



INSTALLATION AND OPERATING INSTRUCTIONS

Rotary module

REH2000

Modbus RTU

Software version SWA000218_A00

DDOC02749

THE KNOW-HOW FACTORY

Glossary

Term	Explanation
Active warning	The product displays a warning if there is a possibility that an error will occur. The product can continue to move when there is an active warning. Causes that can lead to an error, e.g., incorrect parameters or operating conditions, should be taken into account and eliminated.
Active error	The product displays an error, for example, because previous warnings were not taken into account. Errors lead to a standstill of the motor and the power level. When an error occurs, the product is set to the <i>FAULT</i> or <i>LOCKED</i> status and the product can no longer move. Errors remain visible until they are acknowledged, even if the cause has been eliminated.
Saved error	The product displays a saved error when the cause of an error is eliminated but the error has still not been acknowledged.
NVM/ non-volatile memory	Non-volatile memory where information is stored permanently, including when the product is not in operation or not energized.
PDI/ Process data input	Data that are transmitted to the control system for each product cycle.
PDO/ Process data output	Data that are transmitted from the control system to the product for each product cycle.
Power cycle	Cycle in which the logic and actuator voltage of the product is switched off for min. 5 s, then switched back on.
Control system	Device that is used to control a product. A control system could be a PLC, for example.

Data type	Explanation
UINT8	unsigned 8 bits
UINT16	unsigned 16 bits
UINT32	unsigned 32 bits
INT8	signed 8 bits
INT16	signed 16 bits
INT32	signed 32 bits
FLOAT	Simple floating point number with 32 bits (see IEEE 754)

Content

1	Supporting documents	5
1.1	Notices and graphics in the installation and operating instructions	6
2	Safety notices	7
3	Proper use	8
4	Personnel qualification	9
4.1	Electricians	9
4.2	Specialists	9
4.3	Instructed personnel	9
4.4	Service personnel	9
4.5	Additional qualifications	9
5	Product description	10
6	Functional description	10
6.1	LED display	10
7	Technical data	11
8	Accessories/scope of delivery	11
9	Transportation/storage/preservation	11
10	Installation	12
10.1	Installing the product	13
10.2	Installing the power supply	14
10.2.1	Pin assignment	14
10.2.1.1	Connecting line	14
10.2.1.2	Rotary feed-through	14
10.3	Static charge	15
10.4	Heat dissipation	15
10.5	Installing accessories	16
10.5.1	Installing the axis compensation module	16
11	Commissioning	17
11.1	Example code	17
11.2	Finite state machine	18
11.3	Communication	20
11.3.1	Communication settings	20
11.3.1.1	Baud rate	20
11.3.1.2	Station address	20
11.3.1.3	Reestablishing communication	20
11.3.2	Function codes and conventions	21
11.3.2.1	32-bit words	21
11.3.2.2	String	21
11.3.3	Cyclical data	22
11.3.3.1	Process data output	22
11.3.3.2	Process data input	24
11.3.4	PDO.CONTROL	25
11.3.5	PDO.COMMAND	25
11.3.6	Move commands	26
11.3.7	PDO.POSITION	28
11.3.7.1	PDO.TORQUE	29
11.3.7.2	PDO.VELOCITY	29
11.3.7.3	PDI.STATUS	29
11.3.7.4	PDI.ACTUAL_POSITION	30

11.3.7.5	PDI.ERROR	30
11.3.7.6	PDI.WARNING.....	30
11.4	Status of the LEDs	31
11.5	Quickstart	32
11.6	Functions and move commands	33
11.6.1	Enable/start-up	33
11.6.2	Easy Jog (0x0001, 0x0002).....	34
11.6.3	Disable.....	35
11.6.4	Troubleshooting in the LOCKED state	35
11.6.5	Troubleshooting in the FAULT state	36
11.6.6	Positioning.....	36
11.6.6.1	Absolute Positioning Multiturn (0x0104).....	37
11.6.6.2	Relative Positioning Multiturn (0x0204)	38
11.6.6.3	Absolute Positioning Singleturn (0x0304, 0x0384, 0x0604)	38
11.6.7	Close at Position	39
11.6.7.1	Close at Position: Absolute Positioning (0x0208).....	41
11.6.7.2	Close at Position: Relative Positioning (0x0808).....	41
11.6.8	Close up to position.....	42
11.6.8.1	Close up to position: Absolute Positioning (0x0108).....	43
11.6.8.2	Close up to position: Relative Positioning (0x0708)	43
11.6.9	Velocity mode (0x0504, 0x0584).....	44
11.6.10	Stop.....	45
11.6.11	Reset Multiturn Position (0x0F10).....	46
12	Acyclic data	47
12.1	General	47
12.2	Rotary module.....	48
13	Error diagnosis	49
13.1	Error	49
13.2	Warnings.....	50
14	Maintenance	52
15	Decommissioning/disposal	52
16	Appendix	53
16.1	Movement finite state machine.....	53
16.2	Software copyright notices	54
16.2.1	FreeModbus.....	54
17	RoHS declaration.....	55
18	Declaration of Incorporation (Machinery Directive)	56
19	Declaration of Incorporation (Machinery Regulation).....	57
20	Declaration of Conformity	58

1 Supporting documents

NOTICE



Read through the installation and operating instructions before installing or working with the product.

The installation and operating instructions contain important notes for your personal safety. They must be read and understood by all persons who work with or handle the product during any phase of the product lifetime.



The documents listed below are available for download on our website www.zimmer-group.com:

- Installation and operating instructions
- Catalogs, drawings, CAD data, performance data
- Information on accessories
- Technical data sheets
- General Terms and Conditions (GTCs), including warranty information.

⇒ Only the documents currently available on the website are valid.

INFORMATION



These installation and operating instructions are valid for software version SWA000218_A00.

► Contact Customer Service for installation and operating instructions for older software versions.

In these installation and operating instructions, “product” replaces the product designation on the title page.

1.1 Notices and graphics in the installation and operating instructions

DANGER



This notice warns of an imminent danger to the life and health of people. Ignoring these notices can lead to serious injury or even death.

- You absolutely must comply with the described measures for avoiding these dangers.
- ⇒ The warning symbols are assigned according to the type of danger.

WARNING



This notice warns of a situation that is potentially hazardous to personal health. Ignoring these notices can cause serious injury or damage to health.

- You absolutely must comply with the described measures for avoiding these dangers.
- ⇒ The warning symbols are assigned according to the type of danger.

CAUTION



This notice warns of a situation that is potentially hazardous to people. Ignoring these notices can cause minor, reversible injuries.

- You absolutely must comply with the described measures for avoiding these dangers.
- ⇒ The warning symbols are assigned according to the type of danger.

NOTICE



This notice warns of possible material or environmental damage. Ignoring these notices can result in damage to the product or the environment.

- You absolutely must comply with the described measures for avoiding these dangers.
- ⇒ The warning symbols are assigned according to the type of danger.

INFORMATION



This category contains useful tips for handling the product efficiently. Failure to observe these tips will not result in damage to the product. This information does not include any information relevant to health or workplace safety.

2 Safety notices

CAUTION



Risk of injury and material damage in case of non-compliance

The product is state-of-the-art.

The following are examples of situations in which the product may cause a hazard:

- The product is not properly installed, used or maintained.
- The product is not used for its designated purpose.
- The locally applicable regulations, laws, directives or guidelines are not observed.
- ▶ The product may only be used in accordance with these installation and operating instructions and the product's technical data. Any changes or additions to the intended use of the product, as well as modifications to the product, such as those in the following examples, require the written permission of the manufacturer:
 - Use of the product under extreme conditions, such as aggressive fluids or abrasive dusts
 - Additional drilled holes or threads
- ⇒ Zimmer GmbH accepts no liability for any damage caused by improper use. The operator bears sole responsibility.
- ▶ Make sure that the power supply is disconnected before you mount, adjust, modify, maintain or repair the product.
- ▶ Whenever work is carried out on the product, make sure that the product cannot be actuated by mistake.
- ▶ Perform maintenance tasks, retrofitting or attachment work outside of the machine's danger zone when possible.
- ▶ Do not reach into the operational range of the product.
- ▶ Observe the specified maintenance intervals and specifications regarding the quality of the operating material.
- ▶ Always perform maintenance at the required intervals.
- ▶ When using the product under extreme conditions, adjust the maintenance interval according to the degree of soiling.
- ▶ Check the completeness and tightening torques of all mounting screws.

CAUTION



Notes and handling regulations for electrostatically sensitive components

Electrostatically sensitive components are individual components, integrated circuits or assemblies that can be damaged by electrostatic fields or electrostatic discharge.

- ▶ When handling electrostatic components, make sure that persons, the work area and packaging are all fully grounded.
- ▶ Touch electronic components only in appropriately identified areas with conductive flooring and only if:
 - You are grounded by means of special bracelets.
 - You wear shoes that are suitable and approved for the discharge of electrostatic charges.
- ▶ Do not bring electronic assemblies into contact with plastics and parts of clothing that have plastic content.
- ▶ Only place electronic assemblies on conductive surfaces.
- ▶ Do not install electronic assemblies in the vicinity of data back-up devices or monitors (monitor distance > 100 mm).
- ▶ Only perform measurements on electronic assemblies if:
 - The measuring instrument is grounded (e.g. via a ground conductor).
 - The measuring head is momentarily discharged before measuring with a floating measuring instrument.

NOTICE



Cybersecurity

- ▶ The operator is responsible for ensuring that unauthorized persons do not have access to the product, either physically or remotely.

3 Proper use

NOTICE



The product is only to be used in its original state with its original accessories, with no unauthorized changes and within the stipulated parameter limits and operating conditions.

Any other or secondary use is deemed improper.

- ▶ Operate the product only in compliance with the associated installation and operating instructions.
- ▶ Operate the product only when it is in a technical condition that corresponds to the guaranteed parameters and operating conditions.
- ⇒ Zimmer GmbH accepts no liability for any damage caused by improper use. The operator bears sole responsibility.

