HIWIN MIKROSYSTEM

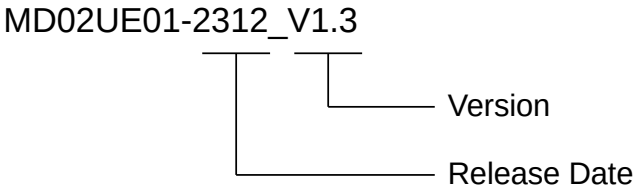
E1 Series Servo Drive

PROFINET Communication Command Manual

www.hiwinmikro.tw
MD02UE01-2312_V1.3

Revision History

The version of the manual is also indicated on the bottom of the front cover.



Release Date	Version	Applicable Product	Revision Contents
Dec. 11st, 2023	1.3	E1 PROFINET drive	1. Update section 3.1 **IO data signals**. 2. Update section 3.2 **Supported telegrams**. 3. Update section 3.3.2 **Telegram 9, Telegram 111**. 4. Update section 4.2 **PNU dictionary table**. 5. Update section 6.1 **Velocity reference value setting**. 6. Update section 6.2 **Velocity limit setting**. 7. Update section 6.3 **Torque limit setting**. 8. Update section 6.4 **Quick stop**. 9. Update section 6.6.1 **MDI setpoints**. 10. Update section 6.8 **JOG**. 11. Add section 6.9 **Absolute encoder initialization**. 12. Update section 7.1 **Configure PROFINET communication by Thunder**.
Jan. 19th, 2023	1.2	E1 PROFINET drive	1. Section 2.2 **Communication specification**: Revise cable length and supported telegram. 2. Section 3.5.1 **Status word 1 (ZSW1) - Telegram 3**: Revise bit description. 3. Section 6.1 **Velocity reference value setting**: Revise the related Pt parameters. 4. Section 6.8 **JOG**: Revise the related Pt parameters. 5. Section 7.1 **Configure PROFINET communication by Thunder**: Revise “PROFINET setup” window in Thunder and its description.
Dec. 30th, 2021	1.1	E1 PROFINET drive	1. Add the information of HIWIN Telegram 111. 2. Revise the unit description of MDI_VELOCITY. 3. Add the description of JOG.
Jun. 18th, 2021	1.0	E1 PROFINET drive	First edition.

Related Documents

Through related documents, users can quickly understand the positioning of this manual and the correlation between manuals and products. Go to HIWIN MIKROSYSTEM's official website → Download → Manual Overview for details (https://www.hiwinmikro.tw/Downloads/ManualOverview_EN.htm).

Table of Contents

(This page is intentionally left blank.)

1. About this manual

1.1 Preface

PROFINET (as a portmanteau for Process Field Net) is an industry technical standard for data communication over Industrial Ethernet. The standard is maintained and supported by PROFIBUS & PROFINET International (PI), an umbrella organization headquartered in Karlsruhe, Germany. This manual mainly describes PROFINET communication and PROFIdrive profile applied to E1 PROFINET drives. For a more complete understanding of E1 series servo drive, please refer to “E1 Series Servo Drive User Manual”.

1.2 Trademark

PROFINET ® is a registered trademark of PROFIBUS & PROFINET International (PI).

1.3 General precautions

Before using the product, please carefully read through this manual. HIWIN Mikrosystem (HIWIN) is not responsible for any damage, accident or injury caused by failure in following the installation instructions and operating instructions stated in this manual.

- Do not disassemble or modify the product. The design of the product has been verified by structural calculation, computer simulation and actual testing. HIWIN is not responsible for any damage, accident or injury caused by disassembly or modification done by users.
- Before installing or using the product, ensure there is no damage on its appearance. If any damage is found after inspection, please contact HIWIN or local distributors.
- Carefully read through the specification noted on product label or technical document. Install the product according to its specification and installation instructions stated in this manual.
- Ensure the product is used with power supply specified on product label or in product requirement. HIWIN is not responsible for any damage, accident or injury caused by incorrect power supply.
- Ensure the product is used with rated load. HIWIN is not responsible for any damage, accident or injury caused by improper usage.
- Do not subject the product to shock. HIWIN is not responsible for any damage, accident or injury caused by improper usage.
- If an error occurs in the drive, please refer to “E1 Series Servo Drive User Manual” and follow the instructions for troubleshooting. After the error is eliminated, power on the drive again.
- Do not repair the product by yourself when it malfunctions. The product can only be repaired by qualified technician from HIWIN.

HIWIN offers 1-year warranty for the product. The warranty does not cover damage caused by improper usage (refer to the precautions and instructions stated in this manual) or natural disaster.

1.4 Safety precautions

- Carefully read through this manual before installation, transportation, maintenance and examination. Ensure the product is correctly used.
- Carefully read through electromagnetic (EM) information, safety information and related precautions before usage.
- Safety precautions in this manual are classified into “Warning”, “Attention”, “Prohibited” and “Required”.

Signal Word	Description
Warning	It indicates if the precaution is not observed, it is likely to cause property loss, serious injury or death.
Attention	It indicates the precaution must be observed.
Prohibited	It indicates prohibited activity.
Required	It indicates mandatory activity.

DANGER

- Ensure the drive is correctly grounded. Use PE bar in the control cabinet as reference potential. Perform low-ohmic grounding for safety reason.
- Do not remove motor power cable from the drive when it is still power-on, or there is a risk of electric shock or damage to the contact.
- Do not touch the live part (contact or bolt) within 15 minutes after disconnecting the drive from power supply. For your own safety, we suggest measuring the voltage in the intermediate circuit and wait until it falls to 50 VDC before touching the live part.

■ Operation

Warning	◆ Do not touch the terminals and the internal part of the product when power on, or it may cause electric shock. ◆ Do not touch the terminals and internal part of the product within 15 minutes after power off, or the residual voltage may cause electric shock. ◆ Do not modify wiring when power on, or it may cause electric shock. ◆ Do not damage, apply excessive force to, place any heavy object on the cable or put the cable between two objects, or it may cause electric shock or fire.
Attention	◆ Do not use the product in location which is subject to humidity, corrosive materials, flammable gas or flammable materials.

■ Storage

Prohibited	◆ Do not store the product in location which is subject to water, water drop, direct sunlight, harmful gas or liquid.

■ Transportation

Attention	◆ Carefully move the product to avoid damage. ◆ Do not apply excessive force to the product. ◆ Do not stack the products to avoid collapse.

■ Installation site

Required	◆ Do not install the product in location with high ambient temperature and high humidity or location which is subject to dust, iron powder or cutting powder. ◆ Install the product in location with ambient temperature stated in the manual. Use cooling fan if the ambient temperature is too high. ◆ Do not install the product in location which is subject to direct sunlight. ◆ The product is not drip-proof or waterproof, so do not install or operate the product outdoor or in location which is subject to water or liquid. ◆ Install the product in location with less vibration. ◆ Motor generates heat when running for a period of time. Use cooling fan or disable the motor when it is not in use, so the ambient temperature will not exceed product specification.

■ Installation

Attention	◆ Do not place heavy object on the product, or it may cause injury. ◆ Prevent any foreign matter from entering the product, or it may cause fire. ◆ Install the product in the specified orientation, or it may cause fire. ◆ Avoid strong shock to the product, or it may cause malfunction or injury. ◆ When installing the product, take the product weight into consideration. Improper installation may cause damage. ◆ Install the product on noncombustible objects, such as metal to avoid fire.

■ Wiring

Attention	◆ Ensure wiring is correctly performed, or it may cause malfunction or burn. There is a risk of injury or fire. ◆ The peripheral devices, including controller, must share the same power supply system with the servo drive. Otherwise, the voltage difference between the devices and the servo drive could result in burn-out.

■ Operation and transportation

Attention	◆ Use power supply specified in product specification, or it may cause injury or fire. ◆ The product may suddenly start to operate after power supply recovers. Please do not get too close to the product.
Required	◆ Set external wiring for emergency stop to stop the motor at any time.

■ Maintenance

Prohibited	◆ Do not disassemble or modify the product. ◆ Do not repair the product by yourself when it malfunctions, please contact HIWIN for help.

2. PROFINET communication

2.1 Brief introduction

PROFINET is a real-time fieldbus protocol based on Ethernet. It classifies all the devices into controllers, supervisors, and field devices.

- **Controller**
  A controller contains process IO image table and user program. A PLC is a typical controller which controls the whole application.
- **Supervisor**
  A supervisor can be programming device (PG), personal computer (PC) or human machine interface (HMI) for commissioning or diagnosis purposes.
- **Field device**
  A field device is a communication slave controlled by the controller. Field device can transmit the processed data and system statuses (such as diagnoses and alarms) according to PROFINET protocol.

E1 PROFINET drives are PROFINET field devices. A device description file, called GSD (**G**eneral **S**tation **D**escription) file, describes the functionality of E1 PROFINET drives. A controller uses GSD file to identify and configure field devices.

