☑
10	Subindex_01	Word	16#0000	16#0000	☐	☑	☑	☑

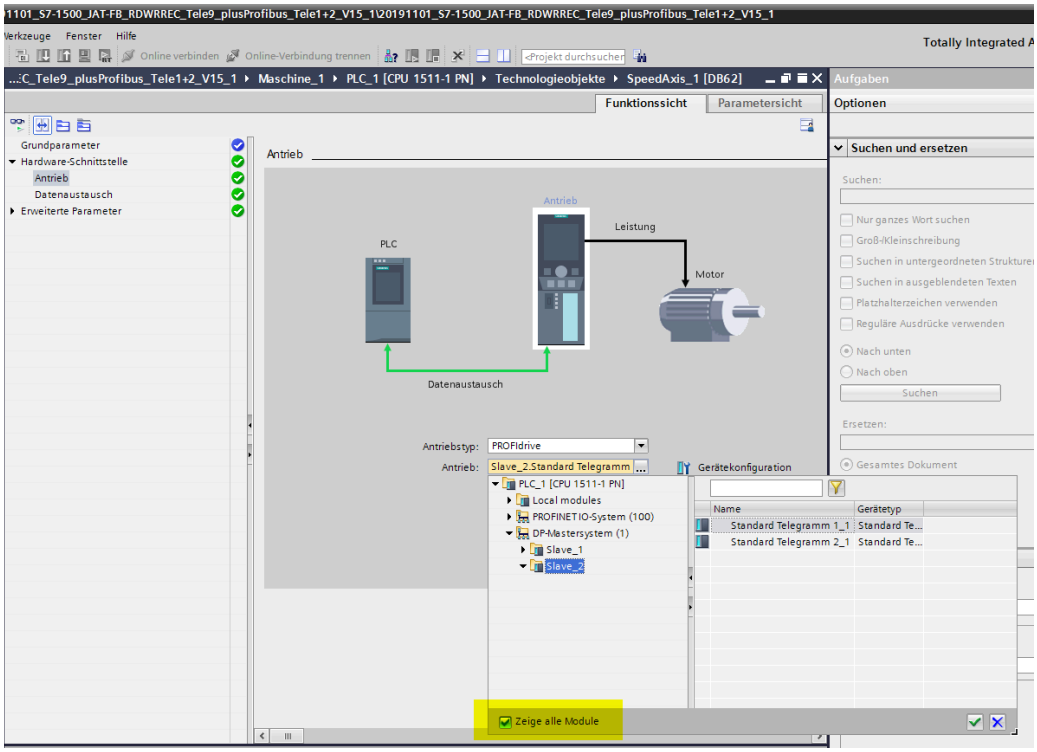
Note: The following Record Data Objects are supported:

Parameter Access Service	Index	Use
Base Mode Parameter Access - Local	0xB02E	PROFINET
Base Mode Parameter Access - Global	0xB02F = 47	PROFIBUS

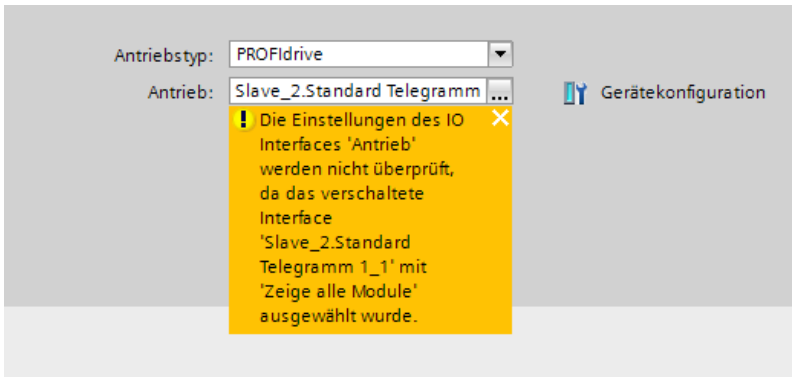
A.4 To chapter 5.2 „Selecting the drive on the technology object“

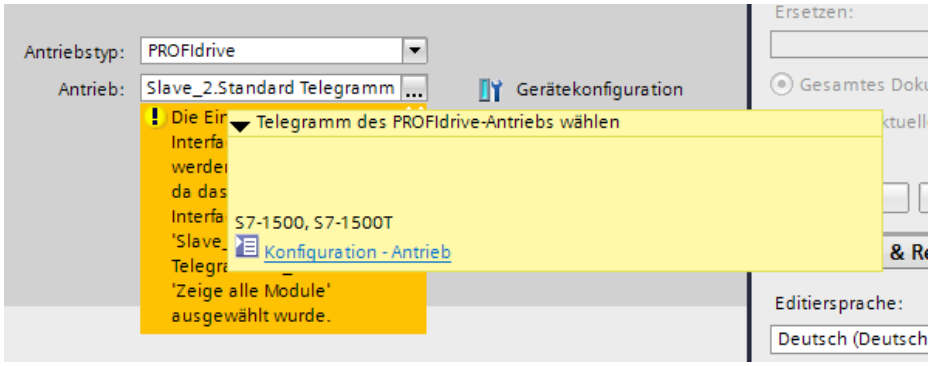
For the telegrams 1 and 2 the SIEMENS technology object **TO_SpeedAxis** can be used. The selection of the drive The selection of the drive in the technology object is described in chapter 5.2.

Note: When configuring the drive in the technology object **TO_SpeedAxis**, the option „***Show all modules***“ must be activated because SIEMENS does not support the automatic detection for third-party drives at the moment.



If the „*Show all modules*“ option has been activated, the following warning is displayed when selecting the drive (example):





Please note the SIEMENS help text concerning the drive selection with the „*Show all modules*“ option:

Antriebstyp: PROFIdrive

Datenanbindung

Wählen Sie in der Klappliste, ob die Datenanbindung direkt zum Antriebsgerät erfolgen soll, oder über einen im Anwenderprogramm bearbeitbaren Datenbaustein.

Antrieb/Datenbaustein

Wählen Sie im Feld "Antrieb" einen bereits konfigurierten PROFIdrive-Antrieb/Slot aus. Wenn Sie einen PROFIdrive-Antrieb ausgewählt haben, können Sie den PROFIdrive-Antrieb über die Schaltfläche "Gerätekonfiguration" konfigurieren.

Wenn kein PROFIdrive-Antrieb zur Auswahl steht, wechseln Sie in die Gerätekonfiguration und fügen Sie in der Netzsicht einen PROFIdrive-Antrieb hinzu.

Hinweis

Option "Zeige alle Module"

Wenn ein bereits konfigurierter PROFIdrive-Antrieb nicht zur Auswahl steht, zeigen Sie mit der Option "Zeige alle Module" alle erreichbaren Module an.

Wenn Sie die Option "Zeige alle Module" aktivieren, wird für alle angezeigten Module nur der Adressbereich der angezeigten Module überprüft. Wenn der Adressbereich eines Moduls groß genug für das gewählte PROFIdrive-Telegramm ist, können Sie das Modul auswählen. Stellen Sie daher sicher, dass Sie einen PROFIdrive-Antrieb auswählen.

Wenn Sie unter Datenanbindung "Datenbaustein" ausgewählt haben, wählen Sie hier einen zuvor erstellten Datenbaustein aus, der eine Variablenstruktur des Datentyps "PD_TELx" enthält ("x" steht für die zu verwendende Telegrammnummer).

Since the telegrams 1 and 2 selected in the HW configuration are standard telegrams according to the PROFIdrive profile, the communication is provided by the telegram structure and the length.