- The product is designed exclusively for electric operation using a 24 V DC supply voltage.
- The product is intended for use in enclosed rooms to grip with positioning accuracy.
- The product is not suitable for clamping workpieces during a machining process.
- The product must always be mounted on thermally conductive materials.
- The product is intended for industrial use.
- Direct contact with perishable goods/food is not permitted.
- The product is not suited for use in a potentially explosive atmosphere.

4 Personnel qualification

WARNING



Injuries and material damage due to inadequate qualification

If inadequately qualified personnel perform work on the product, this can cause serious injuries and significant material damage.

- ▶ All work on the product must be performed by qualified personnel.
- ▶ Before working with the product, read the document in its entirety and make sure that you have understood everything.
- ▶ Observe country-specific accident prevention regulations and the general safety notices.

The following qualifications are a prerequisite for performing various types of work on the product.

4.1 Electricians

Electricians are able to perform work on electrical systems, can recognize and avoid possible dangers and know the relevant standards and provisions due to their technical training, knowledge and experience.

4.2 Specialists

Specialists are able to perform the assigned work, can recognize and avoid possible dangers and know the relevant standards and provisions due to their technical training, knowledge and experience.

4.3 Instructed personnel

Instructed personnel have been trained by the operating company on the tasks and possible dangers of improper behavior.

4.4 Service personnel

Service personnel are able to perform the assigned work and can recognize and avoid possible dangers due to their technical training, knowledge and experience.

4.5 Additional qualifications

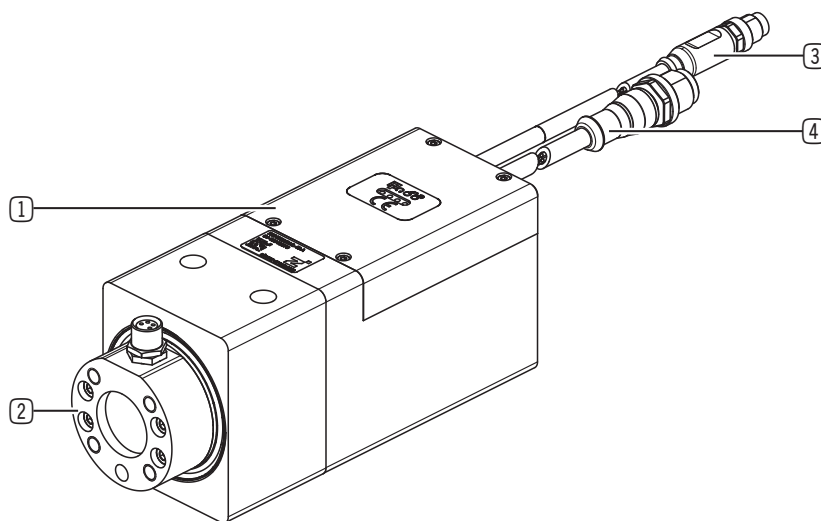
Personnel who work with the product must be familiar with the valid safety regulations and laws as well as the standards, guidelines and laws listed in this document.

Personnel who work with the product must have facility-issued authorization to commission, program, configure, operate, maintain and also decommission this product.

5 Product description

This product is a rotary module. It can communicate with the control system via a Modbus RTU interface.

The state can be controlled using the finite state machine and the movement control.



- ① Stator
- ② Rotor
- ③ Rotary feed-through
- ④ Connecting line

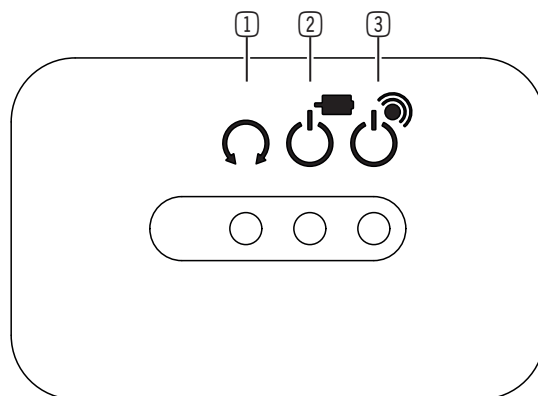
6 Functional description

In the rotary module, the force of the motor is transferred over a gear drive to a shaft. This causes it to rotate.

The product can be easily integrated into existing control systems and controlled via Modbus RTU (RS485).

The gripping force and torque can be optimally adapted to the workpiece conditions via Modbus RTU (RS485).

6.1 LED display



- ① Rotary module
- ② Power supply
- ③ Logic supply

7 Technical data

INFORMATION



- ▶ You can find the information in the technical data sheet on our website.
- This data varies within the series, depending on the specific design.
- ▶ Please contact Customer Service if you have any questions.

8 Accessories/scope of delivery

INFORMATION



- If any accessories not sold or authorized by Zimmer GmbH are used, the function of the product cannot be guaranteed. The accessories from Zimmer GmbH are specifically tailored to the individual products.
- ▶ For information on optional accessories and those included in the scope of delivery, refer to our website.

9 Transportation/storage/preservation

- ▶ Transport and store the product only in the original packaging.
- ▶ During transport, make sure that no uncontrolled movements can occur if the product is already mounted on the higher-level machine unit.
 - ▶ Prior to commissioning and after transport, check all power and communication connections as well as all mechanical connections.
- ▶ Observe the following points when storing the product for longer periods of time:
 - ▶ Keep the storage location as dust-free and dry as possible.
 - ▶ Avoid temperature fluctuations.
 - ▶ Avoid wind, drafts and formation of condensation.
 - ▶ Avoid direct sunlight.
- ▶ Clean all components until all contamination has been removed.
- ▶ Visually inspect all components.
- ▶ Remove any foreign objects.
- ▶ Remove potential corrosion spots properly.
- ▶ Seal electrical connections with suitable covers.

10 Installation

WARNING



Risk of injury due to uncontrolled movement

Risk of injury in case of uncontrolled movements of the machine or system into which the product is to be installed.

- ▶ Switch off the power supply of the machine before all work.
- ▶ Secure the power supply against being switched on unintentionally.
- ▶ Check the machine for any residual energy that may be present.

CAUTION



Risk of injury due to uncontrolled movement

Risk of injury in the event of uncontrolled movement of the product when the power supply is connected.

- ▶ Switch off the power supply to the product before all work.
- ▶ Secure the power supply against being switched on unintentionally.
- ▶ Check the product for any residual energy that may be present.

Assembly requirements

Permissible flatness tolerance [mm]	Mounting surface length [mm] < 100	0.02
	Mounting surface length [mm] > 100	0.05
Strength class of the mounting screw	8.8	

INFORMATION



Further installation information:

- The mounting screws are not included in the scope of delivery.

- ▶ Install the product on an appropriate mounting surface in accordance with the flatness specifications.
- ▶ Make sure that the mounting piece is sufficiently rigid.
- ▶ Ensure the cleanliness of the connection surfaces.
- ▶ Please note the permitted tightening torques of the mounting screws at <https://www.zimmer-group.com/en-us/td>.

10.1 Installing the product

INFORMATION

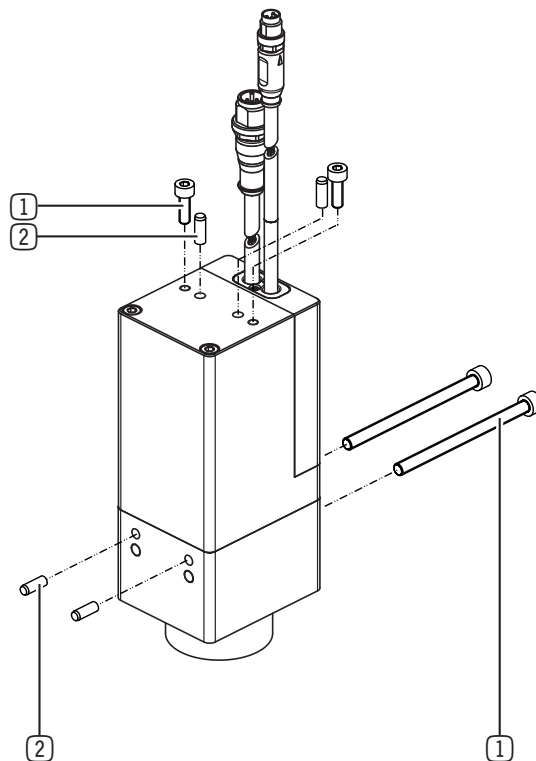


When using the associated axis compensation module, this must be installed first.

► Please note section "10.5.1 Installing the axis compensation module".

The product can be installed from multiple sides. When using the axis compensation module, only installation from the front is possible.

- Insert the straight pins into the designated fits on the product.
- Position the product on the mounting piece using straight pins.
- Mount the product on the mounting piece by using the appropriate mounting screws.



- ① Mounting screw
- ② Straight pin

10.2 Installing the power supply

WARNING



Risk of injury due to electrical voltage

Electric shocks can cause serious injuries as a result of touching parts carrying voltage.

- ▶ Switch off the energy supply of the machine before all work.
- ▶ Secure the energy supply against being switched on unintentionally.

NOTICE



Material damage in case of non-compliance

The cable mounted on the product can be subjected to a torsional angle of $\pm 50^\circ$.

- ▶ Do not route the cable so that it is strained.
 - ▶ You must meet the minimum bending radius of 10x the outer diameter.
 - ▶ Secure free-hanging cables to prevent excessive motion loads or pinching.
 - ▶ The contacts of the energy supply must be dry, clean and undamaged at all times.
- ⇒ Damage to the contacts can result in malfunction of the product.

