

**PROFI[®]
NET**

**PROFI[®]
BUS**



ECOVARIO[®] 114D / 616D
JAT Function Blocks for
PROFIdrive drive profile

Published Editions

Edition	Comment
December 2017	First English Edition
March 2019	Chapter 8.3: Table standard data types added
October 2019	Chapter 4 + 5: Editorial changes. Chapter 6.7.1. + 6.7.2: Order has been switched.
January 2020	Function blocks can be used for PROFIBUS as well

All rights reserved:

Jenaer Antriebstechnik GmbH
 Buchaer Straße 1
 07745 Jena
 Germany

No parts of this documentation may be translated, reprinted or reproduced on microfilm or in other ways without written permission by Jenaer Antriebstechnik GmbH.

The content of this document has been worked out and checked carefully. Nevertheless differences from the real state of the hard and software can never be fully excluded. Necessary corrections will be carried out in the next edition.

PROFINET® and PROFIBUS® are registered trademarks of PROFIBUS International (P.I.), Karlsruhe

ECOSTEP® and ECOVARIO® are registered trademarks of Jenaer Antriebstechnik GmbH, Jena.

Contents

1	About this Documentation	6
2	About PROFdrive	6
3	About PROFINET	7
4	About PROFIBUS	7
4	Preconditions for use	8
5.1	Example projects	8
5.2	Appropriate PLCs	8
5.3	Setup of the projects	9
5.4	Configuration overview	9
6	Description of the Function Blocks	12
6.1	Project Overview	12
6.2	Specific Blocks of the Project.....	13
6.3	JAT-FB_PA	14
6.4	JAT-FB_SFC.....	14
6.5	PZDtoAxisDB_SFC14	15
6.6	AxisDBtoPZD_SFC15	16
6.7	JAT-Axis_Tele9.....	17
6.7.2	Extended Functional Description of the Input Signals	18
6.7.1	AxisDB_A1	19
6.8	JAT-Read_IM	20
6.8.1	DB read_IM_answer.....	21
6.9	JAT-RDWR_PAP	22
6.10	JAT-RDWR_PAP_DP	23
6.10	PZDtoAxisDB	24
6.11	AxisDBtoPZD	25
7	User-defined Data Types	26
7.1	UDT_PAE	26
7.2	UDT_PAA.....	26
7.3	UDT_Axis_Control_IN	26
7.4	UDT_Axis_Status_OUT	27
8	Acyclic Reading and Writing of Parameters	28
8.1	General	28
8.2	Sequence	28
8.3	Data format of the „Base Mode Parameter Access“	28
8.4	Reading a single parameter value	33
8.4.1	DB read_single_Param_order	33
8.4.2	DB read_single_Param_answer.....	33
8.4.3	Example for reading a single parameter value	33
8.5	Reading several parameter values	34
8.6	Writing a single or several parameter values	35

9	Error handling	36
9.1	Warnings.....	36
9.2	Faults	36
10	Hardware Diagnosis.....	39
10.1	Operating display on the PROFINET option card.....	39
11	Comparison TIA Portal / Classic World.....	40
11.1	STEP7 CLASSIC WORLD:	40
11.1.1	Description.....	40
11.2	TIA PORTAL WORLD:.....	40

1 About this Documentation

This manual describes the procedure for functional integration of servo amplifiers series ECOVARIO®114D or ECOVARIO®616D in a „*PROFIdrive application profile for drives*“ via the **PROFINET** network or **PROFIBUS** (Master: Siemens SIMATIC PLC S7) by means of the JAT function blocks.

Prerequisite is that a physical connection exists between the servo amplifier and the PROFINET or PROFIBUS Master via the servo amplifier interface X22.

Appropriate servo amplifiers:

- ➔ PROFINET: ECOVARIO® 114DR-[I,H][J,L]-xxx-xxx, ECOVARIO®616[A,D]R-[I,H]J-xxx-xxx
- ➔ PROFIBUS: ECOVARIO® 114DR-[G,F][J,L]-xxx-xxx, ECOVARIO®616[A,D]R-[G,F]J-xxx-xxx

Further information:

- ➔ Hardware installation: Installation and Operating Instructions ECOVARIO®114 D, ECOVARIO®616(D)
- ➔ Documentation of the program package STEP7 or TIA-Portal (Siemens AG)
- ➔ Application Notes ECOVARIO®114D / ECOVARIO®616(D): Control via PROFINET (AN33)
- ➔ Software commissioning: ECOSTUDIO® Online Help or Operation Manual for ECOVARIO®

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.

Commissioning: Qualified personnel with a broad knowledge 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 SPS program package STEP7 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 PROFIdrive

The application profile PROFIdrive forms the foundation for virtually all drive tasks in the industrial automation technology in connection with controllers of the manufacturer SIEMENS.

It defines the device behavior and the access procedure to drive data of electrical drives at PROFIBUS and PROFINET and also optimally integrates the additional profiles PROFIsafe and PROFInergy.

The demand for continuity must be met in particular by industrial communication. This is only possible with a standardized drive interface which has been specified by PROFIdrive for PROFIBUS and PROFINET. In addition, the application-oriented profile standardized according to IEC 61800-7 contains standardized specifications (syntax and semantics) for communication between drives and automation systems for PROFIBUS and PROFINET ensuring vendor-independence, interoperability and investment protection.

3 About PROFINET

The field bus system PROFINET is standardized by the PROFIBUS Nutzerorganisation (PNO). 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 (Isochronous RealTime) there is a fix bus cycle for the actualisation between master and slave, similar to PROFIBUS DP.

No.	Application Class	Interface	Functions ^b
1	Standard Drive	n-setpoint, torque-setpoint, current-setpoint	Cyclic IO Data interface ^a
2	Standard drive with distributed technology controller (continuous process)	Technological setpoint-actual values (command variables)	Cyclic IO Data interface with Drive to Drive communication ^a
3	Single Axis positioning drive, with local Motion Control	pos-setpoint, run requests	Cyclic IO Data interface ^a
4	Motion Control with central interpolation and speed setpoint interface Optional: DSC (Dynamic Servo Control)	n-setpoint x-actual additionally, for DSC: $\Delta x (x_{err}), K_V (k_{pD})$	Cyclic IO Data interface, Clock Synchronous Operation, DSC (refer to 6.3.5)
5	Motion Control with central interpolation and position setpoint interface	x-setpoint	Cyclic IO Data interface, Clock Synchronous Operation ^c
6	Motion control for clocked processes, or distributed angular synchronism	Command variables, motion instructions	Cyclic IO Data interface, Clock Synchronous Operation, Drive to Drive communication
NOTE The Application Classes described here are assigned Profile functions in 6.3.13, which a drive manufacturer shall implement if he wishes to be compliant with the particular Application Class.			
^a The cyclic interface may also be operated clock-synchronously if, for example, it is a question of synchrony of the actions in the case of several drives.			
^b For all Application Classes: acyclic interface for parameters, diagnostics, identification.			
^c This Application Class is not described in this edition of the Profile. The dynamic characteristics of Application Class 5 are reached with Application Class 4 with DSC.			

PROFdrive Application Classes

4 About PROFIBUS

PROFIBUS is a non-proprietary, open fieldbus standard with a wide scope in the manufacturing and process automation. Die European standard EN 50170 describes three PROFIBUS forms: FMS, DP and PA.

PROFIBUS DP (Decentralized Peripherals) is optimized for speed, efficiency and low connection costs and is especially designed for the communication between automation systems and decen-tralized peripherals. Therefore, it is perfectly fit for controlling highly-dynamic drives such as ECOVARIO® and ECOSTEP® through a higher-level control system.

PROFIBUS DP operates according to the master / slave principle with overlaid token passing between different masters, if required. The master / slave communication is strictly cyclical, whereby the bus' time monitoring ensures that a failure of either master or slave is recognized almost instantly. Additionally, a standardized slave analysis through a master is implemented, offering extensive possibilities to transmit error and status information. A PROFIBUS DP device is identified through a configurable address. Furthermore, every device class has a so called ID num-ber that is identical for all devices of the same type.

4 Preconditions for use

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

- Servo amplifier ECOVARIO 114D or ECOVARIO 616(D) with PROFINET or PROFIBUS interface
- Siemens SIMATIC PLC S7-1500 / S7-1200 / S7-300 with appropriate PROFINET interface
- Program package „Totally Integrated Automation Portal“ (TIA Portal), SIMATIC Manager (Siemens AG)
- Service PC with commissioning tool ECOSTUDIO® V 3.2.0.0 or higher
- The latest GSDML file:
(e.g. from the Download section of our homepage www.jat-gmbh.de)
- The JAT package „FB PROFINET“ (Function blocks ECOVARIO), consists of the blocks listed in chapter 6.

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

Before using the function blocks, the drives of Jenaer Antriebstechnik must be integrated into a control project as described in Application Note 33 „Control via PROFINET“.