2.2 Communication specification

Table 2.2.1

PROFINET	Physical layer	100BASE-TX (IEEE 802.3)
	Baud rate	100 Mbps
	Cable	Ethernet Category 5 or higher (twisted-pair cable with double, aluminum tape and braided shielding)
	Cable length	Max. 100 m (node to node)
	Connector	RJ45
	Communication service	Real-time communication (RT) Isochronous real-time communication (IRT)
	Send clock	RT: 500 μs, 1 ms, 2 ms, 4 ms IRT: Min. 500 μs (500 μs increment)
PROFIdrive	Supported Telegram	Standard Telegram 3 Standard Telegram 9 HIWIN Telegram 111
	Control mode	Speed mode, Position mode

2.3 Communication indicators

Figure 2.3.1 is the panel of an E1 drive. LED1 will display drive alarm code when a drive alarm occurs. LA-IN (D1) and LA-OUT (D2) indicate the connection status. Other LEDs are not functional yet.

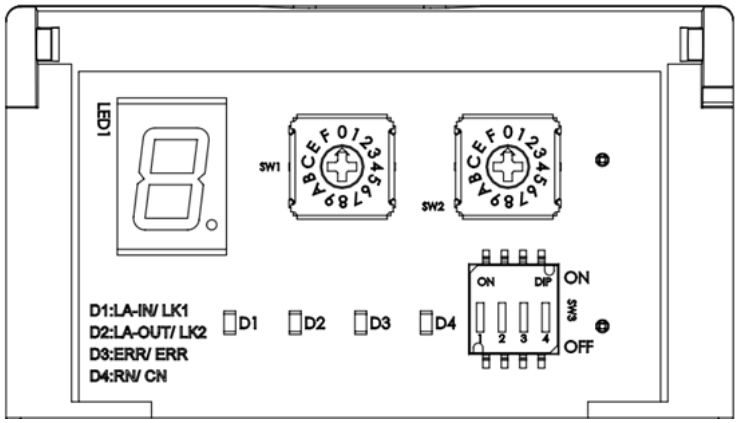


Figure 2.3.1

Table 2.3.1

LED	Color	Status	Description
LA-IN	Green	On	The bus connector is connected and correctly wired.
		Off	The bus connector is not connected.
LA-OUT	Green	On	The bus connector is connected and correctly wired.
		Off	The bus connector is not connected.

2.4 PROFINET device model

PROFINET device model illustrates the relationship among controllers, supervisors and field devices.

2.4.1 Slot, subslot and index

A device model is represented by DAP (**D**evice **A**ccess **P**oint) which defines modules to a particular device. It also addresses all IO signals in field devices, and this requires corresponding specifications to be made during data modeling. The addressing options are illustrated as below.

- **Slot (Module)**
  Slot is the physical slot of an IO module of a field device. A module may contain one or more subslots for data exchange.
- **Subslot**
  Subslot is used as a communication object for IO data exchange, parameter access and alarm mechanism.
- **Index**
  Index specifies the accessible data in a slot/subslot. For example, parameters can be written to a module or read from a module by indexes.

Figure 2.4.1.1 shows a modular device model with a bus interface and three input/output modules.

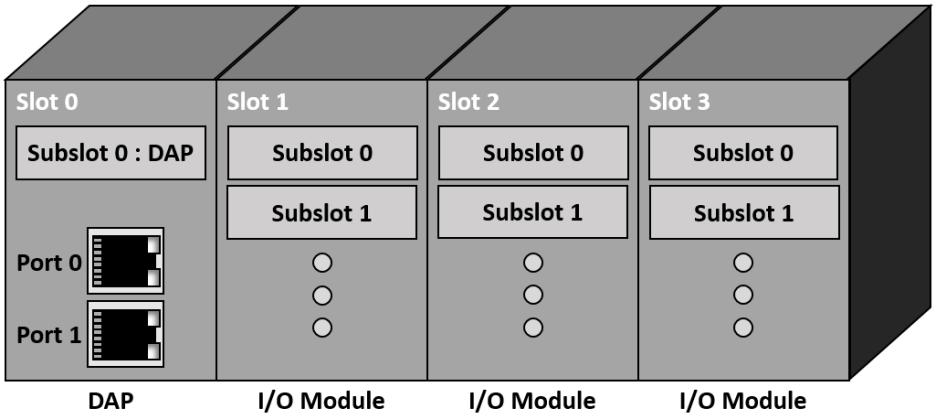


Figure 2.4.1.1

2.4.2 Application process identifier (API)

API (**A**pplication **P**rocess **I**dentifier) is the application profiles registered with PI. PROFIdrive DO is represented by a module with PROFIdrive application process (API=0x3A00).

2.4.3 Application relationship and communication relationship

Each data exchange between a controller/supervisor and a field device is executed with the precisely defined communication channels, which must be set up by the controller before data exchange. The following concept will provide the understanding of PROFINET device model and communication service.

- **Application relationship (AR)**

  Each data exchange is embedded in an application relationship (AR). AR is established between a controller and a field device. The following different ARs are defined in PROFINET communication.

Table 2.4.3.1

	Connect to API	Cyclic Data	Acyclic Data	Alarm	Write Access
IOC-AR	V	V	V	V	V
IOS-AR		V	V	V	V
IOS-DA			V		V
Implicit AR			V		

IOC-AR (**C**ontroller **AR**) defines the relationship between a controller and a field device. IOS-AR (**S**upervisor **AR**) defines the relationship between a supervisor and a field device. IOS-DA (**S**upervisor **D**ata **A**ccess) also defines the relationship between a supervisor and a field device, but this AR only supports acyclic data access. Finally, implicit AR is for read acyclic data between controller/supervisor and field device. This AR is always established and used by a controller. For E1 PROFINET drives, IOC-AR is used to exchange cyclic data, acyclic data and alarms, and this AR is set as PROFIdrive API (0x3A00).

- **Communication relationship (CR)**

  Communication relationship (CR) for data exchange must be established within an AR, which specifies the explicit communication channel between a consumer and a provider. The following different CRs are defined in PROFINET communication.

Table 2.4.3.2

	Cyclic Data	Acyclic Data	Multicast
IO-CR	V		
Alarm-CR		V	
Record Data-CR		V	
MCR	V		V

IO-CR is defined for cyclic processed data exchange. Alarm-CR is defined for acyclic alarm transmission. Record Data-CR is defined for acyclic data exchange. Finally, MCR (**M**ulticast **C**ommunication **R**elationship) defines the communication between field devices. As E1 PROFINET drive application, PROFINET defines IO-CR for cyclic data exchange and Record Data-CR for acyclic data exchange.

2.4.4 Relationship between device model and addressing

When configuring an automation system, engineers specify the data to be exchanged of the field device. Controller can set multiple IO-CRs for the field device. The actual applications in the field device are recognized based on the API. Figure 2.4.4.1 shows the relationship among IO-CRs, APIs, slots and subslots.

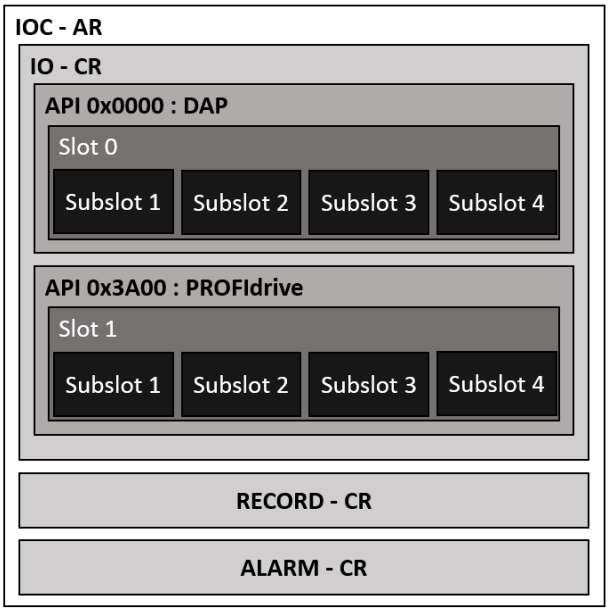


Figure 2.4.4.1

2.5 PROFINET communication services

PROFINET uses three different communication channels to exchange data. Standard Ethernet channel is used for parameterization and configuration of devices for acyclic operation. PROFINET RT (**R**eal-**T**ime) and PROFINET IRT (**I**sochronous **R**eal-**T**ime) are used for real-time application such as a motion control system.

2.5.1 PROFINET real-time class

In real-time communication, the response time must be within 5 to 10 ms. For this reason, it is necessary to add mechanisms to PROFINET that supports both UDP/IP communications and offers an optimized communication path. PROFINET uses Ethertype 0x8892 (which has a higher priority than UDP/IP frames) and Frame_ID to address a particular communication channel. The following shows three kinds of real-time classes of PROFINET communication channels.

- **RT_CLASS_1**
  Non-synchronized RT communication within a subnet. No special addressing information is required for this communication. Standard switches suitable for industrial environments can be used in this RT class.