10.2.1 Pin assignment

10.2.1.1 Connecting line

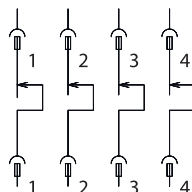
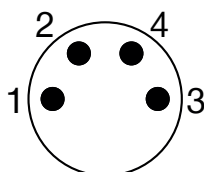
Pin	Color	Function	Explanation	Male connector, M12
1	White	RS485 -	-	
2	Brown	RS485 +	-	
5	Gray	Power +	24 V actuator voltage	
8	Red	Power -	0 V actuator voltage	

10.2.1.2 Rotary feed-through

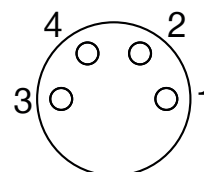
The male connector of the rotary feed-through is connected to the female connector on the rotor via a slip ring. The lines are available for customers to use freely.

- ▶ Please observe the limit values listed in the technical data sheet.

Male connector, M8



Female connector, M8



10.3 Static charge

NOTICE



Material damage from static charge

The movement of the gripper jaws creates low voltages as a result of static charging. These charges cannot be dissipated if the product is mounted on an insulating surface and if discharge is not possible through the workpiece.

- ▶ Please note that ESD-sensitive parts can be damaged if they come into contact with the product.
- ▶ Ground the product for applications that require high EMC resistance.

10.4 Heat dissipation

NOTICE



Material damage from overheating of the product

If the product is operated under a very high ambient temperature and with fast clock cycles on an ongoing basis, this can reduce its service life.

- ▶ If the product is exposed to high ambient temperatures, always install it on heat-conducting materials.
- ▶ Reduce the load as the temperature increases.

10.5 Installing accessories

NOTICE

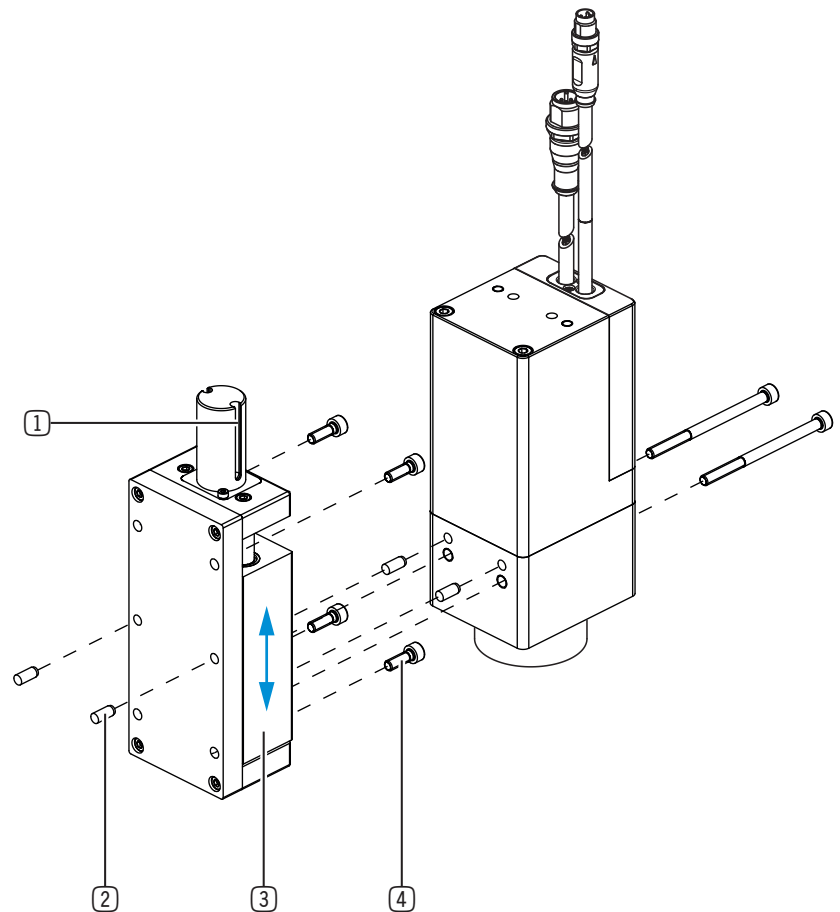


Material damage in case of non-compliance

- ▶ Before installing the accessories, check whether they are suitable for use with the selected variant.
- ▶ You can find more information on our website.
- ▶ Please contact Customer Service if you have any questions.

10.5.1 Installing the axis compensation module

- ▶ Insert the straight pins into the designated fits on the axis compensation module.
- ▶ Position the axis compensation module on the mounting piece using straight pins.
- ▶ Install the axis compensation module on the mounting piece with the mounting screws.
 - ▶ Move the compensation module to install the lower mounting screws.
- ▶ If necessary, install the sensors in the sensor slot.
 - ▶ You can find more information on our website.



- ① Sensor slot
- ② Straight pin
- ③ Compensation module
- ④ Mounting screw

11 Commissioning

WARNING



Risk of injury due to uncontrolled movement

Injuries can result from being caught or wound up during the motion sequence.

- ▶ Do not reach into the operational range of the product.
- ▶ Never stand within the operational range of the product.
- ▶ Operate the product from a safe distance behind a safety guard.
- ▶ Wear suitable personal protective equipment.

CAUTION



Risk of injury due to uncontrolled movement

Injuries can result due to crushing during the motion sequence.

- ▶ Do not reach into the operational range of the product.
- ▶ Never stand within the operational range of the product.
- ▶ Operate the product from a safe distance behind a safety guard.

NOTICE



Exclusion of liability regarding unauthorized use

This product may only be used for the proper use described in this documentation.

In particular, the following actions are prohibited:

- Tampering with the product or software
- Bypassing or disabling security mechanisms
- Unauthorized access to the product or interfaces
- ▶ Please note that the manufacturer assumes no liability for any resulting damage.
- ▶ Please note that warranty and support claims will be void in the event of misuse.

11.1 Example code

It is assumed in the examples that communication with the module remains consistent.

This means that the variables must be updated simultaneously.

In the following example, *PDO.COMMAND* and *PDO.POSITION* are updated simultaneously via Modbus RTU using the *APPLY_PDO()* function.

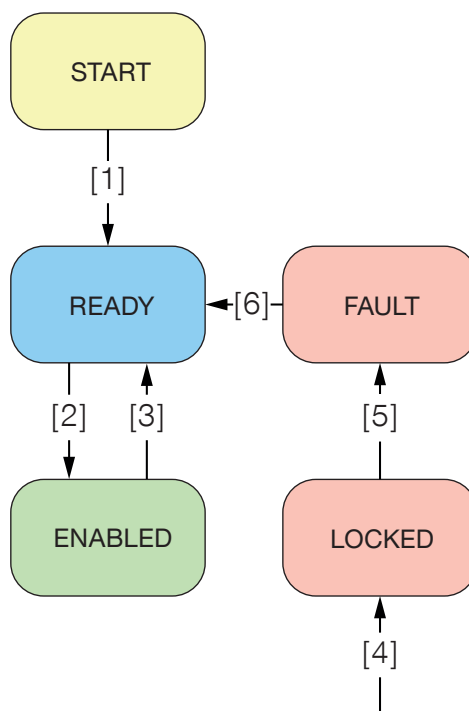
```
CASE iStep OF
  0:
    PDO.COMMAND := 0;
    PDO.POSITION := 0;
    cnt := 0;
    iStep := 10;

  10:
    cnt := cnt + 1;
    IF (cnt > 100) THEN
      PDO.COMMAND := 16#104;
      PDO.POSITION := 1000;
      iStep := 20;
    END_IF

  20:
    ...

END_CASE;
...
APPLY_PDO(PDO);
```

11.2 Finite state machine



State	Description
START	The module is supplied with power and Modbus communication is supported. The internal initialization sequence runs.
READY	The motor and the power level are switched off. There are no active or saved errors. The module is ready to be switched on.
ENABLED	The motor and the power level are switched on. The module can execute movement commands.
LOCKED	An active or saved error. The motor and the power level are switched off. <i>PDO.CONTROL</i> and <i>PDO.COMMAND</i> must be set to 0 before the error can be acknowledged.
FAULT	An active or saved error. The motor and the power level are switched off. This state can trigger an acknowledgment query.

Transition	Description
[1]	Automatic transition when the internal initialization sequence is completed.
[2]	The <i>ENABLE</i> command is received. (<i>PDO.CONTROL.enable</i> = 1)
[3]	The <i>ENABLE</i> command is reset. (<i>PDO.CONTROL.enable</i> = 0)
[4]	An error occurred.
[5]	<i>PDO.CONTROL</i> and <i>PDO.COMMAND</i> are set to 0.
[6]	There are no active or saved errors. (<i>PDI.ERROR</i> = 0)

The state of the modules can be determined using *PDI.STATUS* .

State	PDI.STATUS			
	control_active	enabled	error	busy
START	0	0	0	0
READY	1	0	0	0
ENABLED	1	1	0	-
FAULT	1	0	1	0
LOCKED	1	0	1	1

The following example shows how the state in the PDIs can be retrieved.

```

IF NOT PDI.STATUS.control_active THEN
    state := START;

ELSIF PDI.STATUS.enabled THEN
    state := ENABLED;

ELSIF PDI.STATUS.error AND PDI.STATUS.busy THEN
    state := LOCKED;

ELSIF PDI.STATUS.error THEN
    state := FAULT;

ELSE
    state := READY;

END_IF
    
```

11.3 Communication

11.3.1 Communication settings

The product functions as a Modbus RTU server.

Communication takes place via a serial interface in accordance with **EIA/TIA-485 (RS-485)**.

The transmission mode is 8E1:

- Data length: 8 bits
- Parity: even
- Stop bits: 1
- Flow control: none

11.3.1.1 Baud rate

The *BAUDRATE* is set to 115,200 bits/s by default.

The baud rate can be adjusted using the *BAUDRATE* parameter (see section "12 Acyclic data").

NOTICE



Malfunctions in case of non-compliance

- Make sure that all bus participants are operating at the same baud rate.

11.3.1.2 Station address

The station address is the unique identifier of a Modbus RTU server.

The product's station address is set to 1 by default.

The station address can be adjusted using the *STATION_ADDRESS* parameter (see section "12 Acyclic data").