5 Setup Procedure

5.1 Example projects

For a better understanding of the administration of drives in the TIA portal the following example projects are contained in the function block package „FB PROFINET“ (function blocks ECOVARIO):

- YYYYMMDD_S7-1200_JAT-FB_RDWRREC_Tele9_V14.zap14
- YYYYMMDD_S7-1500_JAT-FB_RDWRREC_Tele9_V14.zap14
- YYYYMMDD_S7-300_JAT-FB_RDWRREC_Tele9_V14.zap14

Erweiterung für Ansteuerung über PROFIBUS mit PROFIdrive und Standard-Telegramm 9:

- YYYYMMDD_S7-1500_JAT-FB_RDWRREC_Tele9_plusProfibus_V15_1.zap15_1

Erweiterung für Ansteuerung über PROFIBUS mit PROFIdrive mit Standard-Telegramm 1 und 2 mit Technologieobjekten:

- YYYYMMDD_S7-1500_JAT-FB_RDWRREC_Tele9_plusProfibus_Tele1+2_V15_1.zap15_1

For integration into the programming environment STEP7 the following projects can be used:

- YYYYMMDD_JAT-FB_RDWRREC_Tele9_classic.zip

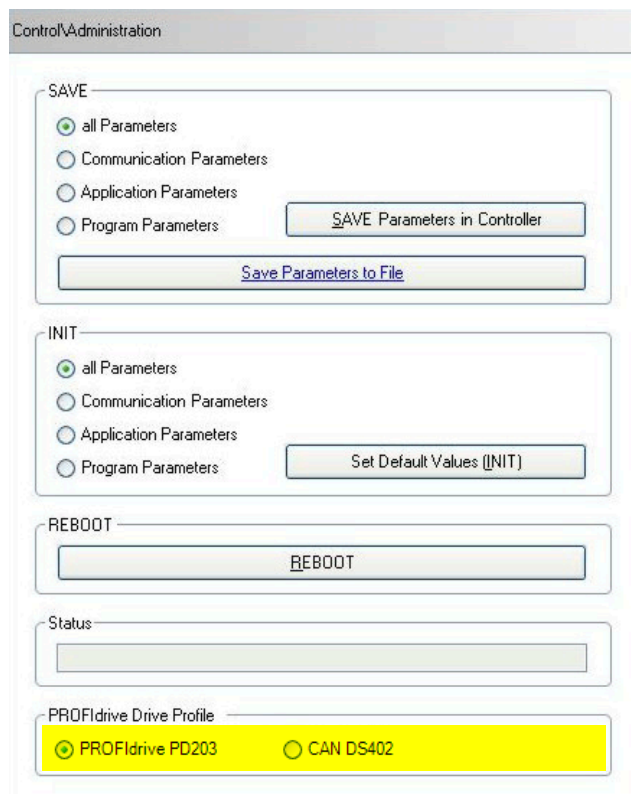
5.2 Appropriate PLCs

- S7-314-2 PN/DP
- S7-1215 C DC/DC/DC + CM 1243-5
- S7-1511 PN

5.3 Setup of the projects

- ➔ Loading one of the sample projects => Dearchive projects
After installing the GSDML file or the GSD file, the servo drives are available in the hardware catalog and can be inserted into the network view of TIA.
- ➔ Adaptation to the control environment => Adapt device configuration
- ➔ For PROFIBUS devices only: Switching the **drive profile** => from PROFIBUS/CAN servo amplifier (DS402) to PROFIBUS/PROFIdrive (PD203):

In ECOSTUDIO® under **Control/Administration**:



Control\Administration

SAVE

all Parameters
 Communication Parameters
 Application Parameters
 Program Parameters

SAVE Parameters in Controller

Save Parameters to File

INIT

all Parameters
 Communication Parameters
 Application Parameters
 Program Parameters

Set Default Values (INIT)

REBOOT

REBOOT

Status

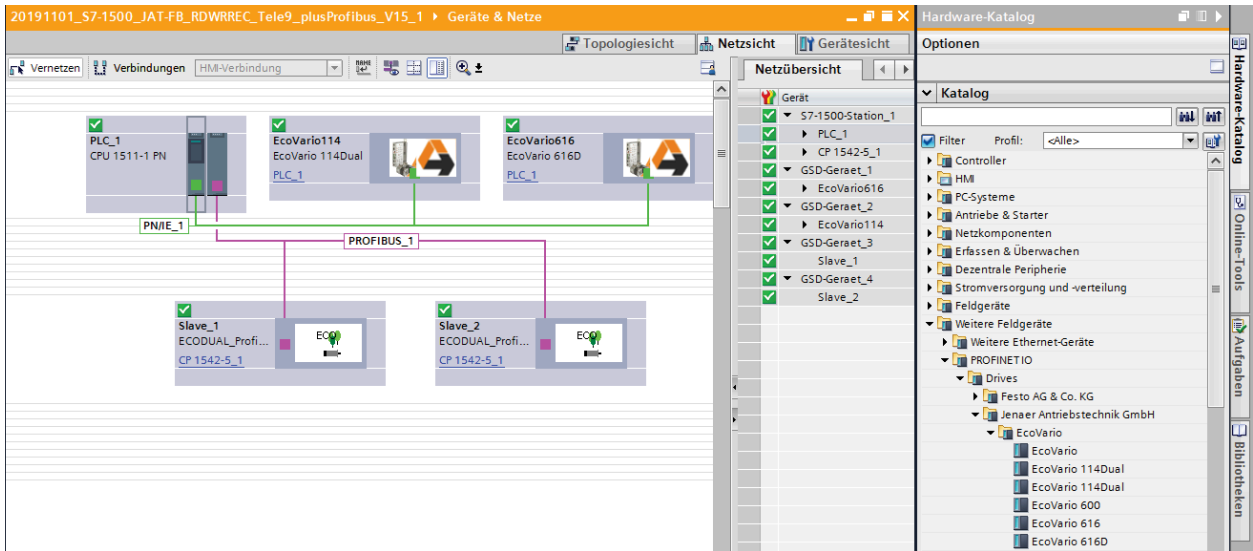
PROFIdrive Drive Profile

PROFIdrive PD203 CAN DS402

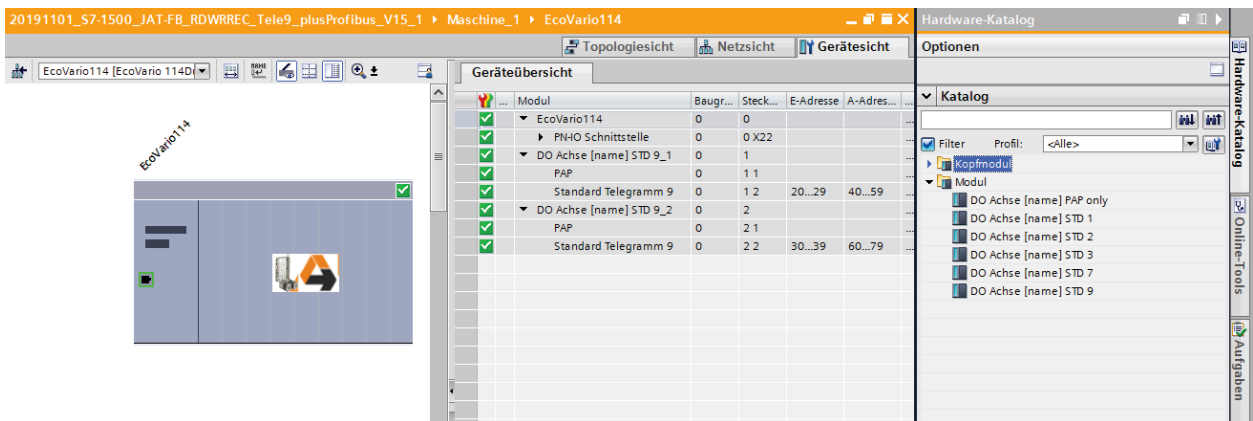
5.4 Configuration overview

Note: PROFINET devices and PROFIBUS devices are two different device types, which are integrated in different places in the hardware catalog.

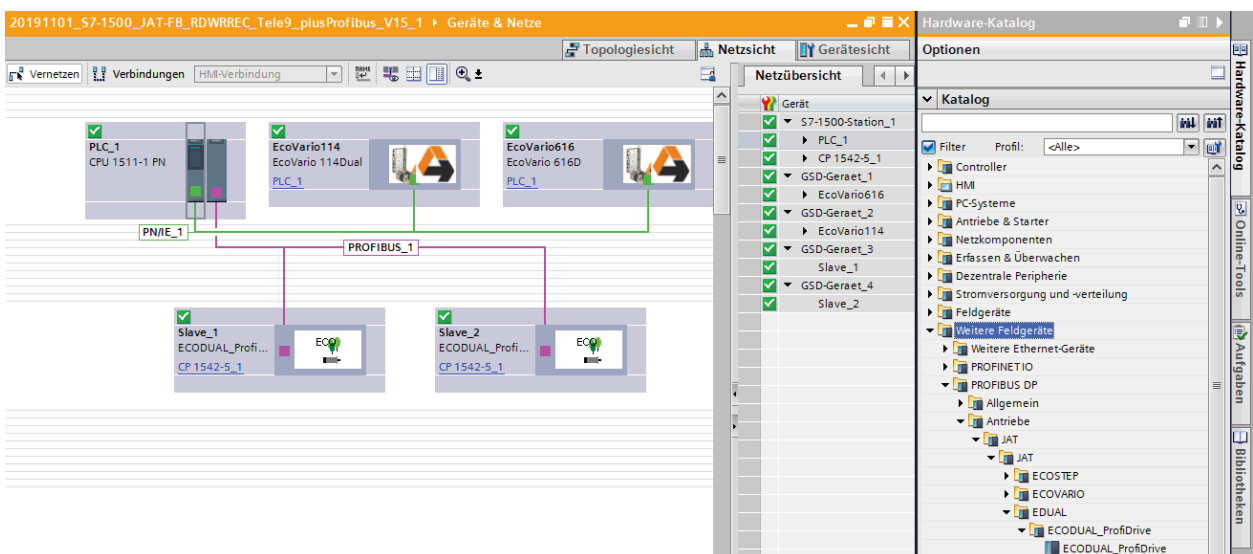
Representation for PROFINET devices:



Example: Selection of telegram 9:

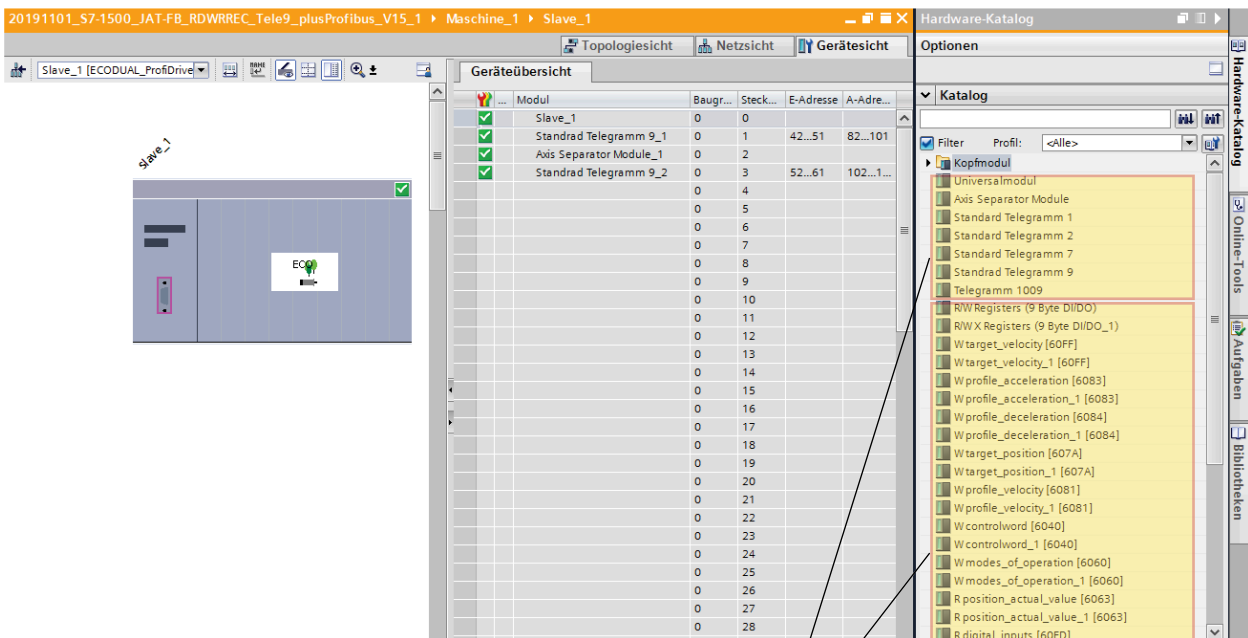


Representation for PROFIBUS devices (ECODUAL_Profidrive):



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 5.3 „Switching the drive profile“.

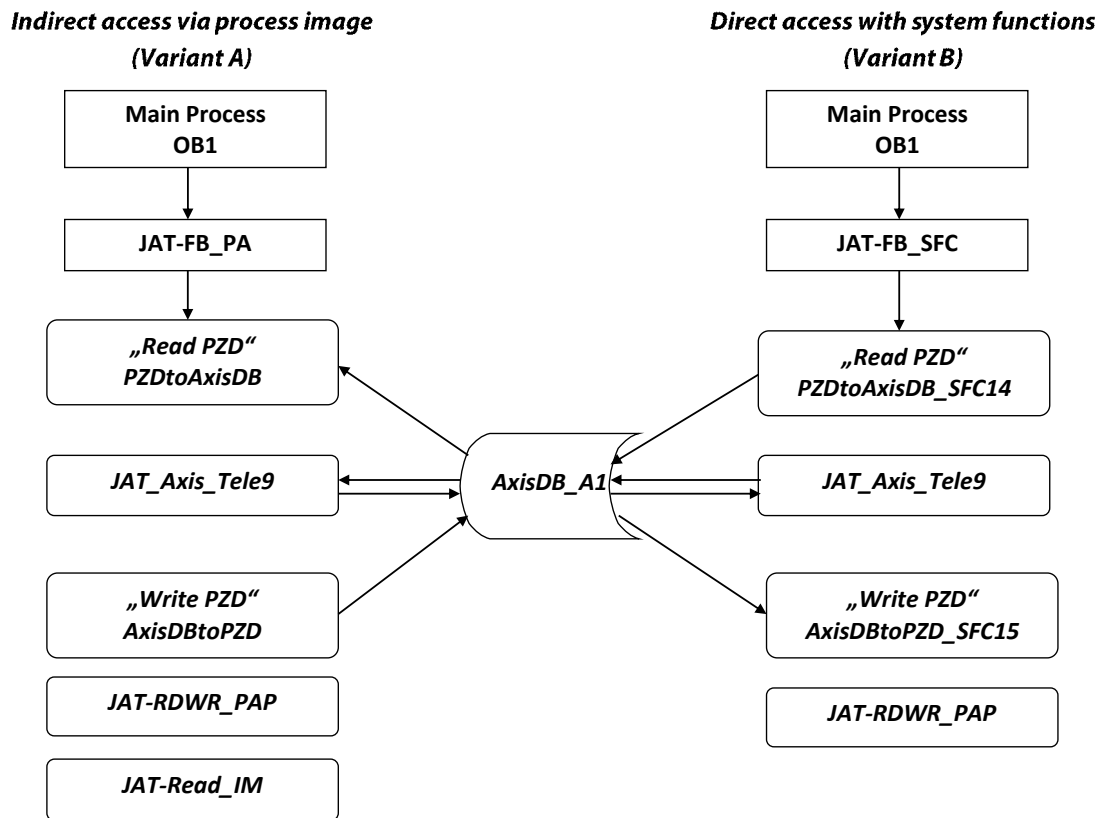


Modules for PROFdrive profile
 Modules for CAN DS402 profile

6 Description of the Function Blocks

6.1 Project Overview

Variants of access to axes in a SIEMENS PLC



Project Overview

6.2 Specific Blocks of the Project

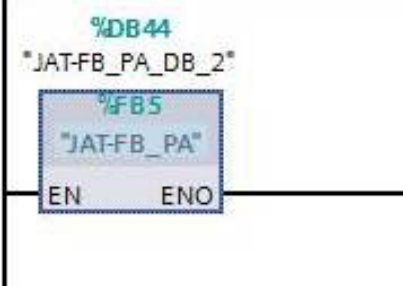
List of specific blocks in the project

Type	Symbolic Name	Description
OB1	„Main“	Main program „organisation block“, is executed continuously
DB	AxisDB_A1	Storage area for axis image axis1
DB	DB_read_IM_answer	Storage area for information and maintenance data
DB	read_IM_answer	Storage area for answer
DB	read_single_Param_answer	Storage area for answer
DB	read_single_Param_order	Storage area for question
FB	JAT-FB_PA	Data exchange via cyclically updated process image (variant A)
FB	JAT-FB_SFC	Data exchange via system functions with direct peripheral access (variant B)
FB	JAT-Axis_Tele9	Realization of a simple positioner with set selection and MDI via telegram 9
FB	JAT-Read_IM	Reading information and maintenance data
FB	JAT-RDWR_PAP	Acyclic reading / writing of parameters, texts and descriptions
FB	JAT-RDWR_PAP_DP	Acyclic reading / writing of parameters, texts and descriptions
FB	PZDtoAxisDB_SFC14	Reading the process data from the drive (system function)
FB	AxisDBtoPZD_SFC15	Writing process data into the drive (system function)
FC	PZDtoAxisDB	Reading the axis data from the process image
FC	AxisDBtoPZD	Writing the axis data to the process image
UDT	UDT_Axis_Control_IN	Data type image axis status input
UDT	UDT_Axis_Status_OUT	Data type image axis status output
UDT	UDT_PAE	Data type Process data axis input
UDT	UDT_PAA	Data type Process data axis output

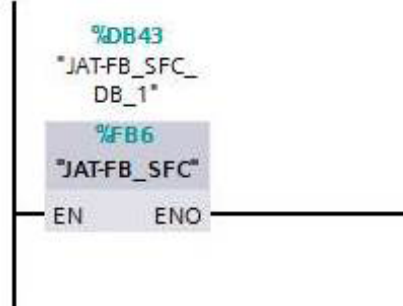
Terms:

- OB **Organisation block**
- DB **Data block**
- FB **Function block**
- FC **Function**
- UDT **User defined data type**