- **RT_CLASS_3**
  Synchronized communication within a subnet. During synchronized communication, the processed data is sent with the maximum precision in an exact order specified during system engineering. This optimized data transmission is called IRT functionality, and the following section will introduce PROFINET IRT communication.

- **RT_CLASS_UDP**
  Non-synchronized cross-subnet communication between different subnets requires addressing information about destination network. This variant is also called RT_CLASS_UDP, and it describes the properties of the local send list control in a device and the switching.

2.5.2 PROFINET acyclic data

Acyclic data exchange is used for parameterizing, configuring devices and reading status information. This is accomplished with read/write frames via standard IT services by UDP/IP service. In addition to the data records available for use by device manufactures, the system data records are specially defined diagnostic information, error log entries, identification information, information function and IO data signals. The Ethernet frame of PROFINET UDP/IP service is shown as below.

2 Bytes	28 Bytes	80 Bytes	2 Bytes	1~1364 Bytes	4 Bytes
Ethertype	UDP/IP	RPC	NDR	PROFINET data block	FCS

To an IPv4 data, Ethertype is 0x0800 and PROFINET data block is different from the transmitted type of PROFINET protocol. For example, parameterizing and configuring devices use the read request and write request in normal.

2.5.3 PROFINET cyclic data

After one AR and IO-CR are successfully created, cyclic data starts transmitting without acknowledgement. As previously mentioned, the processed data is assigned to subsolts. Cyclic data defines IOPS (**IO P**rovider **S**tatus) and IOCS (**IO C**onsumer **S**tatus) for each subslot to specify data status more precisely. Between controller and field device, each input data or output data has its own IOPS and IOCS. Data transmitters transfer IOPS to data receivers, and data receivers transfer IOCS back to data transmitters. IOPS and IOCS will be “Good” in normal, but they must be set as “Bad” in the following conditions.

- **IOPS**
  - A submodule is not available for an established AR.
  - The application in controller detects the received submodule data is not valid, and IOPS of controller also informs field device about the validity of the output data from controller.
- **IOCS**
  - A submodule is not available for an established AR.
  - Device application cannot process the data.

The Ethernet frame of PROFINET cyclic data is shown as below. There are two types, one is the frame form controller to field device, and the other is the frame from field device to controller.

4 Bytes	2 Bytes	2 Bytes	1 Byte	…	1 Byte	4 Bytes	4 Bytes
VLAN	Ethertype	Frame_ID	*IOCS	*Data	*IOPS	ADPU status	FCS

4 Bytes	2 Bytes	2 Bytes	1 Byte	…	1 Byte	4 Bytes	4 Bytes
VLAN	Ethertype	Frame_ID	*IOPS	*Data	*IOCS	ADPU status	FCS

Cyclic data has VLAN tag, and the Ethertype of PROFINET protocol is 0x8892. Each output data has an IOPS, and each input data has an IOCS. ADPU status defines the application protocol data unit status.

2.6 PROFINET IRT communication

To satisfy the requirement of the maximum performance and deterministic behavior, PROFINET defines synchronized PROFINET communication called IRT communication (**I**sochronous **R**eal-**T**ime **C**ommunication). The bus cycle of IRT communication is significantly less than 1 ms, and the maximum deviation from the start of the bus cycle is less than 1 us. To provide the maximum performance, PROFINET communication requires precise planning of communication paths in advance. Figure 2.6.1 shows a user scenario in which both the bus cycle and the specific application in the field devices are synchronized.

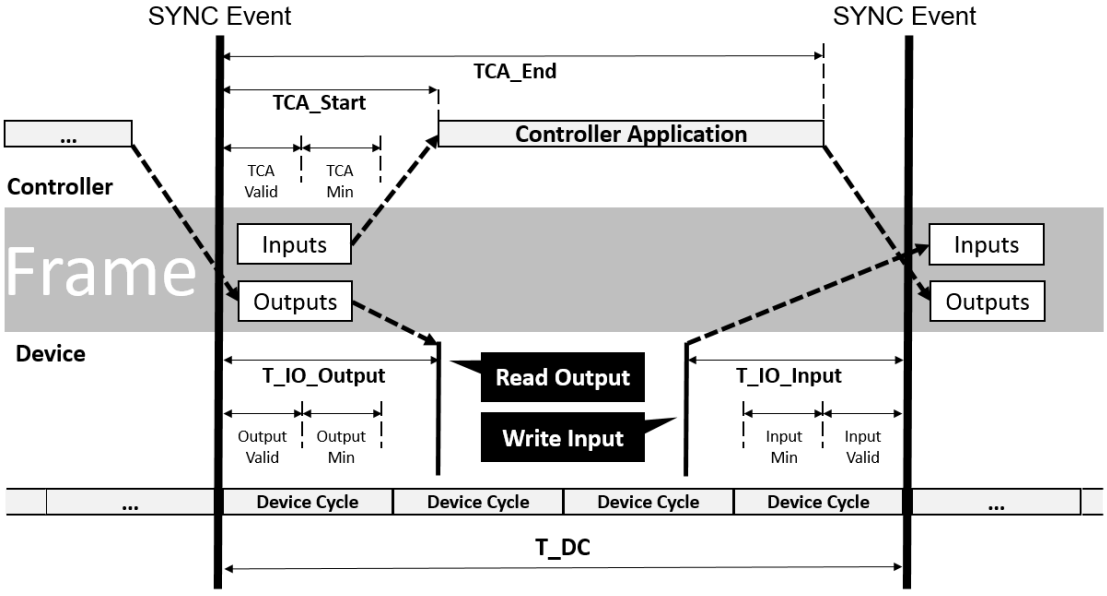


Figure 2.6.1

2.7 PROFINET system startup

Engineers need to set up PROFINET system step by step. The procedure can be separated into five steps, system engineering, download system information, address resolution, system startup and start to data exchange. These steps are explained in detail in the following sections.

2.7.1 System engineering

During system engineering, the GSD files of field devices are necessary. There is a need to map the modules/submodules defined in GSD files onto the real system and assign them to slots/subslots.

2.7.2 Download system information

After the completion of system engineering, engineers need to download the system data to the controller which also contains the system-specific application. In the end of this step, the controller has all the information needed for addressing the field devices and for data exchange.

2.7.3 Address resolution

Before it can perform data exchange with a field device, a controller must assign the field device an IP address before system startup. System startup refers to the start/restart of an automation system after power on or reset the system. The IP address is assigned within the subnet using PROFINET DCP protocol. If the field device is in a different subnet from that of controller, address resolution by a separate DHCP server will be offered.

2.7.4 System startup

A controller always initiates the system startup following start/restart based on the configuration data. This happens automatically from the perspective of the user. During system startup, a controller establishes ARs, CRs, configurations and IO data, then the PROFINET system is ready for data exchange.

2.7.5 Start to data exchange

After the successful completion of system startup, the controller and field devices exchange the processed data, alarms and acyclic data. Figure 2.7.5.1 shows the sequence of startup after power on or reset the system.

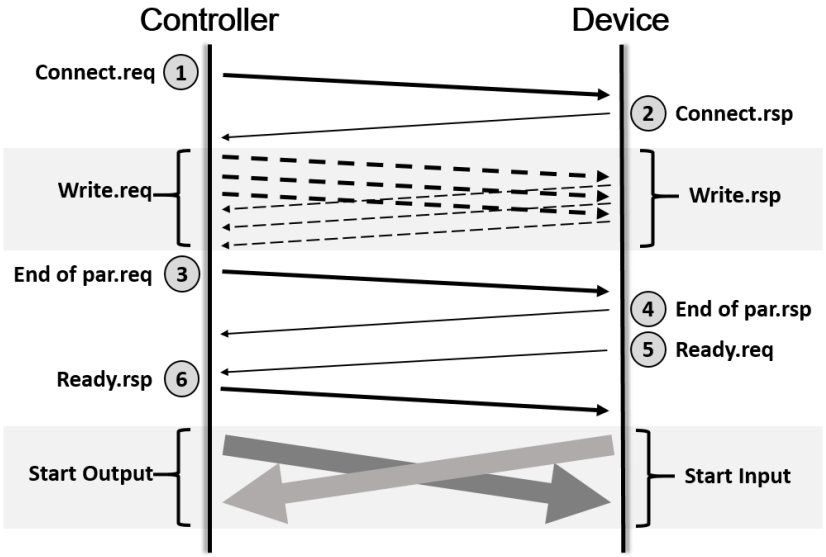


Figure 2.7.5.1

(This page is intentionally left blank.)

3. Supported telegrams and IO data

3.1 IO data signals

Table 3.1.1 describes the IO data signals of E1 drive telegrams.