NOTICE



Malfunctions in case of non-compliance

- Make sure that
 - The station address on the bus is unique.
 - The station address has a value between 1 and 247.

INFORMATION



- Address 0 is reserved for simultaneously addressing all servers (broadcast).
- The station address may also be referred to as the UnitID, SlaveID or Modbus address.

11.3.1.3 Reestablishing communication

If a product's baud rate and station address are unknown, communication can be reestablished as follows:

- Disconnect all servers except the unknown server from the bus.
- For each possible value of *BAUDRATE*, perform the following steps:
 - Send the desired baud rate using *BAUDRATE* to address 0 (broadcast).
 - Send the desired address using *STATION_ADDRESS* to address 0 (broadcast).
- Restart the product by disconnecting it from the power supply and reconnecting it.

11.3.2 Function codes and conventions

Refer to the following table for the Modbus RTU function codes being used:

Function code	Name	Description
FC03	Read Holding Registers	Reads one or more Modbus registers
FC04	Read Input Register	Reads one or more Modbus registers
FC06	Write Single Register	Writes to one Modbus register
FC16	Write Multiple Registers	Writes to one or more Modbus registers
FC23	Read/Write Multiple Registers	Writes to one or more Modbus registers and then reads one or more Modbus registers

INFORMATION



- For a detailed description of the commands and Modbus RTU data packets, please refer to the following link:
- <https://www.modbus.org/file/secure/modbusprotocolspecification.pdf>

11.3.2.1 32-bit words

32-bit words such as UINT32, INT32 or FLOAT must be distributed across the 16-bit registers in little-endian format with a byte swap.

► Consider the following example:

- Value to be written: 0x12345678 (305419896)
- ⇒ Modbus data packet: 01 10 00 01 00 02 04 56 78 12 34 AE 85

Part of the data packet	Description	Value
01	Station address	0x01 (1)
10	Function code	0x10 (16) - Write Multiple Registers
00 01	Start address	Physical: 0x0001 (1)
00 02	Number of data registers	0x0002 (2)
04	Number of data bytes	0x04 (4)
56 78 12 34	Data	0x5678 (22136), 0x1234 (4660)
AE 85	CRC (Checksum)	0xAE85 (44677)

11.3.2.2 String

For the STRING data type, the data must be interpreted in big-endian order. Encoding is done using ASCII.

► Consider the following example:

- Value to be read: Hello
- ⇒ Modbus data packet: 01 03 0A 48 65 6C 6C 6F 00 00 00 00 93 FE

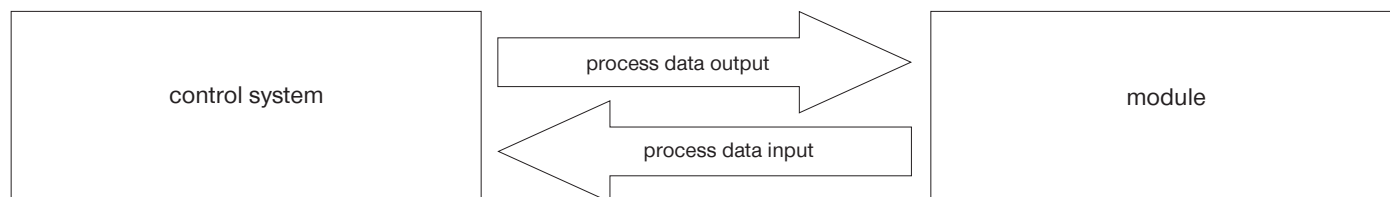
Part of the data packet	Description	Value
01	Station address	0x01 (1)
03	Function code	0x03 (3)
0 A	Number of data bytes	0x0A (10)
48 65 6C 6C 6F 00 00 00 00 00	Data	0x4865 (18533 - "He"), 0x6C6C (27756 - "ll"), 0x6F00 (28416 - "o\0"), 0x0000 (0 - "\0\0"), 0x0000 (0 - "\0\0")
93 FE	CRC (Checksum)	0x93FE (37886)

11.3.3 Cyclical data

There is an option to control the product only with the process data that is transmitted in each cycle.

It is recommended to read/write the process data in a single pass.

The data is grouped by process data input or process data output so that it can be read or written with a single request.



11.3.3.1 Process data output

NOTICE



Malfunctions in case of non-compliance

If *PDO.COMMAND* and *PDO.CONTROL* are not set to 0 during startup on, an error is generated and the module is set to the *LOCKED* status.

- Set *PDO.COMMAND* and *PDO.CONTROL* to 0 during startup.

	Name	Data type	Mode Address register	Description
PDO	CONTROL	UINT8	0x0080 (128)	► For more information, refer to section "11.3.4 PDO.CONTROL".
	RESERVE_1	UINT8	0x0081 (129)	Reserved ► Set this value to 0.
	COMMAND	UINT16	0x0082 (130)	► For more information, refer to section "11.3.5 PDO.COMMAND".
	POSITION	INT32	0x0083 (131)	► For more information, refer to section "11.3.7 PDO.POSITION".
	TORQUE	UINT8	0x0085 (133)	► For more information, refer to section "11.3.7.1 PDO.TORQUE".
	VELOCITY	UINT8	0x0086 (134)	► For more information, refer to section "11.3.7.2 PDO.VELOCITY".
	RESERVE_2	UINT16	0x0087 (135)	Reserved ► Set this value to 0.

Example 1:

This requires that the module be switched on (PDO.Control.enable = 1).

The module receives the "Absolute Positioning Multiturn" command (0x0104) and sets the speed to 98 and the torque to 97. The position is set to 360°.

The corresponding Modbus RTU data packet looks like this:

01 10 00 80 00 08 10 00 01 00 00 01 04 8C A0 00 00 00 61 00 62 00 00 0C 26

Part of the data packet	Description	Value
01	Station address	0x01 (1)
10	Function code	0x10 (16)
00 80	Start address	0x0080 (128)
00 08	Number of data registers	0x0008 (8)
10	Number of data bytes	0x10 (16)
00 01 00 00 01 04 8C A0 00 00 00 61 00 62 00 00	Rotary unit data	Control: 0x01 (1)
		Reserve_1: 0x00 (0)
		Command: 0x0104 (260)
		Position: 0x8CA0 0x0000 ⇒ 0x00008CA0 (36000)
		Torque: 0x0061 (97)
		Velocity: 0x0062 (98)
		Reserve_2: 0x0000 (0)
0C 26	CRC (Checksum)	0x0C26 (3110)

Example 2:

The position of the module is set to 360°.

The corresponding Modbus RTU data packet looks like this:

01 10 00 8B 00 02 04 8C A0 00 00 90 CE

Part of the data packet	Description	Value
01	Station address	0x01 (1)
10	Function code	0x10 (16)
00 83	Start address	0x0083 (131)
00 02	Number of data registers	0x0002 (2)
4	Number of data bytes	0x04 (4)
8C A0 00 00	Rotary unit data	Position: 0x8CA0 0x0000 ⇒ 0x00008CA0 (36000)
90 CE	CRC (Checksum)	0x90CE (37070)

INFORMATION



It is possible to send incomplete packets. However, Zimmer Group GmbH recommends sending only complete packages.

11.3.3.2 Process data input

	Name	Data type	Mode Address register	Description
PDI	STATUS	UINT16	0x0000 (0)	► For more information, refer to section "11.3.7.3 PDI.STATUS".
	RESERVE_1	UINT16	0x0001 (1)	Reserved
	ERROR	UINT16	0x0002 (2)	► For more information, refer to section "11.3.7.5 PDI.ERROR".
	WARNING	UINT16	0x0003 (3)	► For more information, refer to section "11.3.7.6 PDI.WARNING".
	ACTUAL_POSITION	INT32	0x0004 (4)	► For more information, refer to section "11.3.7.4 PDI.ACTUAL_POSITION".
	RESERVE_2	INT32	0x0006 (6)	Reserved

Example 1:

All PDIs are read.

The corresponding Modbus RTU data packet looks like this:

01 04 00 00 00 08 F1 CC

Part of the data packet	Description	Value
01	Station address	0x01 (1)
10	Function code	0x03 (3)
00 00	Start address	0x0000 (0)
00 10	Number of data registers	0x0010 (16)
C1 C5	CRC (Checksum)	0xC1C5 (49605)

The corresponding response can look like this:

01 04 10 00 C3 00 00 00 00 00 00 8A 92 00 00 00 00 00 00 ED AB

Part of the data packet	Description	Value
01	Station address	0x01 (1)
04	Function code	0x04 (4)
10	Number of data bytes	0x0010 (16)
00 C3 00 00 00 00 00 00 8A 92 00 00 00 00 00 00	Register	Status: 0x00C3 (195) ⇒ Bits set for in_target, done, enabled and control_active
		Reserve_1: 0x0000 (0)
		Reserve_2: 0x0000 (0)
		Reserve_3: 0x0000 (0)
		Position: 0x8A92 0x0000 ⇒ 0x00008A92 (35474)
		Reserve_4: 0x0000 (0)
		Reserve_5: 0x0000 (0)
ED AB	CRC (Checksum)	0xEDAB (60843)

11.3.4 PDO.CONTROL

PDO.CONTROL is used to control the finite state machine and the basic functions of the module.

Bit	Name	Description
0	enable	Command <i>ENABLE</i> 1 = The module transitions from the <i>READY</i> state to the <i>ENABLED</i> state. 0 = The module transitions from the <i>ENABLED</i> state to the <i>READY</i> state.
1	reset_error	Command <i>ACK</i> 1 = The module attempts to acknowledge the error. ► For more information, refer to section "11.6.5 Troubleshooting in the <i>FAULT</i> state".
2	reserve_b2	Reserved ► Set this bit to 0.
3	reserve_b3	Reserved ► Set this bit to 0.
4	reserve_b4	Reserved ► Set this bit to 0.
5	reserve_b5	Reserved ► Set this bit to 0.
6	reserve_b6	Reserved ► Set this bit to 0.
7	reserve_b7	Reserved ► Set this bit to 0.