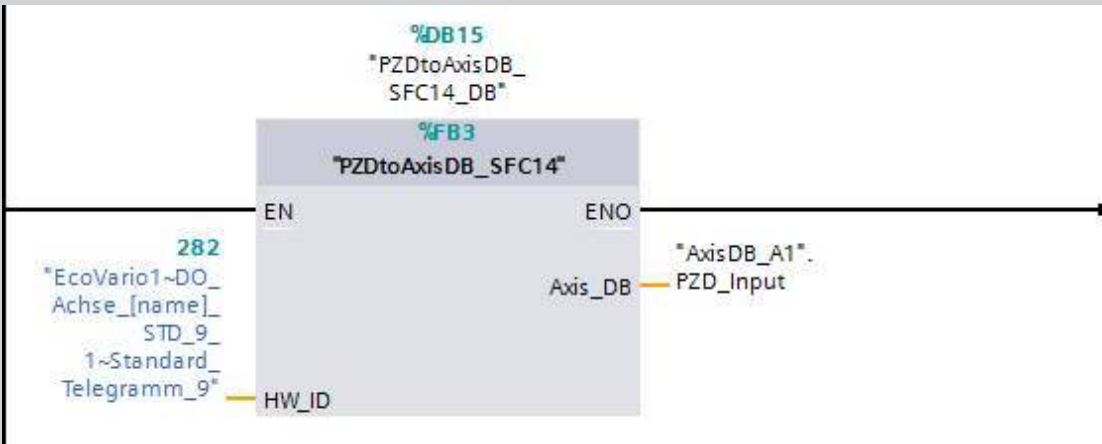
6.3 JAT-FB_PA

Name	JAT-FB_PA	
Type	Function block	
Input parameter	Data type	Comment
Output parameter	Data type	Comment
Graphical representation		
		
Functional description	Data exchange via cyclically updated process image (variant A)	

6.4 JAT-FB_SFC

Name	JAT-FB_SFC	
Type	Function block	
Input parameter	Data type	Comment
Output parameter	Data type	Comment
Graphical representation		
		
Functional description	Data exchange via system functions with direct peripheral access (variant B)	

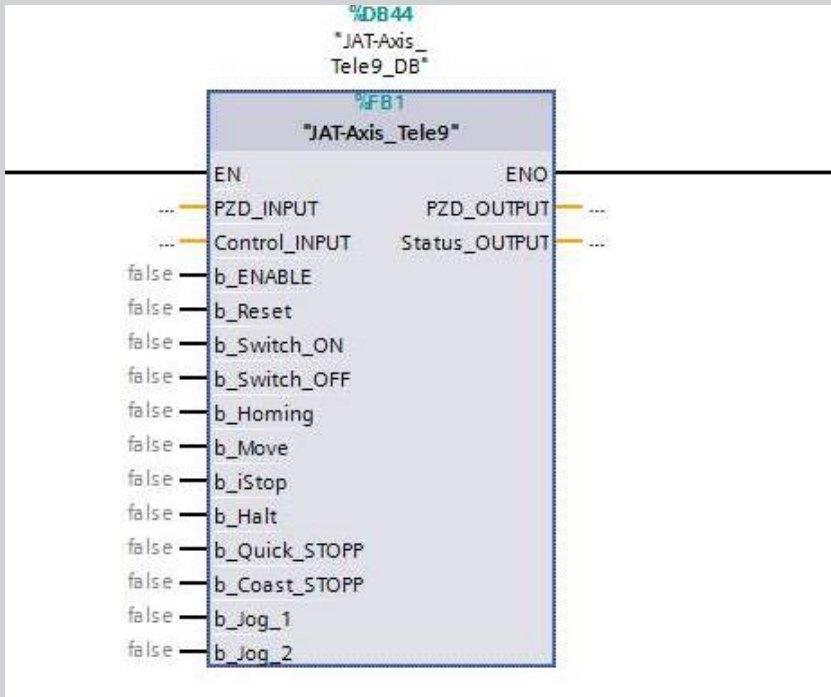
6.5 PZDtoAxisDB_SFC14

Name	PZDtoAxisDB_SFC14	
Type	Function block	
Input parameter		
HW_ID	HW_IO	„EcoVario-Head“
Output parameter		
Axis_DB_A1	UDT_PAE	to „Axis_DB_Ax“ for axis no x
Graphical representation		
		
Functional description	Reading the process data from the drive (system function)	

6.6 AxisDBtoPZD_SFC15

Name	AxisDBtoPZD_SFC15	
Type	Function block	
Input parameter	Data type	Comment
HW_ID	HW_IO	„EcoVario-Head“
Axis_DB_A1	UDT_PAA	from „Axis_DB_Ax“ for axis no x
Output parameter	Data type	Comment
Graphical representation		
Functional description	Writing process data into the drive (system function)	

6.7 JAT-Axis_Tele9

Name	JAT-Axis_Tele9	
Type	Function block	
Input parameter	Data type	Comment
PZD_INPUT	UDT_PAE	read axis process data
Control_INPUT	UDT_Axis_Control_IN	read image axis status
b_ENABLE	Bool	enable FB state machine
b_Reset	Bool	reset drive
b_SwitchON	Bool	switch on drive
b_SwitchOFF	Bool	switch off drive
b_Homing	Bool	run homing
b_Move	Bool	move drive (MDI/PSI)
b_iStop	Bool	intermediate stop (if axis move)
b_Halt	Bool	stop actual move
b_Quick_STOPP	Bool	quick stop
b_Coast_STOPP	Bool	coast stop
b_Jog_1	Bool	Jog 1
b_Jog_2	Bool	Jog 2
Output parameter	Data type	Comment
PZD_OUTPUT	UDT_PAA	axis process data output
Status_OUTPUT	UDT_Axis_Status_OUT	image axis control output
Graphical representation		
		
Functional description	Realization of a simple positioner with set selection and MDI via telegram 9	

6.7.2 Extended Functional Description of the Input Signals

Signal	Description	Control
b_ENABLE	Switching the internal state machine of the FB on or of	Static high
b_Reset	Reset of the servo amplifier (power stage is switched off)	Edge
b_SwitchON	Switch on power stage	Edge
b_SwitchOFF	Switch off power stage	Edge
b_Homing	Execution of homing depending on the setting in the servo amplifier	Edge
b_Move	Execution of the positioning record (for MDI w_SATZANW = 8000hex) Note: New target position is taken over by b_Move (flank) at any time	Edge
b_iStop	Stopover of the current positioning record	Edge
b_Halt	Stop the axis and cancel the current positioning record Note: also necessary in the event of a fault if the positioning record has not been completed	Edge
b_Quick_STOPP	Quick stop	Edge
b_Coast_STOPP	Coast stop	Edge
b_Jog_1	Jog Mode (Speed is parameterized in the controller Jog1)	Button
b_Jog_2	Jog Mode (Speed is parameterized in the controller Jog2)	Button