Table 3.1.1

Signal	Description	Data Type	Access	Valid Range	Unit
STW1	Control word 1	Uint16	Receive	0 ~ 65535	-
STW2	Control word 2	Uint16	Receive	0 ~ 65535	-
ZSW1	Status word 1	Uint16	Send	0 ~ 65535	-
ZSW2	Status word 2	Uint16	Send	0 ~ 65535	-
NSOLL_B	Speed setpoint B (32 bit)*1	Int32	Receive	-2^{31} ~ (2^{31}-1)	$100/2^{30}$ %
NIST_B	Speed actual value B (32 bit) *1	Int32	Send	-2^{31} ~ (2^{31}-1)	$100/2^{30}$ %
G1_STW	Encoder 1 control word	Uint16	Receive	0 ~ 65535	-
G1_ZSW	Encoder 1 status word	Uint16	Send	0 ~ 65535	-
G1_XIST1	Encoder 1 actual position 1	Uint32	Send	-2^{31} ~ (2^{31}-1)	Control unit
G1_XIST2	Encoder 1 actual position 2	Uint32	Send	-2^{31} ~ (2^{31}-1)	Control unit
SATZANW	Position block selection	Uint16	Receive	0 ~ 65535	-
AKTSATZ	Selected position block	Uint16	Send	0 ~ 65535	-
XIST_A	Position actual value A	Int32	Send	-2^{31} ~ (2^{31}-1)	Control unit
MDI_TARPOS	MDI position	Int32	Receive	-2^{31} ~ (2^{31}-1)	Control unit
MDI_VELOCITY	MDI velocity	Int32	Receive	0 ~ (2^{31}-1)	1000control unit/min
MDI_ACC	MDI acceleration override*2	Int16	Receive	0 ~ 16384	$100/2^{14}$ %
MDI_DEC	MDI deceleration override*2	Int16	Receive	0 ~ 16384	$100/2^{14}$ %
MDI_MODE	Position MDI mode	Uint16	Receive	0 ~ 65535	-
POS_STW1	Positioning control word 1	Uint16	Receive	0 ~ 65535	-
POS_STW2	Positioning control word 2	Uint16	Receive	0 ~ 65535	-
POS_ZSW1	Positioning status word 1	Uint16	Send	0 ~ 65535	-
POS_ZSW2	Positioning status word 2	Uint16	Send	0 ~ 65535	-
MELDW	Status word interconnection	Uint16	Send	0 ~ 65535	-
OVERRIDE	Velocity override*3	Uint16	Receive	0 ~ 32767	$100/2^{14}$ %
FAULT_CODE	Fault code	Uint16	Send	0 ~ 65535	N.A
WARN_CODE	Warning code	Uint16	Send	0 ~ 65535	N.A

Note:

*1 Act on velocity reference value (Pt317 or Pt386), the setting ratio will be active based on actual command speed.

For relative information, please refer to section 6.1 **Velocity reference value setting**.

*2 Act on acceleration / deceleration reference value (Pt534 / Pt537), the setting ratio will be active based on actual acceleration / deceleration. For relative information, please refer to section 6.6.1 **MDI setpoints**.

*3 Act on the command value of MDI velocity, the setting ratio will be active based on actual command speed. For relative information, please refer to section 6.6.1 **MDI setpoints**.

3.2 Supported telegrams

Table 3.2.1 shows the supported telegrams of E1 PROFINET drives. “Received word” represents the processed data (PZD) sent from controller to field devices; while “Sent word” represents the processed data sent form field devices to controller.

Table 3.2.1

Telegram	Maximum number of PZD	
	Received word	Sent word
Standard telegram 3	5	9
Standard telegram 9	10	5
HIWIN telegram 111	12	12

Table 3.2.2 is the frame of the supported telegrams.

Table 3.2.2

	Telegram 3		Telegram 9		Telegram 111	
Item	Received word	Sent word	Received word	Sent word	Received word	Sent word
PZD1	STW1	ZSW1	STW1	ZSW1	STW1	ZSW1
PZD2	NSOLL_B	NIST_B	SATZANW	AKTSATZ	POS_STW1	POS_ZSW1
PZD3			STW2	ZSW2	POS_STW2	POS_ZSW2
PZD4	STW2	ZSW2	MDI_TARPOS	XIST_A	STW2	ZSW2
PZD5	G1_STW	G1_ZSW			OVERRIDE	MELDW
PZD6		G1_XIST1	MDI_VELOCITY		MDI_TARPOS	XIST_A
PZD7						
PZD8		G1_XIST2	MDI_ACC		MDI_VELOCITY	NIST_B
PZD9			MDI_DEC			
PZD10			MDI_MOD		MDI_ACC	FAULT_CODE
PZD11					MDI_DEC	WARN_CODE
PZD12						

Note:

(1) 1 PZD = 1 word.

(2) For SATZANW, only MDI submode is functional.

3.3 Control word 1 (STW1)

3.3.1 Telegram 3

Table 3.3.1.1

bit	Description	
0	1: ON	0: OFF
1	1: No Coast stop (No OFF2)	0: Coast stop (OFF2)
2	1: No Quick stop (No OFF3)	0: Quick stop (OFF3)
3	1: Enable operation	0: Disable operation
4	Reserved	
5	Reserved	
6	Reserved	
7	Fault acknowledge (rising edge active)	
8	Reserved	
9	Reserved	
10	1: Control by PLC	0: Control by Device
11	Reserved	
12~15	Reserved	

3.3.2 Telegram 9, Telegram 111

Table 3.3.2.1

bit	Description	
0	1: ON	0: OFF
1	1: No Coast stop (No OFF2)	0: Coast stop (OFF2)
2	1: No Quick stop (No OFF3)	0: Quick stop (OFF3)
3	1: Enable operation	0: Disable operation
4	1: Do Not Reject Traversing Task	0: Reject Traversing Task
5	1: No Intermediate Stop	0: Intermediate Stop
6	Activate Traversing Task (rising edge active)	
7	Fault acknowledge (rising edge active)	
8	1: JOG 1 ON (jog positive)*1	0: JOG 1 OFF
9	1: JOG 2 ON (jog negative)*1	0: JOG 2 OFF
10	1: Control by PLC	0: Control by Device
11	1: Start Homing Procedure	0: Stop Homing Procedure
12~15	Reserved	

Note:

*1 If Pt53A = 1 (,which will reverse the jog direction of JOG 1 and JOG 2), the definition of JOG moving direction will be changed. For example, when Pt53A = 1, JOG 1 is negative; JOG 2 is positive; the default of Pt53A is 0.

Pt No.	Pt53A	PNU Number	0x253A		
Data Type	Unit16	Setting Range	0~1	Default	0
Name	PROFIdrive JOG mode moving direction inverse setting	Unit	-	Applicable Motor	All
Effective	Immediately	Attribute	Setup	Applicable Telegram	9, 111

3.4 Control word 2 (STW2)

Table 3.4.1

bit	Description
0~11	Reserved
12~15	Controller Sign-of-Life

3.5 Status word 1 (ZSW1)

3.5.1 Telegram 3

Table 3.5.1.1

bit	Description	
0	1: Ready to switch on	0: Not ready to switch on
1	1: Ready to operation	0: Not ready to operation
2	1: Operation enable	0: Operation disable
3	1: Fault	0: No fault
4	1: No Coast stop act (No OFF2)	0: Coast stop act (OFF2)
5	1: No Quick stop act (No OFF3)	0: Quick stop act (OFF3)
6	1: Switch on inhibited	0: Switch on not inhibited
7	1: Warning present	0: No warning
8	1: Speed error within tolerance range	0: Speed error out of tolerance range
9	1: Control requested	0: No Control requested
10	1: Speed is reached	0: Speed is not reached
11	Reserved	
12~15	Reserved	

3.5.2 Telegram 9, Telegram 111

Table 3.5.2.1

bit	Description	
0	1: Ready to switch on	0: Not ready to switch on
1	1: Ready to operation	0: Not ready to operation
2	1: Operation enable	0: Operation disable
3	1: Fault	0: No fault
4	1: No Coast stop act (No OFF2)	0: Coast stop act (OFF2)
5	1: No Quick stop act (No OFF3)	0: Quick stop act (OFF3)
6	1: Switch on inhibited	0: Switch on not inhibited
7	1: Warning present	0: No warning
8	1: Following error in tolerance range	0: Following error out of tolerance range
9	1: Control requested	0: No Control requested
10	1: Target position reached	0: Not at target position
11	1: Home position set	0: Home position not yet set
12	Traversing task acknowledgement (rising edge active)	
13	1: Motor stopped	0: Motor moving
14~15	Reserved	

3.6 Status word 2 (ZSW2)

3.6.1 Telegram 3

Table 3.6.1.1

bit	Description
0~11	Reserved
12~15	Drive Sign-of-Life

3.6.2 Telegram 9, Telegram 111

Table 3.6.2.1

bit	Description	
0~10	Reserved	
11	1: Pulses enabled	0: Pulses disabled
12~15	Drive Sign-of-Life	