11.3.5 PDO.COMMAND

The movements of the module are controlled using *PDO.COMMAND*.

► For more information, please refer to sections "11.5 Quickstart" and "11.6 Functions and move commands".

INFORMATION



► Please note that movements can only be initiated when the module is in the *ENABLED* state.

11.3.6 Move commands

PDO.COMMAND																	Command
Bits															Value		
Advanced								Direction	Reserve_b6	Reserve_b5	Homing	Close	Move	Jog-		Jog+	
15	14	13	12	11	10	9	8										
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0x0000 (0)	No operation
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0x0001 (1)	Easy Jog Clockwise
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0x0002 (2)	Easy Jog Counterclockwise
0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0x0104 (260)	Absolute Positioning Multiturn
0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0x0108 (264)	Close Up to Position (Absolute positioning)
0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0x0204 (516)	Relative Positioning Multiturn
0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0x0208 (520)	Close at Position (Absolute Positioning)
0	0	0	0	0	0	1	1	0	0	0	0	0	1	0	0	0x0304 (772)	Absolute Positioning Singleturn, Clockwise
0	0	0	0	0	0	1	1	1	0	0	0	0	1	0	0	0x0384 (900)	Absolute Positioning Singleturn, Counter-clockwise
0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	0	0x0504 (1284)	Velocity Mode Clockwise
0	0	0	0	0	1	0	1	1	0	0	0	0	1	0	0	0x0584 (1412)	Velocity Mode Counterclockwise
0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0x0604 (1540)	Absolute Positioning Singleturn, Shortest
0	0	0	0	0	1	1	1	0	0	0	0	1	0	0	0	0x0708 (1800)	Close Up to Position (Relative Positioning)
0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0x0808 (2056)	Close at Position (Relative Positioning)
0	0	0	0	1	1	1	1	0	0	0	1	0	0	0	0	0x0F10 (3856)	Reset Multiturn Position
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0x8000 (32768)	Stop

Value	Command	Command Class	Position
0x0000 (0)	No Operation (NOP) Maximum Position	No operation	-
0x0001 (1)	Easy Jog Clockwise	Jog	-
0x0002 (2)	Easy Jog Counterclockwise	Jog	-
0x0104 (260)	Absolute Positioning Multiturn	Position	Absolute Multiturn
0x0108 (264)	Close Up to Position (Absolute Positioning)	Close	Absolute Multiturn
0x0204 (516)	Relative Positioning Multiturn	Position	Relative Multiturn
0x0208 (520)	Close at Position (Absolute Positioning)	Close	Absolute Multiturn
0x0304 (772)	Absolute Positioning Singleturn, Clockwise	Position	Absolute Multiturn
0x0384 (900)	Absolute Positioning Singleturn, Counter- clockwise	Position	Absolute Multiturn
0x0504 (1284)	Velocity Mode Clockwise	Velocity	-
0x0584 (1412)	Velocity Mode Counterclockwise	Velocity	-
0x0604 (1540)	Absolute Positioning Singleturn, Shortest	Position	Absolute Multiturn
0x0708 (1800)	Close Up to Position (Relative Positioning)	Close	Relative Multiturn
0x0808 (2056)	Close at Position (Relative Positioning)	Close	Relative Multiturn

11.3.7 PDO.POSITION

PDO.POSITION defines the target position for positioning commands.

The *Multiturn* mode enables the position to be determined over several revolutions.

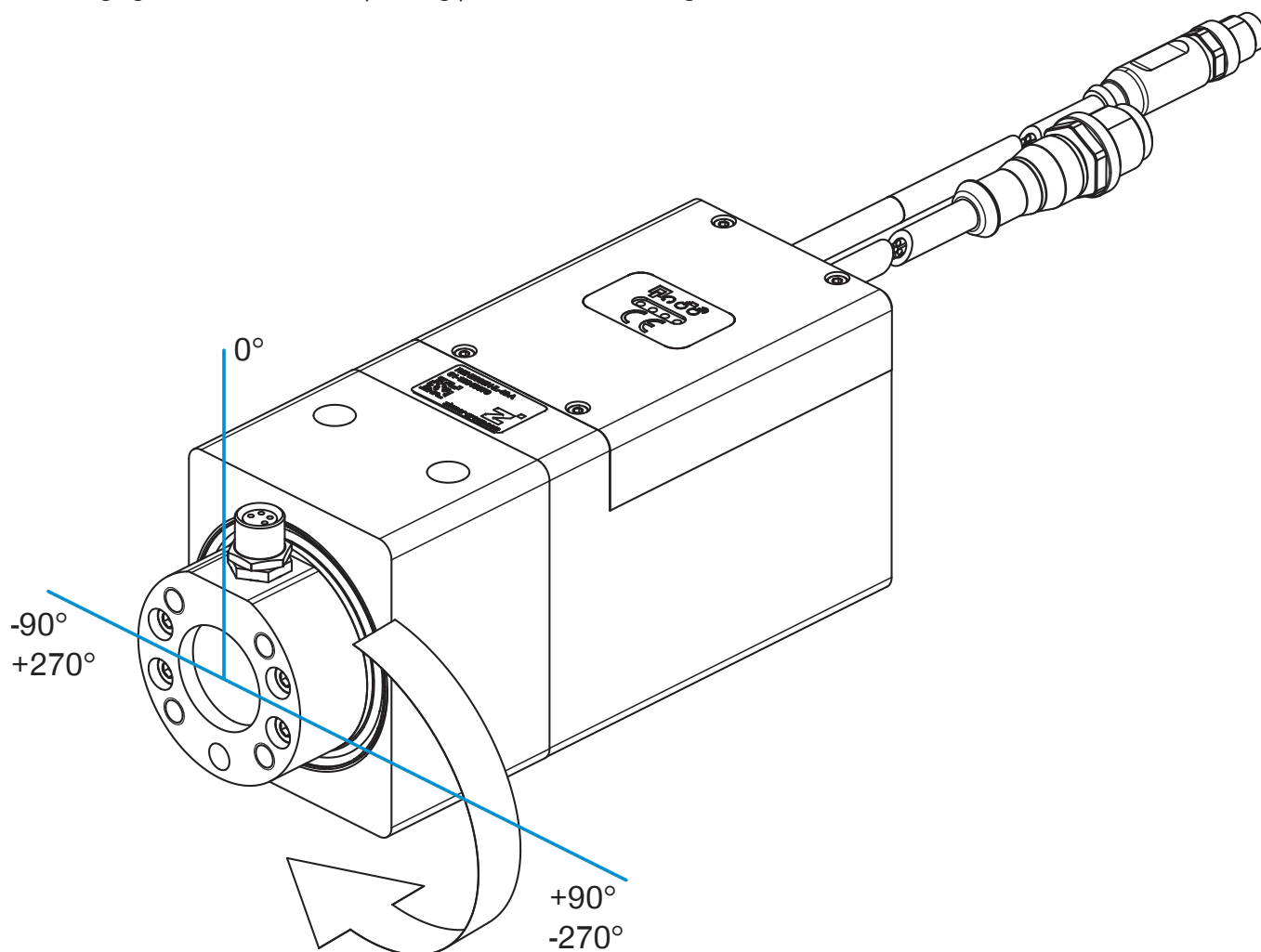
In *Singleturn* mode, the position is specified within a single revolution. Therefore, for movement commands in *Singleturn* mode, only target positions in the range from 0° to < 360° are permitted.

Data type	INT32
Value range	Singleturn mode: • 0 – 36000 Multiturn mode: • -1179648000 to 1179647999
Conversion	$PDO.POSITION = target_position_degree * 100$
Unit	0.01°

Example:

Target position	PDO.POSITION
360°	$360 * 100 = 36000 = 0x0000\ 8CA0$
-720°	$720 * 100 = 72000 = 0xFFFE\ E6C0$

The following figure shows the corresponding positions when turning clockwise.



11.3.7.1 PDO.TORQUE

PDO.TORQUE defines the torque as a percentage of the maximum torque for *Close* commands.

Data type	UINT8
Value range	15 to 100 Default value: 15 (For values outside the limit values)

INFORMATION



PDO.TORQUE limits the amperage available for the movement.

If the value of PDO.TORQUE is too low, the product may not be able to reach the speed setpoint specified by PDO.VELOCITY.

Typically, the rated speed (PDO.VELOCITY = 100) is not reached if PDO.TORQUE < 30.

11.3.7.2 PDO.VELOCITY

PDO.VELOCITY defines the rotational speed as a percentage of the maximum possible rotational speed.

Data type	UINT8
Value range	0 to 100

11.3.7.3 PDI.STATUS

PDI.STATUS provides information about the state of the module and allows you to track the execution of movement commands.

Bit	Name	Description
0	control_active	If <i>control_active</i> = 1, then the initialization sequence is completed. The switch to 1 corresponds to exiting the <i>START</i> status. ► For more information, refer to section "11.2 Finite state machine".
1	enabled	If <i>enabled</i> = 1, then the module is in the <i>ENABLED</i> state.
2	error	If <i>error</i> = 1, then the module is in the <i>FAULT</i> or <i>LOCKED</i> state.
3	warning	If <i>warning</i> = 1, then a warning is active.
4	reserve_b04	0
5	busy	If <i>busy</i> = 1, then the following possibilities exist: • A movement command is being processed if the module is in the <i>ENABLED</i> state. • The module is in the <i>LOCKED</i> state if <i>PDI.STATUS.error</i> = 1.
6	done	► Please refer to sections "11.5 Quickstart" and "11.6 Functions and move commands".
7	in_target	► Please refer to sections "11.5 Quickstart" and "11.6 Functions and move commands".
8	new_order	This changes the state when a new movement command has been received and confirmed by the module.
9	order_rejected	If <i>order_rejected</i> = 1, then the last move command was rejected by the module: This means that an invalid command or parameter was entered.
10	in_motion	If <i>in_motion</i> = 1, then movement is taking place.
11	reserve_b11	0
12	reserve_b12	0
13	reserve_b13	0
14	reserve_b14	0
15	reserve_b15	0

11.3.7.4 PDI.ACTUAL_POSITION

PDI.ACTUAL_POSITION reports the current position.