6.7.1 AxisDB_A1

Name	Data type	Start value	Observation value	Retention	Reachable	Visible	Settings	Comment
Static								
PZD_Input	UDT_PAE			False	True	True	False	
w_PZD1_ZSW1	Word	16#0	16#0248	False	True	True	False	
w_PZD2_AKSATZ	Word	16#0	16#0000	False	True	True	False	
w_PZD3_ZSW2	Word	16#0	16#0000	False	True	True	False	
dw_PZD4_XIST_A	DWord	16#0	16#FFFC_38C5	False	True	True	False	
Control_In	UDT_Axis_Control_IN			False	True	True	False	
w_SATZANW	Word	16#8000	16#8000	False	True	True	False	16#8000 for MDI
b_multiMDIactiv	Bool	0	TRUE	False	True	True	False	set multi MDI Mode active (if TRUE =new MDIPos or MDIvelo change on the fly in MDI Mode)
r_MDI_TARPOS	Real	90.0	275.0	False	True	True	False	in ° or mm
r_MDI_VELOCITY	Real	1.0	1.0	False	False	False	False	in % (0-199.99%)
w_MDI_MOD	Word	16#0001	16#0001	False	True	True	False	0= rel. or 1 = abs.
b_Inkgeber	Bool	0	FALSE	False	True	True	False	1 = with Incremental Encoder
r_mechanic	Real	2.912.711	2.912.711	False	True	True	False	inc/° or inc/mm
r_MDI_ACC	Real	100.0	100.0	False	True	True	False	in % (0-199.99%)
r_MDI_DEC	Real	100.0	100.0	False	True	True	False	in % (0-199.99%)
b_Modulo	Bool	TRUE	TRUE	False	True	True	False	Modulo Axis (switch on Modulo Position Output)
dw_Modulo_mask	DWord	16#000FFFFF	16#000F_FFFF	False	True	True	False	2^n x i - 1 (mit n= ST resolution encoder i= gearbox 4009:05) / sample 2^17 x s^(4-1) - 1 = (131072 * 8)-1 = 1048575dec = 000F FFFFhex
Status_Out	UDT_Axis_Status_OUT			False	True	True	False	
b_RUN	Bool	false	FALSE	False	True	True	False	feedback from FB statemaschine
b_DONE	Bool	false	TRUE	False	True	True	False	feedback from FB statemaschine
b_ERROR	Bool	false	FALSE	False	True	True	False	feedback from FB statemaschine
s_step	String[20]	"	'Move done '	False	True	True	False	step statemaschine
w_AKSATZ	Word	16#0	16#0000	False	True	True	False	feedback MDI/PSI 16#8000 for MDI
w_ZSW1	Word	16#0	16#0248	False	True	True	False	Statusword
di_XIST_Position	DInt	0	-247611	False	True	True	False	current position as DINT
r_XIST_Position	Real	0.0	-850.105	False	True	True	False	current position as REAL
b_00_Ready_to_switch_on	Bool	false	FALSE	False	True	True	False	bit from Statusword (ZSW1)
b_01_Ready_to_operate	Bool	false	FALSE	False	True	True	False	bit from Statusword (ZSW1)
b_02_Operation_enable	Bool	false	FALSE	False	True	True	False	bit from Statusword (ZSW1)
b_03_Fault	Bool	false	TRUE	False	True	True	False	bit from Statusword (ZSW1)
b_04_Coast_stop_not_activated	Bool	false	FALSE	False	True	True	False	bit from Statusword (ZSW1)
b_05_Quick_stop_not_activated	Bool	false	FALSE	False	True	True	False	bit from Statusword (ZSW1)
b_06_Switch_on_disable	Bool	false	TRUE	False	True	True	False	bit from Statusword (ZSW1)
b_07_Warning	Bool	false	FALSE	False	True	True	False	bit from Statusword (ZSW1)
b_08_Following_error_within_tolerance_range	Bool	false	FALSE	False	True	True	False	bit from Statusword (ZSW1)
b_09_Control_by_PLC_requested	Bool	false	TRUE	False	True	True	False	bit from Statusword (ZSW1)
b_10_Target_position_reached	Bool	false	FALSE	False	True	True	False	bit from Statusword (ZSW1)
b_11_Reference_found	Bool	false	FALSE	False	True	True	False	bit from Statusword (ZSW1)
b_12_Traversing_task_acknowledgement	Bool	false	FALSE	False	True	True	False	bit from Statusword (ZSW1)
b_13_Drive_stopped	Bool	false	FALSE	False	True	True	False	bit from Statusword (ZSW1)
b_14_Commutation_found	Bool	false	FALSE	False	True	True	False	bit from Statusword (ZSW1)
b_15_Internal_limit_active	Bool	false	FALSE	False	True	True	False	bit from Statusword (ZSW1)
Error	Word	16#0		False	False	False	False	from GET_ER_ID() local Error handing
r_XIST_Position_Modulo	Real	0.0		False	True	True	False	current modulo position as REAL
PZD_Output	UDT_PAA			False	True	True	False	
w_PZD1_STW1	Word	16#0		False	True	True	False	
w_PZD2_SATZANW	Word	16#0		False	True	True	False	
w_PZD3_STW2	Word	16#0		False	True	True	False	
dw_PZD4_MDI_TARPOS	DWord	16#0		False	True	True	False	
dw_PZD5_MDI_VELOCITY	DWord	16#0		False	True	True	False	
w_PZD6_MDI_ACC	Word	16#0		False	True	True	False	
w_PZD7_MDI_DEC	Word	16#0		False	True	True	False	
w_PZD8_MDI_MOD	Word	16#0		False	True	True	False	

Storage area for axis image

6.8 JAT-Read_IM

Name	JAT-Read_IM	
Type	Function block	
Input parameter	Data type	Comment
SwitchRDREC	Bool	Start trigger
HW_ID	HW_IO	„EcoVario-Head“
ui_LEN_Antwort_DB	UInt	Lenght of string
Record_Antwort	Variant	read_IM_answer_DB
Output parameter	Data type	Comment
b_RUN_RD_o	Bool	feedback from FB statemaschine
b_DONE_RD_o	Bool	feedback from FB statemaschine
b_ERROR_RD_o	Bool	feedback from FB statemaschine
w_ERROR_ID_o	Word	ERROR-Code
Graphical representation		
Functional description	Reading information and maintenance data	

Contained in the sample project in one instance.

6.8.1 DB read_IM_answer

read_IM_answer								
	Name	Datentyp	Startwert	Remanenz	Erreichbar a...	Sichtbar i...	Einstellwert	Kommentar
1	▼ Static			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
2	▼ Antwort	Struct		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
3	Byte_0	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
4	Byte_1	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
5	Byte_2	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
6	Byte_3	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
7	Byte_4	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
8	Byte_5	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
9	Byte_6	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
10	Byte_7	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
11	Byte_8	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
12	Byte_9	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
13	Byte_10	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
14	Byte_11	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
15	Byte_12	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
16	Byte_13	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
17	Byte_14	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
18	Byte_15	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
19	Byte_16	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
20	Byte_17	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
21	Byte_18	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
22	Byte_19	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
23	Byte_20	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
24	Byte_21	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
25	Byte_22	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
26	Byte_23	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
27	Byte_24	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
28	Byte_25	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
29	Byte_26	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
30	Byte_27	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
31	Byte_28	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
32	Byte_29	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
33	Byte_30	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
34	Byte_31	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
35	Byte_32	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
36	Byte_33	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
37	Byte_34	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
38	Byte_35	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
39	Byte_36	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
40	Byte_37	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
41	Byte_38	Byte	16#0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

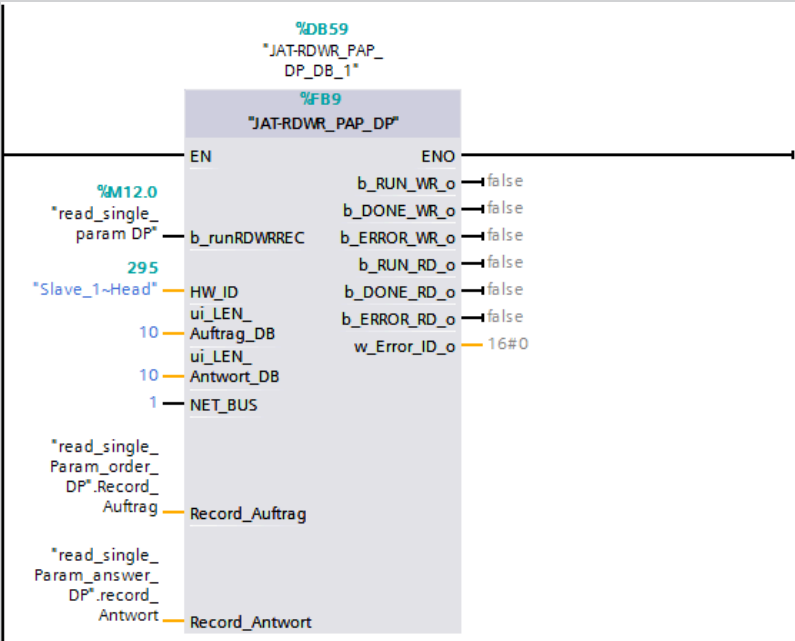
Storage area for the answer

6.9 JAT-RDWR_PAP

Name	JAT-RDWR_PAP	
Type	Function block for PROFINET (new Version 0.2: changed arrangement of the parameters Record_Auftrag and Record_Antwort under In/Out parameter) <i>Note:</i> For PROFIBUS use the function block JAT_RDWR_PAP_DP, see chapter 6.10	
Input parameter	Data type	Comment
b_runRDWRREC	Bool	Start trigger
HW_ID	HW_IO	~PAP for PROFINET
ui_LEN_Auftrag_DB	UInt	Length of string
ui_LEN_Antwort_DB	UInt	Length of string
Output parameter	Data type	Comment
b_RUN_WR_o	Bool	feedback from FB statemaschine
b_DONE_WR_o	Bool	feedback from FB statemaschine
b_ERROR_WR_o	Bool	feedback from FB statemaschine
b_RUN_RD_o	Bool	feedback from FB statemaschine
b_DONE_RD_o	Bool	feedback from FB statemaschine
b_ERROR_RD_o	Bool	feedback from FB statemaschine
w_ERROR_ID_o	Word	ERROR-Code
In/Out parameter	Data type	Comment
Record_Auftrag	Variant	read_single_Param_order_DB
Record_Antwort	Variant	read_single_Param_answer_DB
Graphical representation		
Functional description	Acyclic reading / writing of parameters, texts and descriptions. Single and multiple parameter accesses are demonstrated via different data blocks.	

Contained in the sample project in seven instances for different types of access.