3.7 Encoder 1 control word (G1_STW)

Table 3.7.1

bit	Description	
0	Function 1 (Reference mark 1)	
1	Function 2 (Reference mark 2)	
2	Function 3 (Reference mark 3)	
3	Function 4 (Reference mark 4)	
4	0: No function 1: Activate functions 2: Read value 3: Cancel functions Other: Reserved	
5		
6		
7	1: Reserved	0: Reference mark search
8	Reserved	
9	Reserved	
10	Reserved	
11	Reserved	
12	Reserved	
13	Reserved	
14	1: Activate parking sensor	
15	1: Acknowledging a sensor error	

3.8 Encoder 1 status word (G1_ZSW)

Table 3.8.1

bit	Description
0	Function 1 (Reference mark 1)
1	Function 2 (Reference mark 2)
2	Function 3 (Reference mark 3)
3	Function 4 (Reference mark 4)
4	Value 1 (Reference mark 1)
5	Value 2 (Reference mark 2)
6	Value 3 (Reference mark 3)
7	Value 4 (Reference mark 4)
8	Reserved
9	Reserved
10	Reserved
11	Error acknowledgement in process
12	Reserved
13	Reserved
14	1: Parking sensor activated
15	1: Sensor error

3.9 Position block selection (SATZANW)

Table 3.9.1

bit	Description	
0~9	Reserved	
10~14	Reserved	
15	1: Activate MDI submode	0: Deactivate MDI submode

3.10 Selected position block (AKTSATZ)

Table 3.10.1

bit	Description	
0~9	Reserved	
10~14	Reserved	
15	1: Activate MDI submode	0: Deactivate MDI submode

3.11 Position MDI mode (MDI_MODE)

Table 3.11.1

bit	Description	
0	1: Absolute positioning	0: Relative positioning
1	Reserved	
2	Reserved	
3~15	Reserved	

3.12 Positioning control word 1 (POS_STW1)

Table 3.12.1

bit	Description	
0~7	Reserved	
8	1: Absolute positioning selected	0: Relative positioning selected
9~14	Reserved	
15	1: Activate MDI submode	0: Deactivate MDI submode

3.13 Positioning control word 2 (POS_STW2)

Table 3.13.1

bit	Description
0	Reserved
1	1: Set current position as reference point
2~15	Reserved

3.14 Positioning status word 1 (POS_ZSW1)

Table 3.14.1

bit	Description
0~7	Reserved
8	1: Negative overtravel active
9	1: Positive overtravel active
10	1: Jog active
11	1: Homing procedure active
12~14	Reserved
15	1: MDI active

3.15 Positioning status word 2 (POS_ZSW2)

Table 3.15.1

bit	Description
0~1	Reserved
2	1: Setpoint available
3	Reserved
4	1: Axis moves forwards
5	1: Axis moves backwards
6~14	Reserved
15	1: Traversing command active

3.16 Status word interconnection (MELDW)

Table 3.16.1

bit	Description
0	Reserved
1	1: Torque limit is not active
2~5	Reserved
6	1: Not motor overload warning
7	1: No I^2T warning
8	1: Speed deviation is in tolerance (not functional, always is 1)
9~10	Reserved
11	1: Controller enabled
12	1: Drive ready
13	1: Pulses enabled
14~15	Reserved

4. Parameters

4.1 PROFIdrive parameters

Table 4.1.1 describes the supported PROFIdrive parameters.

Table 4.1.1

<table>
<tr><th>PNU (dec)</th><th>Read / Write</th><th>Data Type</th><th>Significance</th><th>Default</th></tr>
<tr><td rowspan="2">922</td><td>Read</td><td>Uint16</td><td>Telegram selection</td><td>0</td></tr>
<tr><td colspan="4">This parameter displays the current activated telegram number.</td></tr>
<tr><td rowspan="2">925</td><td>Read / Write</td><td>Uint16</td><td>Tolerance of the number of Controller Sign-of-Life failures</td><td>5</td></tr>
<tr><td colspan="4">The setting tolerance of Controller Sign-of-Life failures.</td></tr>
<tr><td rowspan="2">930</td><td>Read</td><td>Uint16</td><td>Operation mode</td><td>-</td></tr>
<tr><td colspan="4"><table><tr><th>Value</th><th>Description</th></tr><tr><td>1</td><td>Speed mode with RFG functionality</td></tr><tr><td>2</td><td>Position mode</td></tr><tr><td>3</td><td>Speed mode without RGF functionality</td></tr></table></td></tr>
<tr><td rowspan="2">944</td><td>Read</td><td>Uint16</td><td>Fault message counter</td><td>-</td></tr>
<tr><td colspan="4">The fault message counter is incremented each time the fault buffer changes.</td></tr>
<tr><td rowspan="2">945</td><td>Read</td><td>Uint16 Array[64]</td><td>Fault code</td><td>-</td></tr>
<tr><td colspan="4">The fault codes of alarms.
Refer to table 5.1.1 for the fault codes of drive alarms.</td></tr>
<tr><td rowspan="2">947</td><td>Read</td><td>Uint16 Array[64]</td><td>Fault number</td><td>-</td></tr>
<tr><td colspan="4">The internal fault numbers of alarms.
Refer to table 5.1.1 for the fault numbers of drive alarms.</td></tr>
<tr><td rowspan="2">950</td><td>Read</td><td>Uint16 Array[2]</td><td>Scaling of the fault buffer</td><td>-</td></tr>
<tr><td colspan="4">This parameter defines the number of fault situations (Subindex 0) and the number of fault messages in a fault situation (Subindex 1) of the fault buffer.</td></tr>
<tr><td rowspan="2">952</td><td>Read / Write</td><td>Uint16</td><td>Fault situation counter</td><td>-</td></tr>
<tr><td colspan="4">This parameter specifies the number of fault situations.
If this parameter is set to 0, the complete fault buffer is deleted.</td></tr>
</table>

<table>
<tr><th>PNU (dec)</th><th>Read / Write</th><th>Data Type</th><th>Significance</th><th>Default</th></tr>
<tr><td rowspan="2">964</td><td>Read</td><td>Uint16 Array[7]</td><td>Drive unit identification</td><td>-</td></tr>
<tr><td colspan="4">Data for the drive identification.
<table>
<tr><th>Subindex</th><th>Significance</th><th>Description</th></tr>
<tr><td>0</td><td>Manufacturer</td><td>Fix 0xAAA</td></tr>
<tr><td>1</td><td>Drive unit type</td><td>Fix 0x05</td></tr>
<tr><td>2</td><td>Firmware version</td><td>xxyy (decimal)
Example: Version 2.1 results in 0201.</td></tr>
<tr><td>3</td><td>Firmware data (year)</td><td>yyyy (decimal)</td></tr>
<tr><td>4</td><td>Firmware data (day/month)</td><td>ddmm (decimal)</td></tr>
<tr><td>5</td><td>Number of DO</td><td>Fix 1</td></tr>
<tr><td>6</td><td>Minor Firmware Version</td><td>-</td></tr>
</table></td></tr>
<tr><td rowspan="2">979</td><td>Read</td><td>Uint32 Array[31]</td><td>Encoder format</td><td>-</td></tr>
<tr><td colspan="4">
<table>
<tr><th>Subindex</th><th>Significance</th><th>Description</th></tr>
<tr><td>0</td><td>Header</td><td>-</td></tr>
<tr><td>1</td><td>Encoder type</td><td>-</td></tr>
<tr><td>2</td><td>Encoder resolution</td><td>-</td></tr>
<tr><td>3</td><td>Shift factor for G1_XIST1</td><td>-</td></tr>
<tr><td>4</td><td>Shift factor for G1_XIST2</td><td>-</td></tr>
<tr><td>5</td><td>Determinable revolutions</td><td>-</td></tr>
<tr><td>6~30</td><td>Reserved</td><td>-</td></tr>
</table></td></tr>
</table>