Data type	INT32
Value range	• -1179648000 to 1179647999
Conversion	Position = PDI.POSITION / 100
Unit	0.01°

INFORMATION



If the value range is exceeded, an overflow or underflow occurs. These cases must be managed by the control system.

Example:

PDI.POSITION	Actual position
0x0000 ACF8	Multiturn: • $0x0000\ ACF8 * 0.01 = 45000 * 0.01 = 450^\circ$
	Singleturn: • $(0x0000\ AFC8 * 0.01) \bmod 360 = (-45000 * 0.01) \bmod 360 = 90^\circ$
0xFFFE E0E4	Multiturn: • $0xFFFE\ E0E4 * 0.01 = -73500 * 0.01 = -735^\circ$
	Singleturn: • $(0xFFFE\ E0E4 * 0.01) \bmod 360 = (-73500 * 0.01) \bmod 360 = 345^\circ$

11.3.7.5 PDI.ERROR

PDI.ERROR indicates the saved error with the highest priority.

INFORMATION



0 indicates that there are no longer any saved or active errors.

► For more information, refer to section "13.1 Error".

11.3.7.6 PDI.WARNING

PDI.WARNING indicates the active warning with the highest priority.

INFORMATION



► For more information, refer to section "13.2 Warnings".

11.4 Status of the LEDs

LED display	Icon	Status	Meaning
Logic supply		LED lights up continuously green.	Operating voltage OK
		LED does not light up.	Operating voltage not OK
Status		LED lights up continuously yellow.	Module is in the <i>START</i> state.
		LED lights up continuously red.	Module is in the <i>FAULT</i> state. ⇒ The module cannot move.
		LED flashes red.	Module is in the <i>LOCKED</i> state. ⇒ The module cannot move.
		LED lights up continuously green.	Module is in the <i>ENABLED</i> state.
		LED flashes green.	Module is in the <i>ENABLED</i> state. ⇒ A warning is active.
		LED lights up continuously blue.	Module is in the <i>READY</i> state.
		LED flashes blue.	Module is in the <i>READY</i> state. ⇒ A warning is active.
		LED does not light up.	Operating voltage not OK
Power supply		LED lights up continuously green.	Operating voltage OK
		LED flashes orange.	Warning: Operating voltage is too low
		LED lights up continuously orange.	Error: Operating voltage is too low ⇒ The module cannot move.
		LED flashes red.	Warning: Operating voltage is too high
		LED lights up continuously red.	Error: Operating voltage is too high ⇒ The module cannot move.
		LED does not light up.	Operating voltage not OK

11.5 Quickstart

In this example, the rotary module rotates counterclockwise three times and then returns to its starting position.

```
// Rotary unit is in the ENABLED state.

CASE iStep OF
  90:                                     // Initialize ISDUs
      RU_WORKPIECE_TOLERANCE = 3000;    // 30°
      RU_CLOSE_STOP = 10000;           // 100°
      iStep := 100;

  100:                                  // Perform a task before the rotary unit starts moving
      iStep := 100;

  110:                                  // Perform 3 rotations
      PDO.TORQUE := 50;                 // Set rotary unit torque value
      PDO.VELOCITY := 50;               // Set rotary unit velocity value
      PDO.POSITION := -108000;          // -1080°
      order_flag := PDI.STATUS.new_order; // Order_flag is used to detect that the device has accepted a new order
      PDO.COMMAND := 16#0204;           // Relative positioning
      iStep := 120;

  120:
      IF order_flag <> PDI.STATUS.new_order THEN // The device has accepted the order
          IF NOT PDI.STATUS.order_rejected THEN
              iStep := 130;
          ELSE
              iStep := 500; // Something has gone wrong
          END_IF
      END_IF

  130:
      IF NOT PDI.STATUS.busy THEN // Movement done
          IF PDI.STATUS.done AND PDI.STATUS.in_target THEN // The rotary unit is in position
              iStep := 140;
          ELSIF NOT PDI.STATUS.done THEN // Something has gone wrong
              iStep := 500;
          END_IF
      END_IF

  140:                                  // Do something while the rotary is in position
      iStep := 150;

  150:                                  // Place the rotary unit in its initial position
      PDO.POSITION := 108000;           // 1080°
      order_flag := PDI.STATUS.new_order; // Order_flag is used to detect that the device has accepted a new order
      PDO.COMMAND := 16#0204;           // Relative positioning
      iStep := 160;

  160:
      IF order_flag <> PDI.STATUS.new_order THEN // The device has accepted the order
          IF NOT PDI.STATUS.order_rejected THEN
              iStep := 170;
          ELSE
              iStep := 500; // Something has gone wrong
          END_IF
      END_IF

  170:
      IF NOT PDI.STATUS.busy THEN // Movement done
          IF PDI.STATUS.done AND PDI.STATUS.in_target THEN // The rotary unit is in initial position
              iStep := 400;
          ELSE
              iStep := 500; // Something has gone wrong
          END_IF
      END_IF

  400:                                  // Done (OK)

  500:                                  // Something wrong has happened

END_CASE
```

11.6 Functions and move commands

11.6.1 Enable/start-up

The following example shows how to perform a cold start and how to set the module to the *ENABLED* state.

CASE iStep OF

```

0:
  PDO.COMMAND := 0;           // Defined state for command
  PDO.CONTROL.enable := 0;     // Defined state for control
  iStep := 10;

10:
  iStep := 20;                // Logic and power supply can be switched on

20:
  // Wait for READY state
  IF PDI.STATUS.control_active THEN
    IF PDI.STATUS.error THEN
      iStep := 1000;           // Error handling
    ELSE
      iStep := 30;
    END_IF
  END_IF

30:
  // The product can be switched on
  // Command to switch on
  PDO.CONTROL.enable:=1;
  IF PDI.STATUS.enabled THEN
    iStep := 40;
  END_IF

40:
  // The device is in the ENABLED state

```

END_CASE;

11.6.2 Easy Jog (0x0001, 0x0002)

This command sets the rotary module to jog mode and slowly moves it clockwise or counterclockwise.

INFORMATION



With Easy Jog, the module always uses a default value of 10% of the speed.
The value entered in *PDO.VELOCITY* has no influence on the speed.

Program example:

// In this example the two commands *Easy Jog Minimum Position* and *Easy Jog Maximum Position* are used to approach two different positions.
// The device is in the ENABLED state. The current rotary unit position is 0°.

CASE iStep OF

```

100:
    order_flag := PDI.STATUS.new_order;           // Order_flag is used to detect that the device has accepted a new order
    PDO.COMMAND := 16#0002;                       // Easy Jog Counterclockwise
    iStep := 110;

110:
    IF order_flag <> PDI.STATUS.new_order THEN    // The device has accepted the order
        IF NOT PDI.STATUS.order_rejected THEN
            iStep := 120;
        ELSE
            iStep := 500;
        END_IF
    END_IF

120:
    IF PDI.ACTUAL_POSITION < -5000 THEN
        PDO.COMMAND := 16#0000;                 // No Operation
        iStep := 130;
    ELSIF PDI.STATUS.Error THEN
        iStep := 500;
    END_IF

130:
    order_flag := PDI.STATUS.new_order;           // Order_flag is used to detect that the device has accepted a new order
    PDO.COMMAND := 16#0001;                       // Easy Jog Clockwise
    iStep := 140;

140:
    IF order_flag <> PDI.STATUS.new_order THEN    // The device has accepted the order
        IF NOT PDI.STATUS.order_rejected THEN
            iStep := 150;
        ELSE
            iStep := 500;
        END_IF
    END_IF

150:
    IF PDI.ACTUAL_POSITION > 40000 THEN
        PDO.COMMAND := 16#0000;                 // No Operation
        iStep := 400;
    ELSIF PDI.STATUS.Error THEN
        iStep := 500;
    END_IF

400:
    // Done (OK)

500:
    // Something wrong has happened

END_CASE;
```

11.6.3 Disable

The following example shows how to set the module from the *ENABLED* state to the *READY* state.

INFORMATION



A movement command in the *READY* status generates an error.

► Always set *PDO.COMMAND* to 0 before you set *PDO.CONTROL.enable* to 0.

Program example:

```
// The device is in the ENABLED state.

CASE iStep OF

  0:
    PDO.COMMAND := 0;           // Stop all commands
    PDO.CONTROL.enable := 0;    // Disable device
    iStep := 10;

  10:
    IF NOT PDI.STATUS.enabled THEN
      iStep := 20;
    END_IF

  20:
    // The device is in the READY state

END_CASE;
```

11.6.4 Troubleshooting in the LOCKED state

The *LOCKED* status prevents the deletion of errors and unintentional movements.

Before errors can be acknowledged, this state must be exited by setting *PDO.COMMAND* and *PDO.CONTROL* to 0.

Program example:

```
// The device is in the LOCKED state.

CASE iStep OF

  1000:
    PDO.COMMAND := 0;           // PDO.COMMAND and PDO.CONTROL must be set to 0 to avoid an unexpected reset or restart
    PDO.CONTROL.enable := 0;    // Stop all commands
                                // Disable device

    IF NOT PDI.STATUS.busy THEN
      iStep := 1100;
    END_IF

  1100:
    // The device is in the FAULT state. Errors can be acknowledged and the terminal element can be opened.

END_CASE;
```

11.6.5 Troubleshooting in the FAULT state

Errors can be acknowledged in the *FAULT* status via the *ACK* command (*PDO.COMMAND.error_reset*).

In the *FAULT* state, the module attempts to acknowledge all errors as long as the *ACK* command is active (*PDO.CONTROL.error_reset* = 1).