6.10 JAT-RDWR_PAP_DP

Name	JAT-RDWR_PAP_DP	
Type	Function block for PROFINET/PROFIBUS (see example projects: YYYYMMDD _S7-1500_JAT-FB_RDWRREC_Tele9_plusProfibus_Tele1+2_V15_1.zap15_1)	
Input parameter		
b_runRDWRREC	Bool	Starttrigger
HW_ID	HW_IO	~PAP for PROFINET or ~Head for PROFIBUS
ui_LEN_Auftrag_DB	UInt	Length of string
ui_LEN_Antwort_DB	UInt	Length of string
NET_BUS	Bool	Switch for PROFIBUS = true or PROFINET = false
Output parameter		
b_RUN_WR_o	Bool	feedback from FB state machine
b_DONE_WR_o	Bool	feedback from FB state machine
b_ERROR_WR_o	Bool	feedback from FB state machine
b_RUN_RD_o	Bool	feedback from FB state machine
b_DONE_RD_o	Bool	feedback from FB state machine
b_ERROR_RD_o	Bool	feedback from FB state machine
w_ERROR_ID_o	Word	ERROR-Code
In/Out parameter		
Record_Auftrag	Variant	read_single_Param_order_DB
Record_Antwort	Variant	read_single_Param_answer_DB
Graphical representation		
 <p>The diagram shows the graphical representation of the JAT-RDWR_PAP_DP function block. It is a function block (FB9) with the following connections:</p> <ul style="list-style-type: none"> Inputs: EN, ENO, b_runRDWRREC (Bool), HW_ID (HW_IO), ui_LEN_Auftrag_DB (UInt, value 10), ui_LEN_Antwort_DB (UInt, value 10), NET_BUS (Bool, value 1), Record_Auftrag (Variant), Record_Antwort (Variant). Outputs: b_RUN_WR_o (Bool, value false), b_DONE_WR_o (Bool, value false), b_ERROR_WR_o (Bool, value false), b_RUN_RD_o (Bool, value false), b_DONE_RD_o (Bool, value false), b_ERROR_RD_o (Bool, value false), w_Error_ID_o (Word, value 16#0). Data Blocks: read_single_Param_order_DP (Variant), read_single_Param_answer_DP (Variant). 		
Functional description	Acyclic reading / writing of parameters, texts and descriptions. Single and multiple parameter accesses are demonstrated via different data blocks.	

The „NET_BUS“ bit is used to switch between the two types of access Local (for PROFIBUS) or Global (for PROFINET).

For PROFIBUS devices only:

The system constant for ~Head 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):

- ☞ Axis => value = 01 for axis A
- ☞ Axis => value = 02 for axis B

...9_plusProfibus_V15_1 ▶ Maschine_1 ▶ PLC_1 [CPU 1511-1 PN] ▶ Programmbausteine ▶ read_single_Param_order_DP [DB6

Aktualwerte behalten Momentaufnahme Momentaufnahmen in Startwerte kopieren

read_single_Param_order_DP

Name	Datentyp	Startwert	Beobachtungswert	Remanenz	Erreichbar a..	Schrei...	Sichtbar i...	Ei
1	Static							
2	Record_Auftrag	Struct			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3	Anforderungsrefer...	Byte	16#03	16#03	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4	Anforderungs_ID	Byte	16#01	16#01	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
5	Achse	Byte	16#01	16#01	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
6	Anzahl_der_Param...	Byte	16#01	16#01	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
7	Attribut_01	Byte	B#16#10	16#10	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
8	Anzahl_der_Eleme...	Byte	16#01	16#01	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
9	Parameter_Numme...	Word	16#2100	16#2100	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
10	Subindex_01	Word	16#0000	16#0000	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

6.10 PZDtoAxisDB

Name	PZDtoAxisDB	
Type	Function	
Input parameter	Data type	Comment
PAE_PZD1_ZSW1	Word	ZSW1 from Drive (from variable table)
PAE_PZD2_AKSATZ	Word	AKSATZ from Drive (from variable table)
PAE_PZD3_ZSW2	Word	ZSW2 from Drive (from variable table)
PAE_PZD3_XIST_A	DWord	XIST_A from Drive (from variable table)
Output parameter	Data type	Comment
Axis_DB	UDT_PAE	to „Axis_DB_Ax“ for axis no x
Graphical representation		
Functional description	Reading the process data from the drive	

6.11 AxisDBtoPZD

Name	AxisDBtoPZD	
Type	Function	
<hr/>		
Input parameter	Data type	Comment
Axis_DB	UDT_PAA	from „Axis_DB_Ax“ for axis no x
<hr/>		
Output parameter	Data type	Comment
A1_PAA_PZD1_STW1	Word	STW1 to Drive (from variable table)
A1_PAA_PZD2_SATZANW	Word	SATZANW to Drive (from variable table)
A1_PAA_PZD3_STW2	Word	STW2 to Drive (from variable table)
A1_PAA_PZD4_MDI_TARPOS	Real	TARPOS to Drive (from variable table)
A1_PAA_PZD5_MDI_VELOCITY	Real	VELOCITY to Drive (from variable table)
A1_PAA_PZD6_MDI_ACC%	Real	ACC to Drive (from variable table)
A1_PAA_PZD7_MDI_DEC%	Real	DEC to Drive (from variable table)
A1_PAA_PZD8_MDI_MOD	Word	MDI_MOD to Drive (from variable table)
<hr/>		
Graphical representation		
Functional description	Writing the process data into the drive	

7 User-defined Data Types

7.1 UDT_PAE

UDT_PAE							
	Name	Datentyp	Defaultwert	Erreichbar a...	Sichtbar i...	Einstellwert	Kommentar
1	w_PZD1_ZSW1	Word	16#0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
2	w_PZD2_AKSATZ	Word	16#0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
3	w_PZD3_ZSW2	Word	16#0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
4	dw_PZD4_XIST_A	DWord	16#0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Process data axis input

7.2 UDT_PAA

UDT_PAA							
	Name	Datentyp	Defaultwert	Erreichbar a...	Sichtbar i...	Einstellwert	Kommentar
1	w_PZD1_STW1	Word	16#0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
2	w_PZD2_SATZANW	Word	16#0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
3	w_PZD3_STW2	Word	16#0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
4	dw_PZD4_MDI_TARPOS	DWord	16#0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
5	dw_PZD5_MDI_VELOCITY	DWord	16#0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
6	w_PZD6_MDI_ACC	Word	16#0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
7	w_PZD7_MDI_DEC	Word	16#0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
8	w_PZD8_MDI_MOD	Word	16#0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Process data axis output

7.3 UDT_Axis_Control_IN

UDT_Axis_Control_IN							
	Name	Datentyp	Defaultwert	Erreichbar a...	Sichtbar i...	Einstellwert	Kommentar
1	w_SATZANW	Word	16#0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	16#8000 for MDI
2	b_multiMDIactiv	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	set multi MDI Mode active (if TRUE =new MDIPos or MDIVelo change on the fly in MDI Mode)
3	r_MDI_TARPOS	Real	0.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	in ° or mm
4	r_MDI_VELOCITY	Real	0.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	in % (0-199.99%)
5	w_MDI_MOD	Word	16#0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	0= rel. or 1 = abs.
6	b_Inkgeber	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	1 = with Incremental Encoder
7	r_mechanic	Real	1.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	inc# or inc/mm
8	r_MDI_ACC	Real	0.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	in % (0-199.99%)
9	r_MDI_DEC	Real	0.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	in % (0-199.99%)
10	b_Modulo	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Modulo Axis (switch on Modulo Position Output)
11	dw_Modulo_mask	DWord	16#0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	*2^n xi - 1* (mit n= ST resolution encoder in gearboxus 4009.05) / sample 2^17 xs^(4-1) - 1 = (131072 * 8)-1 = 1048575dec = 000FFFFhex

Image axis status input

7.4 UDT_Axis_Status_OUT



























UDT_Axis_Status_OUT							
	Name	Datentyp	Defaultwert	Erreichbar a...	Sichtbar i...	Einstellwert	Kommentar
1	 b_RUN	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	feedback from FB state maschine
2	 b_DONE	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	feedback from FB state maschine
3	 b_ERROR	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	feedback from FB state maschine
4	 s_step	String[20]	"	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	step state maschine
5	 w_AKSATZ	Word	16#0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	feedback MDI/PSI 16#8000 for MDI
6	 w_ZSW1	Word	16#0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Statusword
7	 di_XIST_Position	DInt	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	current position as DINT
8	 r_XIST_Position	Real	0.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	current position as REAL
9	 b_00_Ready_to_switch_on	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	bit from Statusword (ZSW1)
10	 b_01_Ready_to_operate	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	bit from Statusword (ZSW1)
11	 b_02_Operation_enable	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	bit from Statusword (ZSW1)
12	 b_03_Fault	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	bit from Statusword (ZSW1)
13	 b_04_Coast_stop_not_activa...	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	bit from Statusword (ZSW1)
14	 b_05_Quick_stop_not_activa...	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	bit from Statusword (ZSW1)
15	 b_06_Switch_on_disable	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	bit from Statusword (ZSW1)
16	 b_07_Warning	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	bit from Statusword (ZSW1)
17	 b_08_Following_error_within...	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	bit from Statusword (ZSW1)
18	 b_09_Control_by_PLC_reque...	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	bit from Statusword (ZSW1)
19	 b_10_Target_potion_reached	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	bit from Statusword (ZSW1)
20	 b_11_Reference_found	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	bit from Statusword (ZSW1)
21	 b_12_Traversing_task_ackno...	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	bit from Statusword (ZSW1)
22	 b_13_Drive_stopped	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	bit from Statusword (ZSW1)
23	 b_14_Commutation_found	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	bit from Statusword (ZSW1)
24	 b_15_Internal_limit_active	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	bit from Statusword (ZSW1)
25	 Error	Word	16#0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	from GET_ER_ID() local Error handing
26	 r_XIST_Position_Modulo	Real	0.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	current modulo position as REAL

Image axis status ouput

8 Acyclic Reading and Writing of Parameters

8.1 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.