4.2 PNU dictionary table

Table 4.2.1

PNU (Hex)	Sub-Index	Name	Data type	Read write attribute	Applicable Telegram	Valid value	Unit
2XXXh	00h	The 2000h series objects are from servo parameters. Please refer to “E1 Series Servo Drive User Manual” for more information. The mapping relationship between servo parameter numbers and object indexes is as follows: Object index = 2000h + servo parameter number					
3000h	00h	Motor type	U16	ro	All	0 ~ 2	-
3001h	00h	Inner encoder resolution	I32	ro	All	-2147483648 ~ 2147483647	-
3056h	00h	Software state[12]	U16	ro	All	0 ~ 0xFFFF	-
3057h	00h	Application mode of gantry system	U16	rw	All	1, 2, 11	-
3058h	00h	Yaw target position	I32	rw	All	-2147483648 ~ 2147483647	control unit
3059h	00h	Yaw feedback position	I32	ro	All	-2147483648 ~ 2147483647	control unit
3060h	00h	Use touch probe enable error map	U16	rw	All	0 ~ 1	-
3061h	00h	Enable position trigger function	U16	rw	All	0 ~ 1	-
3062h	00h	Overtravel stop mode selection	U16	rw	All	0 ~ 1	-
3110h	00h	Drive warning events 1	U16	ro	All	0 ~ 0xFFFF	-
3111h	00h	Drive warning events 2	U16	ro	All	0 ~ 0xFFFF	-
3200h	00h	Absolute encoder initialization	I32	rw	All	0 ~ 1	-
3201h	00h	General object i1	I32	rw	All	-2147483648 ~ 2147483647	-
3202h	00h	General object i2	I32	rw	All	-2147483648 ~ 2147483647	-
3203h	00h	General object i3	I32	rw	All	-2147483648 ~ 2147483647	-
3204h	00h	General object i4	I32	rw	All	-2147483648 ~ 2147483647	-
3205h	00h	General object i5	I32	rw	All	-2147483648 ~ 2147483647	-
3206h	00h	General object i6	I32	rw	All	-2147483648 ~ 2147483647	-
3207h	00h	General object i7	I32	rw	All	-2147483648 ~ 2147483647	-
3208h	00h	General object i8	I32	rw	All	-2147483648 ~ 2147483647	-
3209h	00h	General object i9	I32	rw	All	-2147483648 ~ 2147483647	-
3210h	00h	General object f0	F32	rw	All	-3.40282e+38 ~ 3.40282e+38	-
3211h	00h	General object f1	F32	rw	All	-3.40282e+38 ~ 3.40282e+38	-
3212h	00h	General object f2	F32	rw	All	-3.40282e+38 ~ 3.40282e+38	-
3213h	00h	General object f3	F32	rw	All	-3.40282e+38 ~ 3.40282e+38	-
3214h	00h	General object f4	F32	rw	All	-3.40282e+38 ~ 3.40282e+38	-

PNU (Hex)	Sub-Index	Name	Data type	Read write attribute	Applicable Telegram	Valid value	Unit
3215h	00h	Reset drive	I16	rw	All	0 ~ 1	-
3216h	00h	Send parameter to flash	-	rw	All	0 ~ 1	-
4XXXh	00h	The 4000h series objects are from servo Ut parameters. Please refer to section 14.3.4 **List of monitoring items** in “E1 Series Servo Drive User Manual” for more information. The mapping relationship between servo Ut parameter numbers and object indexes is as follows: Object index = 4000h + servo Ut parameter number Example: For servo drive’s parameter Ut095, its corresponding object is 4095h.					

4.3 Parameter access

E1 PROFINET drives only support the single parameter request. Figure 4.3.1 describes the data flow of a parameter access.

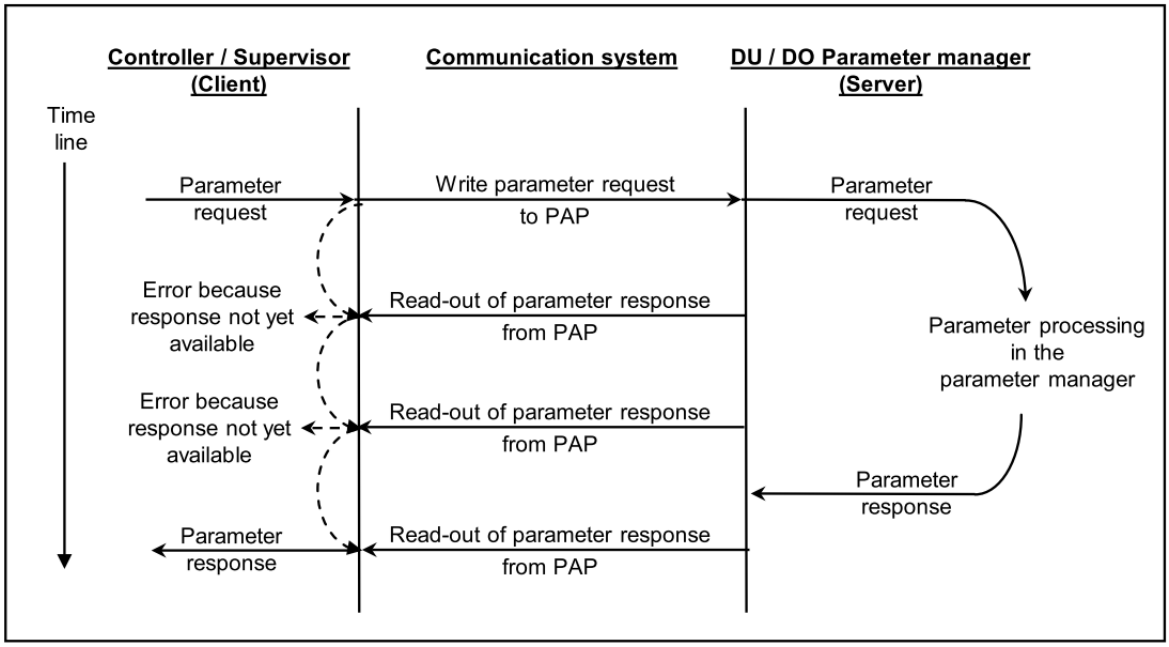


Figure 4.3.1

4.3.1 Struct of read a value

Table 4.3.1.1 Parameter request

Byte	Block Definition	Field	Value
0	Request header	Request reference	0x01~0xFF (by master)
1		Request ID	0x01
2		DO-ID	0
3		Number of parameters	1
4	Parameter address	Attribute	0x10 (Value)
5		Number of elements	0
6		Parameter number	PNU number
7			
8		Subindex (irrelevant)	0
9			

Table 4.3.1.2 Parameter response positive

Byte	Block Definition	Field	Value
0	Response header	Request reference mirrored	-
1		Response ID	0x01
2		DO-ID mirrored	-
3		Number of parameters	1
4	Parameter value	Format	Refer to table 4.3.5.1
5		Number of values	1
6		Value	data
7			
8			
9			

Table 4.3.1.3 Parameter response negative

Byte	Block Definition	Field	Value
0	Response header	Request reference mirrored	-
1		Response ID	0x81
2		DO-ID mirrored	-
3		Number of parameters	1
4	Parameter value	Format	0x44
5		Number of values	1
6		Error value	Refer to table 4.3.6.1
7			

4.3.2 Struct of write a value

Table 4.3.2.1 Parameter request

Byte	Block Definition	Field	Value
0	Request header	Request reference	0x01~0xFF (by master)
1		Request ID	0x02
2		DO-ID	0
3		Number of parameters	1
4	Parameter address	Attribute	0x10 (Value)
5		Number of elements	0
6		Parameter number	PNU number
7			
8		Subindex (irrelevant)	0
9			
10	Parameter value	Format	Refer to table 4.3.5.1
11		Number of values	1
12		Value	data
13			
14			
15			

Table 4.3.2.2 Parameter response positive

Byte	Block Definition	Field	Value
0	Response header	Request reference mirrored	-
1		Response ID	0x02
2		DO-ID mirrored	-
3		Number of parameters	1

Table 4.3.2.3 Parameter response negative

Byte	Block Definition	Field	Value
0	Response header	Request reference mirrored	-
1		Response ID	0x82
2		DO-ID mirrored	-
3		Number of parameters	1
4	Parameter value	Format	0x44
5		Number of values	1
6		Error value	Refer to table 4.3.6.1
7			

4.3.3 Struct of read array elements

Table 4.3.3.1 Parameter request

Byte	Block Definition	Field	Value
0	Request header	Request reference	0x01~0xFF (by master)
1		Request ID	0x01
2		DO-ID	0
3		Number of parameters	1
4	Parameter address	Attribute	0x10 (Value)
5		Number of elements	n
6		Parameter number	PNU number
7			
8		Subindex	0
9			

Table 4.3.3.2 Parameter response positive

Byte	Block Definition	Field	Value
0	Response header	Request reference mirrored	-
1		Response ID	0x01
2		DO-ID mirrored	-
3		Number of parameters	1
4	Parameter value	Format	Refer to table 4.3.5.1
5		Number of values	n
6		Value	data 1 to n
7			
8			
9			
…			

Table 4.3.3.3 Parameter response negative

Byte	Block Definition	Field	Value
0	Response header	Request reference mirrored	-
1		Response ID	0x81
2		DO-ID mirrored	-
3		Number of parameters	1
4	Parameter value	Format	0x44
5		Number of values	1
6		Error value	Refer to table 4.3.6.1
7			

4.3.4 Struct of write array elements

Table 4.3.4.1 Parameter request

Byte	Block Definition	Field	Value
0	Request header	Request reference	0x01~0xFF (by master)
1		Request ID	0x02
2		DO-ID	0
3		Number of parameters	1
4	Parameter address	Attribute	0x10 (Value)
5		Number of elements	n
6		Parameter number	PNU number
7			
8		Subindex	Subindex value
9			
10	Parameter value	Format	Refer to table 4.3.5.1
11		Number of values	n
12		Value	data 1 to n
13			
14			
15			
…			