If successful, the module switches to the *READY* state and the *ACK* command can be reset (*PDO.CONTROL.error_reset* = 0).

INFORMATION



- Leave the *ENABLE* command at 0 until the module is in the *READY* state so that no error is generated.

Program example:

// The device is in the *FAULT* state and *PDI.ERROR* is not equal to 0.

CASE iStep OF

```

1100:
    PDO.COMMAND := 0;           // Stop all commands
    PDO.CONTROL.enable := 0;     // Disable device
    PDO.CONTROL.reset_error := 1; // Trigger to reset error
    iStep := 1110;

1110:
    // Wait until all errors have been acknowledged
    IF NOT PDI.STATUS.error THEN // The device is in READY state
        PDO.CONTROL.reset_error := 0; // Release reset signal
        iStep := 400;
    END_IF

400:
    // Done (OK)
    
```

END_CASE;

11.6.6 Positioning

These commands are used to position or open a cover.

TARGET_POSITION defines the position that the rotary module must reach at the end of the movement.

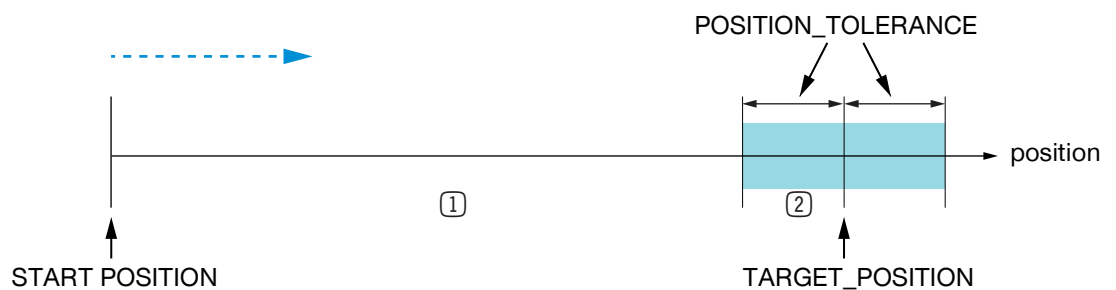
If *TARGET_POSITION* is reached taking *POSITION_TOLERANCE* into account, then the process was successful.

The direction of movement is defined by *START_POSITION* and *TARGET_POSITION*.

As long as the command is active, the rotary module attempts to reach or maintain *TARGET_POSITION*.

- For more information, refer to section "12 Acyclic data".

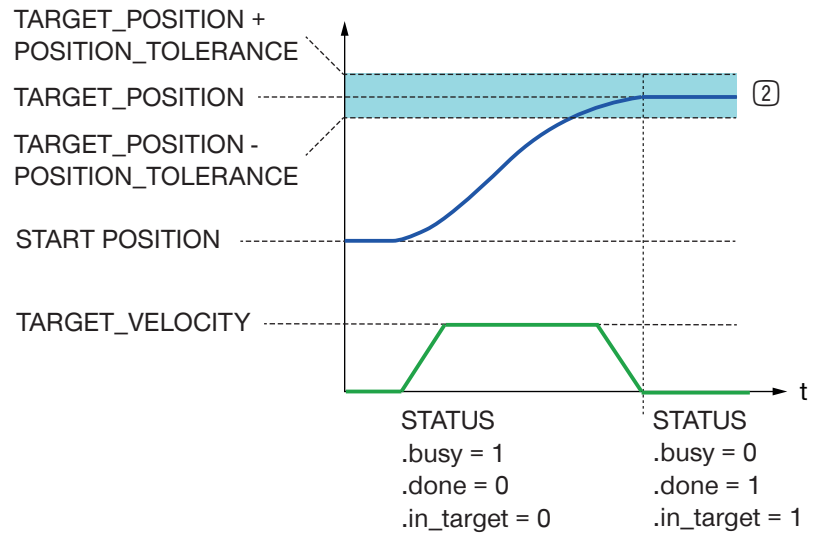
Clockwise



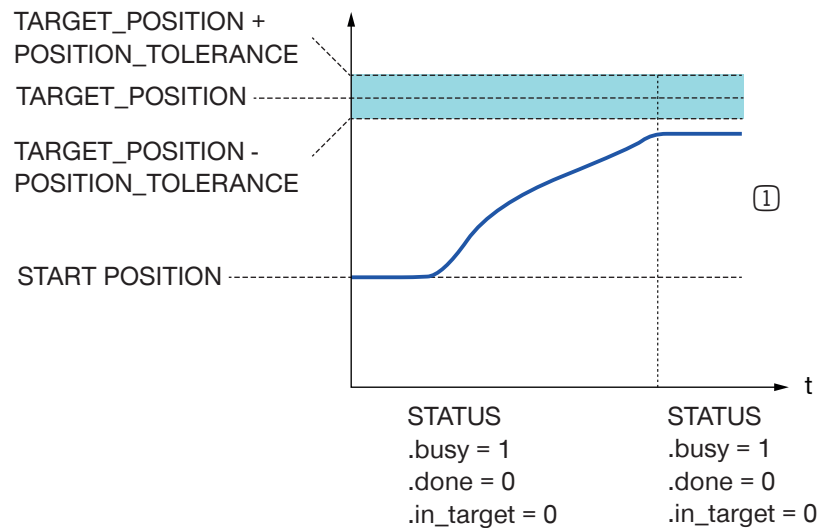
Counterclockwise



- If the movement stops in area ②, the process was successful.
- If the movement initially stops in area ② and was therefore successful, but then leaves area ②, then $PDI.STATUS.in_target = 0$.



- If the movement stops in area ①, then $PDI.STATUS.busy = 1$.



The following parameters define the movement:

- **TARGET_POSITION:** Is defined with $PDO.POSITION$.
 - See section "11.6.6.1 Absolute Positioning Multiturn (0x0104)".
 - See section "11.6.6.2 Relative Positioning Multiturn (0x0204)".
 - See section "11.6.6.3 Absolute Positioning Singleturn (0x0304, 0x0384, 0x0604)".
- **POSITION_TOLERANCE:** Defines the tolerance range within which the target position is considered to have been reached.
 - See $POSITION_TOLERANCE$ in section "12 Acyclic data".
- **START_POSITION:** Corresponds to the current position of the rotary module ($PDI.ACTUAL_POSITION$).
- **TARGET_VELOCITY:** Is defined with $PDO.VELOCITY$. If $PDO.VELOCITY = 0$, then a default value is used and a warning is set.
- **ACCELERATION:** See $ACCELERATION$ in section "12 Acyclic data".

11.6.6.1 Absolute Positioning Multiturn (0x0104)

$TARGET_POSITION = PDO.POSITION$

The direction of movement depends on the current position:

- $PDI.ACTUAL_POSITION < PDO.POSITION$ = Clockwise
- $PDI.ACTUAL_POSITION > PDO.POSITION$ = Counterclockwise

11.6.6.2 Relative Positioning Multiturn (0x0204)

$TARGET_POSITION = PDI.ACTUAL_POSITION + PDO.POSITION$

The direction of movement depends on the sign of PDO.POSITION.

- $PDO.POSITION > 0$ = Clockwise
- $PDO.POSITION < 0$ = Counterclockwise

11.6.6.3 Absolute Positioning Singleturn (0x0304, 0x0384, 0x0604)

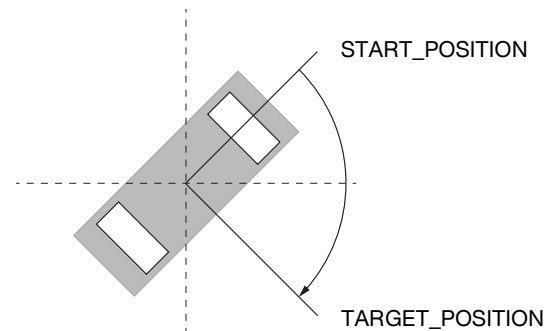
In *Singleturn* mode, the position is specified within a single revolution. Therefore, for movement commands in *Singleturn* mode, only target positions in the range from 0° to $< 360^\circ$ are permitted.

Otherwise, the command is rejected and a warning is set.

$TARGET_POSITION = PDO.POSITION$

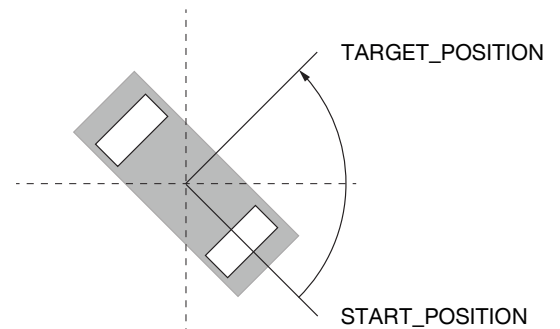
• Clockwise

The rotary module reaches the target position by moving clockwise.



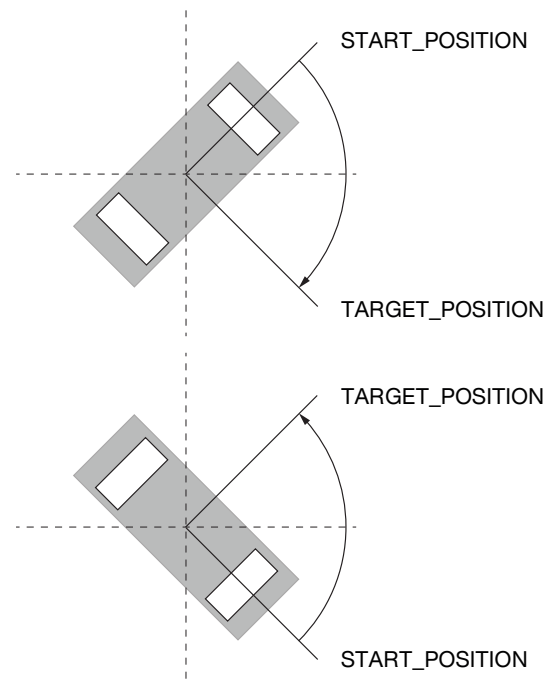
• Counterclockwise

The rotary module reaches the target position by moving counterclockwise.