In the function block package „FB PROFINET“ the blocks „[JAT-RDWR_PAP](#)“ (for PROFINET devices) and „[JAT-RDWR_PAP_DP](#)“ (for PROFIBUS and PROFINET devices) are defined for this purpose. Structure and parameter setting are described in chapter 6.9 and 6.10 of this document.

Detailed information can be found in the 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 Objects are supported:

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

8.2 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 RDREC mit der Antwort gefüllt.

8.3 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 response²

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 Response (+) 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 27) 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 28)	

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	Access 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.

8.4 Reading a single parameter value

8.4.1 DB read_single_Param_order

read_single_Param_order									
	Name	Datentyp	Startwert	Remanenz	Erreichbar a..	Sichtbar i...	Einstellwert	Kommentar	
1	Static								
2	Record_Auftrag	Struct			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
3	Anforderungsrefer...	Byte	16#03		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Nummer der Aufträge
4	Anforderungs_ID	Byte	16#01		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Parameter lesen = 1; Parameter schreib
5	Achse	Byte	16#01		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Adressierung einer Achse / Multi-Ach...
6	Anzahl_der_Param...	Byte	16#01		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			lesen der Parameter r0965
7	Attribut_01	Byte	B#16#10		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Typ des Objektes: 10 Wert, 20 Beschr...
8	Anzahl_der_Eleme...	Byte	16#00		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Anzahl der Indizes 1
9	Parameter_Numme...	Word	965		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Parameter r0965[1] (965 dez= 3C5 h...
10	Subindex_01	Word	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Subindex

Request DB

8.4.2 DB read_single_Param_answer

read_single_Param_answer									
	Name	Datentyp	Startwert	Remanenz	Erreichbar a..	Sichtbar i...	Einstellwert	Kommentar	
1	Static								
2	record_Antwort	Struct			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
3	Anf_referenz_gesp...	Byte	16#0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Nummer des Auftrag gespiegelt
4	Antwort_ID	Byte	16#0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Parameter lesen
5	Achse_gespiegelt	Byte	16#0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Adressierung der Achse gespiegelt
6	Anz_Parameter	Byte	16#0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Anforderung auf die Anzahl der Para...
7	Format_Parameter...	Byte	16#0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Anforderung auf das Format des Para...
8	Anz_Parameterver...	Byte	16#0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			Anforderung auf die Anzahl der Wert...
9	Param1_Wert_01	Word	16#0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Wert aus Index 0

Response DB

8.4.3 Example for reading a single parameter value

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

Auftrag_DB_single									
	Name	Datentyp	Offset	Startwert	Remanenz	Erreichbar aus HMI	Sichtbar in HMI	Einstellwert	Kommentar
1	Static								
2	Record_Auftrag	Struct	0.0			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
3	Anforderungsreferenz	Byte	0.0	16#03		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Nummer der Aufträge
4	Anforderungs_ID	Byte	1.0	16#01		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Parameter lesen = 1; Parameter schreiben = 2
5	Achse	Byte	2.0	16#01		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Adressierung einer Achse / Multi-Achsenantrieb
6	Anzahl_der_Parameter	Byte	3.0	16#01		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		lesen der Parameter r0965
7	Attribut_01	Byte	4.0	B#16#10		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Typ des Objektes: 10 Wert, 20 Beschreibung, 30 Text
8	Anzahl_der_Elemente_01	Byte	5.0	16#00		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Anzahl der Indizes 1
9	Parameter_Nummer_01	Word	6.0	965		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Parameter r0965[1] (965 dez= 3C5 hex) nur 1
10	Subindex_01	Word	8.0	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Subindex

Request DB (RECORD)

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

Response DB (RECORD)

8.5 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

Auftrag_DB_multi									
	Name	Datentyp	Offset	Startwert	Remanenz	Erreichbar a...	Sichtbar i...	Einstellwert	Kommentar
1	Static								
2	Record_Auftrag	Struct	0.0			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
3	Anforderungsreferenz	Byte	0.0	16#1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Nummer der Aufträge
4	Anforderungs_ID	Byte	1.0	16#1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Parameter lesen = 1; Parameter schreiben = 2
5	Achse	Byte	2.0	16#1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Adressierung einer Achse / Multi-Achsenantrieb
6	Anzahl_der_Parameter	Byte	3.0	16#4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		lesen der Parameter 965[1], 930[61], 2714[1] u. 5100
7	Attribut_01	Byte	4.0	16#10		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Typ des Objektes: 10 Wert, 20 Beschreibung, 30 Text
8	Anzahl_der_Elements_01	Byte	5.0	16#0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Anzahl der Indizes 1
9	Parameter_Nummer_01	Word	6.0	16#3c5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Parameter 965[1] - Profilidentifikation (965 dez= 3C5 hex) nur 1
10	Subindex_01	Word	8.0	16#0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Subindex
11	Attribut_02	Byte	10.0	16#10		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Typ des Objektes: 10 Wert, 20 Beschreibung, 30 Text
12	Anzahl_der_Elements_02	Byte	11.0	16#0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Anzahl der Indizes 1
13	Parameter_Nummer_02	Word	12.0	16#2714		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Parameter 2714[1] - SingleStepVelo (=CANObjekt 270207) nur 1
14	Subindex_02	Word	14.0	16#0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Subindex
15	Attribut_03	Byte	16.0	16#10		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Typ des Objektes: 10 Wert, 20 Beschreibung, 30 Text
16	Anzahl_der_Elements_03	Byte	17.0	16#0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Anzahl der Indizes 1
17	Parameter_Nummer_03	Word	18.0	16#3A2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Parameter 930[1] - Betriebsart (930 dez= 3A2 hex) nur 1
18	Subindex_03	Word	20.0	16#0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Subindex
19	Attribut_04	Byte	22.0	16#10		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Typ des Objektes: 10 Wert, 20 Beschreibung, 30 Text
20	Anzahl_der_Elements_04	Byte	23.0	16#0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Anzahl der Indizes 1
21	Parameter_Nummer_04	Word	24.0	16#5100		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Parameter 5100[1] - Velo_Norm_Para nur 1
22	Subindex_04	Word	26.0	16#0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Subindex

Request DB (RECORD)

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

Response DB (RECORD)

8.6 Writing a single or several parameter values

In the example the value of parameter 5100 (normalization speed) is set to 0x3E8 (decimal: 1000) rpm. The number of values to be written is specified with 0x01 in the field „number_of_parameters“.

Auftrag_DB_single_write									
	Name	Datentyp	Offset	Startwert	Remanenz	Erreichbar a...	Sichtbar i...	Einstellwert	Kommentar
1	Static								
2	Record_Auftrag	Struct	0.0			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
3	Anforderungsreferenz	Byte	0.0	16#04		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Nummer der Aufträge
4	Anforderungs_ID	Byte	1.0	16#02		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Parameter lesen = 1; Parameter schreiben = 2
5	Achse	Byte	2.0	16#01		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Adressierung einer Achse / Multi-Achsenantrieb
6	Anzahl_der_Parameter	Byte	3.0	16#01		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		schreiben 5100
7	Attribut_01	Byte	4.0	16#10		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Typ des Objektes: 10 Wert, 20 Beschreibung, 30 Text
8	Anzahl_der_Elements_01	Byte	5.0	16#00		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Anzahl der Indizes 1
9	Parameter_Nummer_01	Word	6.0	16#5100		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Parameter 5100[1] Normierungsgeschwindigkeit
10	Subindex_01	Word	8.0	16#00		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Subindex
11	Data_type	Byte	10.0	16#03		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Daten Typ (x03 = integer 16)
12	No_of_values	Byte	11.0	16#01		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Number of values = Number of elements
13	Value	Word	12.0	16#03e8		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Wert x3E8 auf 5100

Request DB (RECORD)

Response DB (RECORD)

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

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.

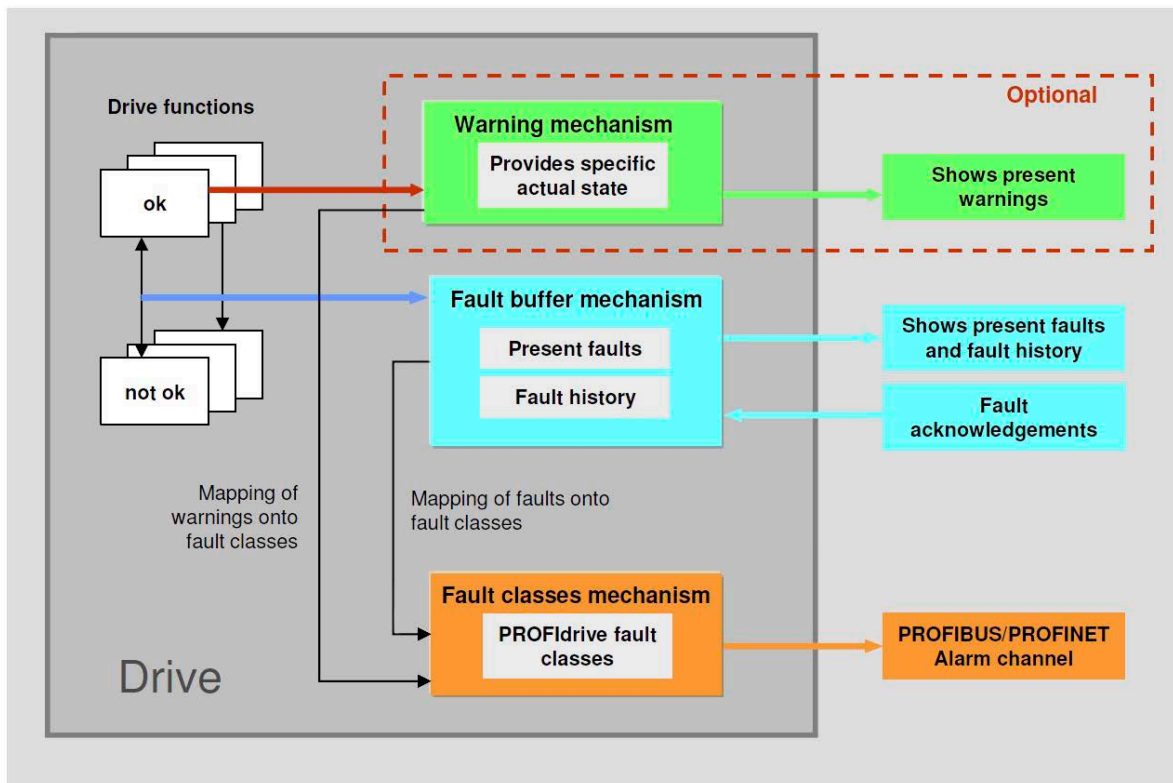
In principle, several warnings can be present at the same time (for example, „Increased temperature of the motor winding“ and „DC link voltage too low“). Warnings, unlike errors, do not stop the drive.