Table 4.3.4.2 Parameter response positive

Byte	Block Definition	Field	Value
0	Response header	Request reference mirrored	-
1		Response ID	0x02
2		DO-ID mirrored	-
3		Number of parameters	1

Table 4.3.4.3 Parameter response negative

Byte	Block Definition	Field	Value
0	Response header	Request reference mirrored	-
1		Response ID	0x82
2		DO-ID mirrored	-
3		Number of parameters	1
4	Parameter value	Format	0x44
5		Number of values	1
6		Error value	Refer to table 4.3.6.1
7			

4.3.5 Struct information

Table 4.3.5.1

Field	Data Type	Value	Note
Format	Uint8	0x00: Reserved 0x01: Boolean 0x02: int8 0x03: int16 0x04: int32 0x05: Uint8 0x06: Uint16 0x07: Uint32 0x08: Float point 32 0x41: Byte 0x42: Word 0x43: Double word 0x44: Error Other: Reserved	-

4.3.6 Error number in parameter response

Table 4.3.6.1

Error No.	Error Name	Description
0x00	Impermissible parameter number	Access to an unavailable parameter.
0x01	Parameter value cannot be changed	Change access to a parameter value that cannot be changed.
0x02	Low or high limit exceeded	Change access with the value out of the value limits.
0x03	Faulty subindex	Access to an unavailable subindex of array or string parameter.
0x04	No array	Access with subindex to non-indexed parameter.
0x05	Incorrect data type	Change access with the value that does not match the data type of the parameter.
0x06	Setting not permitted (may only be reset)	Change access with the value unequal to 0. This is not permitted.
0x07	Description element cannot be changed	Change access to a description element that cannot be changed.
0x08	Reserved	-
0x09	Unavailable description data	Access to an unavailable description data.
0x0A	Reserved	-
0x0B	No operation priority	Change access without rights to change parameters.
0x0C	Reserved	-
0x0D		
0x0E		
0x0F	Unavailable text array	Access to an unavailable text array.
0x10	Reserved	-
0x11	Request cannot be executed because of operating state	Access is temporarily not possible for reasons that are not specified in detail.
0x12	Reserved	-
0x13		
0x14	Impermissible value	Change access with the value within the value limits, but not permissible for other long-term reasons (parameter with defined single values).
0x15	Response too long	The length of the current response exceeds the maximum transmittable length of the response transport block. In case of a multi parameter request, the response block was shortened by omitting the parameter requests.
0x16	Impermissible parameter address	Illegal value (reserved) or value which is not supported for

Error No.	Error Name	Description
		the attribute, illegal or not supported number of elements, illegal parameter number or illegal subindex or a combination.
0x17	Illegal format	Write request: Illegal format or format of the parameter data not supported.
0x18	Number of values are not consistent	Write request: Number of values of the parameter data does not match number of elements in parameter address.
0x19	Nonexistent Axis/DO	Access to an Axis/DO which does not exist.
0x1A~0x1F	Reserved	-
0x20	Parameter text element cannot be changed	Change access to a parameter text element that cannot be changed.
0x21	Service not supported	Illegal or unknown Request ID. (Response ID = 0x80)
0x22	Too much parameter requests	Multi parameter request: The response block does not contain all parameter responses since the maximum number of the supported parameter requests per multi parameter request exceeded.
0x23	Multi parameter access not supported	Device parameter manager does not support multi parameter requests. Request is discarded.
0x24~0xFF	Reserved	-

5. Diagnostics

5.1 Fault number / code of drive alarm

Table 5.1.1

Fault Number (dec)	Fault Code (hex)	Alarm Name	Alarm No.
4	024	System alarm 1	AL.024
5	025	System alarm 2	AL.025
6	030	Main circuit malfunction	AL.030
7	040	Parameter setting error	AL.040
11	050	Combination error	AL.050
12	070	Motor change detected	AL.070
14	0b0	Invalid servo on command	AL.0b0
15	100	Overcurrent detected	AL.100
16	320	Regenerative energy overflow	AL.320
17	400	Overvoltage	AL.400
18	410	Undervoltage	AL.410
19	510	Overspeed	AL.510
20	511	Encoder pulse output overspeed	AL.511
24	710	Overload (instantaneous maximum load)	AL.710
25	720	Overload (continuous maximum load)	AL.720
30	7A2	Power board temperature error	AL.7A2
33	800	Encoder absolute position lost	AL.800
34	810	Encoder battery undervoltage	AL.810
35	820	Encoder communication error	AL.820
36	830	Encoder data error	AL.830
37	840	Encoder communication crc error	AL.840
38	850	Encoder counting error	AL.850
39	860	Encoder data writing error	AL.860
40	870	Encoder overheating	AL.870
41	880	Incremental encoder signal phase order error	AL.880
42	890	Excellent Smart Cube (ESC) - incremental encoder disconnection	AL.890
43	8A0	First set of encoder - Excellent Smart Cube (ESC) signal error	AL.8A0
44	8b0	First set of encoder - encoder signal error	AL.8b0
45	8C0	Second set of encoder - Excellent Smart Cube (ESC) signal error	AL.8C0

Fault Number (dec)	Fault Code (hex)	Alarm Name	Alarm No.
46	8d0	Second set of encoder - encoder signal error	AL.8d0
47	8E0	Digital encoder disconnection	AL.8E0
48	8F0	Excellent Smart Cube (ESC) internal error	AL.8F0
49	861	Motor overheating	AL.861
50	b10	Velocity command A/D converter error	AL.b10
52	b20	Torque command A/D converter error	AL.b20
53	b33	Current detection malfunction	AL.b33
54	C10	Motor out of control	AL.C10
55	C20	Phase detection error	AL.C20
56	C21	Hall sensor error	AL.C21
58	C50	Electrical angle detection failure	AL.C50
59	C51	Overtravel detected during electrical angle detection	AL.C51
60	C52	Electrical angle detection incomplete	AL.C52
62	d00	Position deviation overflow	AL.d00
65	d10	Motor-load position deviation overflow	AL.d10
66	Eb0	Safety function alarm	AL.Eb0
67	Eb1	Safety function signal input timing error	AL.Eb1
68	Eb2	Safety function module error	AL.Eb2
69	F10	Power cable open phase	AL.F10
70	F50	Motor main circuit cable disconnection	AL.F50
71	FA0	Encoder power error	AL.FA0
72	FB0	Fieldbus communication hardware malfunction	AL.FB0
73	FB1	Fieldbus communication error	AL.FB1
74	FC0	Group control system communication error	AL.FC0
75	FC1	Slave axis error in group control system	AL.FC1
76	891	Incremental encoder signal error	AL.891
77	FB2	Fieldbus communication setup error	AL.FB2

(This page is intentionally left blank.)

6. Function descriptions

6.1 Velocity reference value setting

The velocity reference value is used to calculate normalized speed setpoints (NSOLL_A, NSOLL_B) and speed actual values (NIST_A, NIST_B). For NSOLL_A and NIST_A, 0x4000 is 100% of the velocity reference value; for NSOLL_B and NIST_B, 0x40000000 is 100% of the velocity reference value. The relationship among velocity reference value, normalized speed setpoints and speed actual values is shown as below.

$$Actual\ Command\ speed = \frac{NSOLL\_A}{4000h} \times (velocity\ reference\ value)$$

$$Actual\ Command\ speed = \frac{NSOLL\_B}{40000000h} \times (velocity\ reference\ value)$$

$$NIST\_A = \frac{(actual\ speed)}{(velocity\ reference\ value)} \times 4000h$$

$$NIST\_B = \frac{(actual\ speed)}{(velocity\ reference\ value)} \times 40000000h$$

To change the velocity reference value, set Pt317 for rotary motors, and set Pt386 for linear motors.

Pt No.	Pt317	PNU Number	0x2317		
Data type	Unit16	Setting Range	1~65535	Default	3000
Name	Motor reference velocity (rotary servo motor)*1	Unit	1 rpm	Applicable Motor	Rotary
Effective	Immediately	Attribute	Value	Applicable Telegram	3, 9, 111

Pt No.	Pt386	PNU Number	0x2386		
Data type	Unit16	Setting Range	1~100	Default	20
Name	Motor reference velocity (linear servo motor)*1	Unit	100 mm/s	Applicable Motor	Linear
Effective	Immediately	Attribute	Value	Applicable Telegram	3, 9, 111

Note:

*1 The default value of Pt317 is 3000 and Pt386 is 20. These are the velocity commands which 100% correspond to the controller commands.

6.2 Velocity limit setting

To change the velocity limit, set Pt316 for rotary motors, and set Pt385 for linear motors.