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

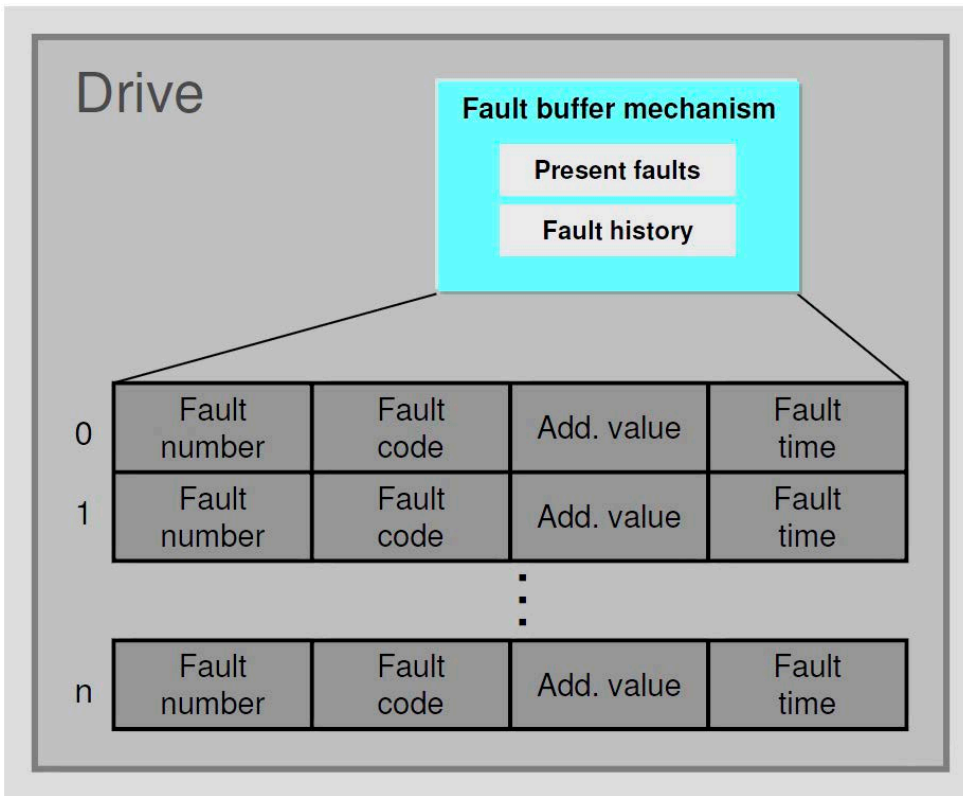
A fault condition in the drive (e.g., overtemperature) always triggers a device specific response, i. As a rule, the drive is switched off. At the same time, one or more fault messages characterizing the fault state are entered in the fault buffer (see figure below).

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 (Bild 14). 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

The screenshot shows the 'Benutzerdefinierte Variablen' (User Defined Variables) window in the ECOSTUDIO software. The window title is 'Analyse/Benutzerdefinierte Variablen'. The interface includes a menu bar with 'Benutzermaske 1' through 'Benutzermaske 5', 'Unimapper/IP', 'Bach'sche Z-Achse', and 'ds_402_faults x26'. The main area contains a table of variables:

Variable	Description	Value
53B300	aktuell: Fehlernummer 00	0022 hex
53B301	aktuell: Fehlernummer 01	0016 hex
53B302	aktuell: Fehlernummer 02	0017 hex
53B217		0B04 hex
53B216		0B03 hex
53B222		0D13 hex
53B308	Historie 00: Fehlernummer 00	0022 hex
53B309	Historie 00: Fehlernummer 01	0017 hex
53B30A	Historie 00: Fehlernummer 02	0 dec
53B30B	Historie 00: Fehlernummer 03	0 dec
53B100		0D13 hex
53B101		0B03 hex
53B102		0B04 hex
53B10E		0D13 hex
~		

Annotations on the right side of the table use brackets to group variables:

- 947, 00-07 actual: points to variables 53B300-53B302.
- 946, plus sub index from 947: points to variables 53B217-53B222.
- 947, 08-63 history: points to variables 53B308-53B30B.
- 945, 00-07 actual: points to variables 53B100-53B102.
- 945, 08-63 history: points to variable 53B10E.

Fault parameters 945, 946 and 947 (ECOSTUDIO® User Defined Variables)

➤ 947 (x53B3 in ECOSTUDIO®)

Sub index 00-07 contains the actual „fault sub index“ for parameters 946 and 951
Sub index 08-63 contains the fault history (7x8 entries)

➤ 946 (x53B2 in ECOSTUDIO®)

cf. 947
here the actual display code can be seen

➤ 945 (x53B1 in ECOSTUDIO®)

Sub index 00-07 contains the actual display error code 3 for the current situation
Sub index 08-63 contains the fault history (7x8 entries)

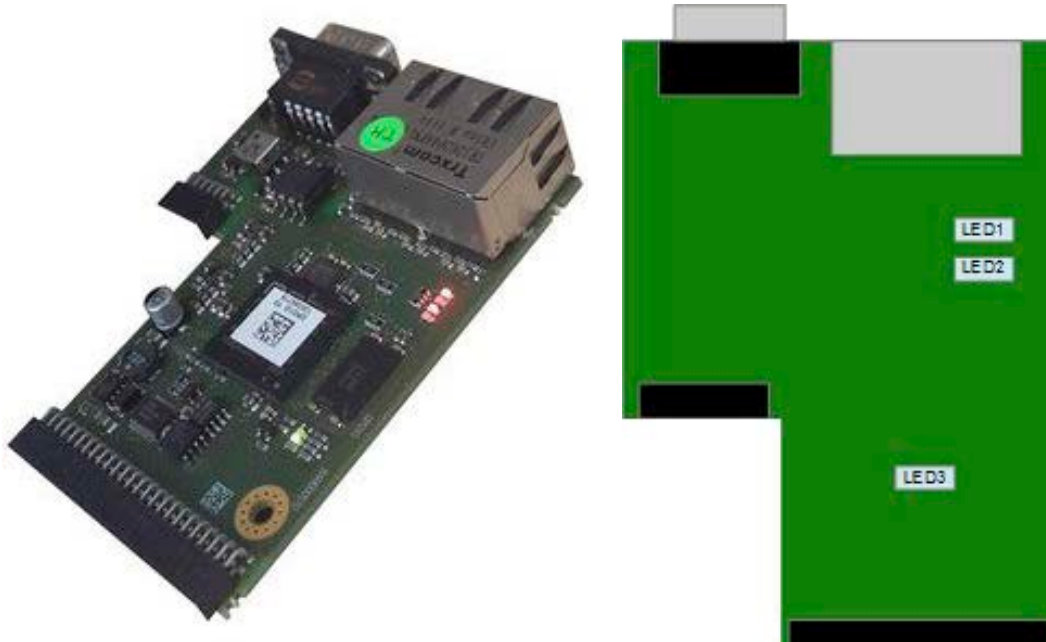
➤ 951 (x53B7 in ECOSTUDIO®)

cf. 947
- Read parameter 951 with argument „Text“ (30) = error text
- Read parameter 951 with argument „Value“ (10) = parameter number
(The parameter displayed here is related to the error message)

Note: the error texts are limited in their statement

10 Hardware Diagnosis

10.1 Operating display on the PROFINET option card



LED1: RUN

- OFF = There is an application relationship
- 1Hz red = Network share, card not yet configured
- ON red = No network share

LED2: Communication error

- ON red = Card error (communication error)

LED3: Option card status

- OFF = HW fault
- 1Hz yellow = No protocol stack and no second stage loader (HW OK)
- 1Hz yellow/green = Second stage loader OK, no protocol stack (HW OK)
- ON green = Everything OK