Pt No.	Pt316	PNU Number	0x2316		
Data type	Unit16	Setting Range	0~65535	Default	10000
Name	Maximum motor velocity (rotary servo motor)	Unit	1 rpm	Applicable Motor	Rotary
Effective	After power on	Attribute	Value	Applicable Telegram	3, 9, 111

Pt No.	Pt385	PNU Number	0x2385		
Data type	Unit16	Setting Range	0~100	Default	50
Name	Maximum motor velocity (linear servo motor)	Unit	100 mm/s	Applicable Motor	Linear
Effective	After power on	Attribute	Value	Applicable Telegram	3, 9, 111

6.3 Torque limit setting

To change the torque limit, set Pt402 and Pt403 for rotary motors, and set Pt483 and Pt484 for linear motors.

Pt No.	Pt402	PNU Number	0x2402		
Data type	Unit16	Setting Range	0~800	Default	800
Name	Forward torque limit	Unit	1%	Applicable Motor	Rotary
Effective	Immediately	Attribute	Value	Applicable Telegram	3, 9, 111

Pt No.	Pt403	PNU Number	0x2403		
Data type	Unit16	Setting Range	0~800	Default	800
Name	Reverse torque limit	Unit	1%	Applicable Motor	Rotary
Effective	Immediately	Attribute	Value	Applicable Telegram	3, 9, 111

Pt No.	Pt483	PNU Number	0x2483		
Data type	Unit16	Setting Range	0~800	Default	30
Name	Forward force limit value for internal force limit (linear servo motor)	Unit	1%	Applicable Motor	Linear
Effective	Immediately	Attribute	Value	Applicable Telegram	3, 9, 111

Pt No.	Pt484	PNU Number	0x2484		
Data type	Unit16	Setting Range	0~800	Default	30
Name	Reverse force limit value for internal force limit (linear servo motor)	Unit	1%	Applicable Motor	Linear
Effective	Immediately	Attribute	Value	Applicable Telegram	3, 9, 111

6.4 Quick stop

To change the quick stop deceleration time, set Pt30A for speed mode, and set Pt538 for position mode.

Pt No.	Pt30A	PNU Number	0x230A		
Data type	Unit16	Setting Range	0~10000	Default	0
Name	Deceleration time for servo off and forced stop	Unit	1 ms	Applicable Motor	Rotary Linear
Effective	Immediately	Attribute	Value	Applicable Telegram	3

Pt No.	Pt538	PNU Number	0x2538		
Data type	Unit16	Setting Range	2~10000	Default	10
Name	Program P2P emergency deceleration time	Unit	1 ms	Applicable Motor	Rotary Linear
Effective	Immediately	Attribute	Value	Applicable Telegram	9, 111

6.5 Coast stop

The coast stop method could be set by Pt001 = t.□□□X.

Parameter		Servo Motor Stopping Method	Servo Motor State After Stop	Effective
Pt001	t.□□□0 (default)	Dynamic brake	Dynamic brake	After power on
	t.□□□1		Free run	
	t.□□□2	Free run	Free run	

6.6 MDI submode

6.6.1 MDI setpoints

When working with MDI submode, MDI setpoints should be set as below. For the details of control unit, refer to section 6.11 **Electronic gear ratio** in “E1 Series Servo Drive User Manual”.

MDI setpoint	
MDI_TARPOS	1h = 1 control unit
MDI_VELOCITY	1h = 1000 control unit/min
MDI_ACC	4000h = 100%
MDI_DEC	4000h = 100%
OVERRIDE	4000h = 100%

The acceleration reference value could be set by Pt534.

Pt No.	Pt534	PNU Number	0x2534		
Data type	Unit16	Setting Range	2~10000	Default	100
Name	Program P2P acceleration time	Unit	1 ms	Applicable Motor	Rotary Linear
Effective	Immediately	Attribute	Value	Applicable Telegram	9, 111

The deceleration reference value could be set by Pt537.

Pt No.	Pt537	PNU Number	0x2537		
Data type	Unit16	Setting Range	2~10000	Default	100
Name	Program P2P deceleration time	Unit	1 ms	Applicable Motor	Rotary Linear
Effective	Immediately	Attribute	Value	Applicable Telegram	9, 111

6.7 Homing

Only Telegram 9 and Telegram 111 can apply internal homing of drive. For the details of homing methods and parameters setting, refer to section 8.11 **Internal homing** in “E1 Series Servo Drive User Manual”.

6.8 JOG

Only Telegram 9 and Telegram 111 can apply jog. Set STW1.8 = 1 to jog positive; set STW1.9 = 1 to jog negative. To change the jog velocity, set Pt533 for rotary motors, and set Pt585 for linear motors.

Pt No.	Pt533	PNU Number	0x2533		
Data type	Unit16	Setting Range	1~10000	Default	600[*1]
Name	Program P2P velocity	Unit	1 rpm	Applicable Motor	Rotary
Effective	Immediately	Attribute	Value	Applicable Telegram	9, 111

Note: [*1] If direct drive motor is used, the default value of Pt533 is 60 rpm.

Pt No.	Pt585	PNU Number	0x2585		
Data type	Unit16	Setting Range	1~10000	Default	50
Name	Program P2P velocity (linear servo motor)	Unit	1 mm/s	Applicable Motor	Linear
Effective	Immediately	Attribute	Value	Applicable Telegram	9, 111

The acceleration reference value could be set by Pt534.

Pt No.	Pt534	PNU Number	0x2534		
Data type	Unit16	Setting Range	2~10000	Default	100
Name	Program P2P acceleration time	Unit	1 ms	Applicable Motor	Rotary Linear
Effective	Immediately	Attribute	Value	Applicable Telegram	9, 111

The deceleration reference value could be set by Pt537.

Pt No.	Pt537	PNU Number	0x2537		
Data type	Unit16	Setting Range	2~10000	Default	100
Name	Program P2P deceleration time	Unit	1 ms	Applicable Motor	Rotary Linear
Effective	Immediately	Attribute	Value	Applicable Telegram	9, 111

The PROFIdrive JOG mode moving direction inverse reference value can be set by Pt53A.

Pt No.	Pt53A	PNU Number	0x253A		
Data type	Unit16	Setting Range	0~1	Default	0
Name	PROFIdrive JOG mode moving direction inverse setting	Unit	-	Applicable Motor	All
Effective	Immediately	Attribute	Setup	Applicable Telegram	9, 111

6.9 Absolute encoder initialization

When using a rotary absolute encoder, it is necessary to clear multi-turn data at the first start up after installing the battery. There are two types of data in a rotary absolute encoder: single-turn data and multi-turn data. The single-turn data shows the position of the motor's rotation within a single turn. The multi-turn data counts the number of the turns, and the backup is stored by the battery.

■ Method 1

For relative information of function blocks usage, please refer to section 3.4 **Initialize absolute encoder** in "Function Blocks Application Manual E1 PROFINET Drive with Siemens TIA Portal."

■ Method 2

Adopt the following steps to clear multi-turn data.

Step 1. Disable the motor.

Step 2. Set 3200h to 1.

Step 3. Wait until 3200h changes to 4 (the command is successfully executed).

Step 4. Reset the drive (set 3215h to 1).

Note: For the parameter attribute of 3200h and 3215h, please refer to section 4.2 **PNU dictionary table**.

■ Definition of object 3200h

Table 6.9.1

Value	Definition
0	Not in operation.
1	Send the command of clearing multi-turn data.
2	The command of clearing multi-turn data is being executed.
4	The command of clearing multi-turn data is successfully executed.
16	Do not clear multi-turn data when the motor is enabled. Please disable the motor again before issuing the command again.
32	Fail to execute the command of clearing multi-turn data.

Note:

(1) This function is only supported in firmware version 2.8.16 or above.

(2) Users can directly download the attachment of "Function Blocks Application Manual E1 PROFINET Drive with Siemens TIA Portal" for usage.

(This page is intentionally left blank.)

7. Appendix

7.1 Configure PROFINET communication by Thunder

Thunder offers users an interface to set the parameters related to PROFINET communication. Select **Tools** in the menu bar, and click **PROFINET setup** to open “PROFINET setup” window.

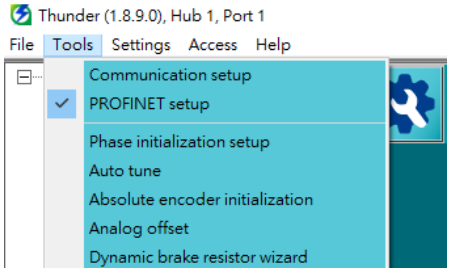


Figure 7.1.1

Figure 7.1.2

Drive Status displays the current activated setting of servo drive. To change the current setting, set **Device name**, **Protocol**, **Telegram**, **Velocity reference value** and **Fine resolution**. Then, click **Send to drive** to send the new setting to servo drive. The new setting will be activated after the process is completed.

Note:

(1) The device name must be unique within a PROFINET network.

(2) Velocity reference value and **Fine resolution** should always correspond with the reference speed and the encoder resolution configured in controller if application requires. Otherwise, these can be ignored.

(3) For the setting of **Fine resolution**, users can refer to “Application Note E1 PROFINET Drive Complete Setup with Siemens TIA Portal.”