• Shortest

The direction of rotation depends on $TARGET_POSITION$ and $START_POSITION$. The module takes the shortest route.



11.6.7 Close at Position

These commands are used to close a cover with a defined torque.

TARGET_POSITION defines the position that the rotary module must reach at the end of the movement.

WORKPIECE_TOLERANCE can be used to define a valid range around this position.

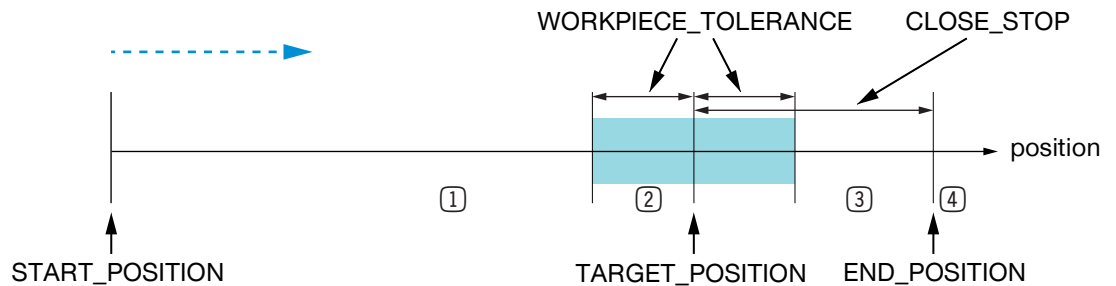
CLOSE_STOP defines the position at which the movement stops if no workpiece is present. If this position is reached, then the process was not successful.

The direction of movement is defined by START_POSITION and END_POSITION.

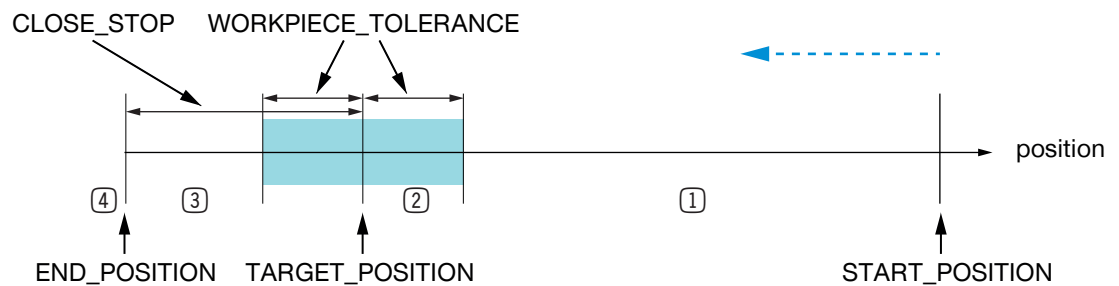
As long as the command is active, the module attempts to reach or maintain END_POSITION.

► For more information, refer to section "12 Acyclic data".

Clockwise



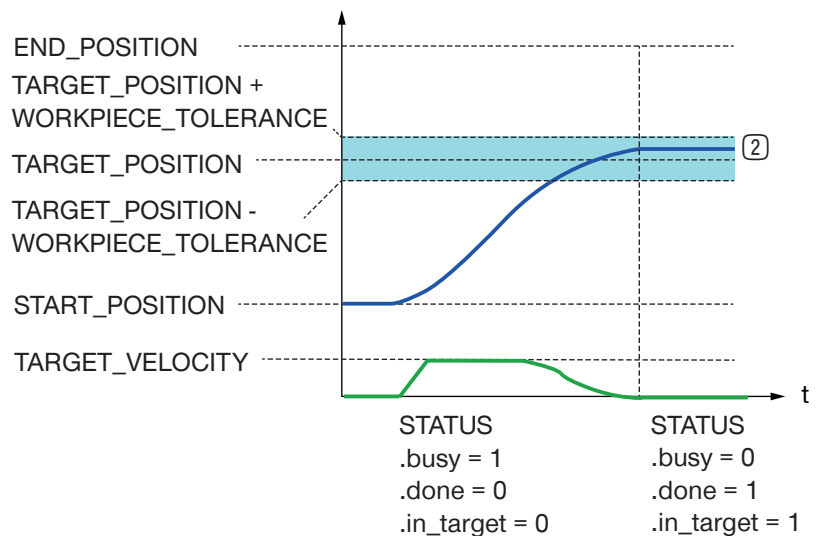
Counterclockwise



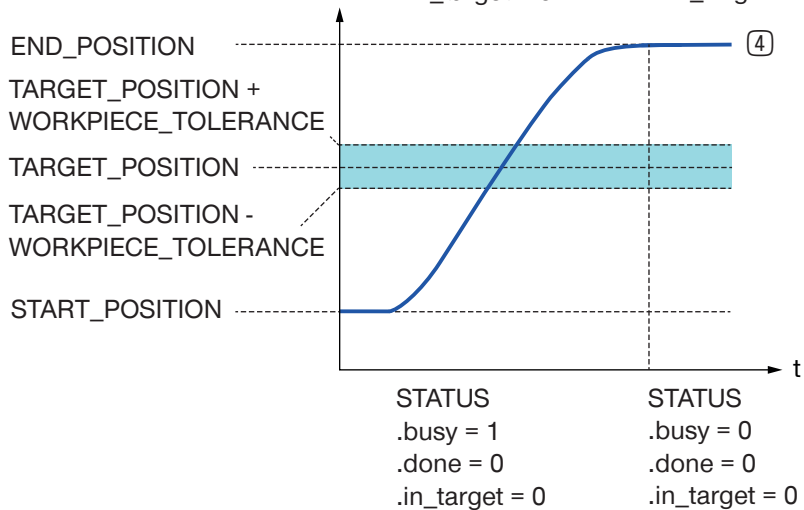
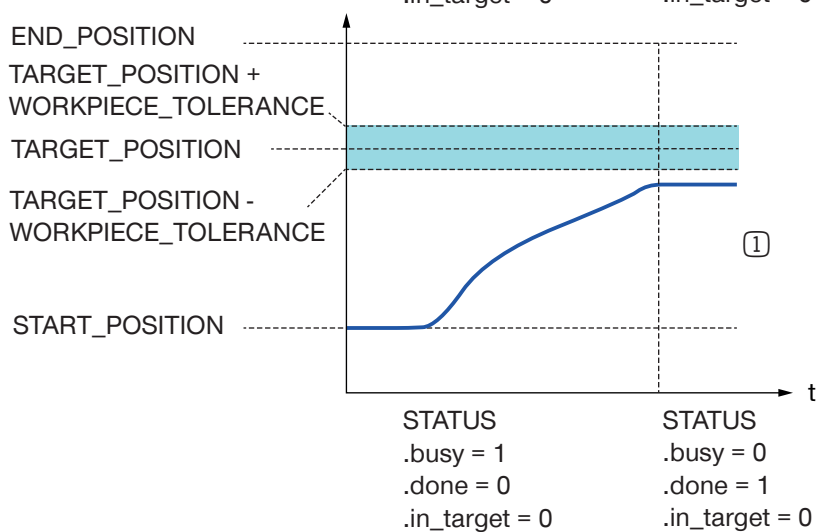
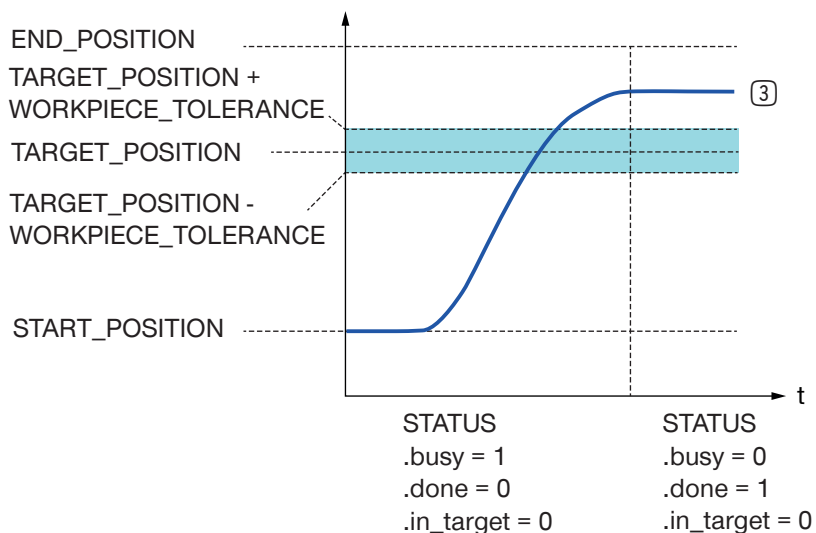
If the command is accepted, PDI.STATUS.busy is set to 1.

As soon as the movement stops, PDI.STATUS.busy is set to 0 and the module sets *in_target* and *done* according to the result of the operation.

- If the movement stops in area ②, the process was successful.
- If the movement initially stops in area ② and was therefore successful, but then leaves area ②, PDI.STATUS.in_target becomes 0.



- If the movement stops in area ①, ③ or ④, the process was not successful.



The following parameters define the movement:

- TARGET_TORQUE: Is defined with PDO.TORQUE.
- TARGET_POSITION: Is defined with PDO.POSITION.
 - See section "11.6.7.1 Close at Position: Absolute Positioning (0x0208)".
 - See section "11.6.7.2 Close at Position: Relative Positioning (0x0808)".
- CLOSE_STOP: Is required to define END_POSITION.
 - See CLOSE_STOP in section "12 Acyclic data".
 - Clockwise movement: $END_POSITION = TARGET_POSITION + CLOSE_STOP$
 - Counterclockwise movement: $END_POSITION = TARGET_POSITION - CLOSE_STOP$
- WORKPIECE_TOLERANCE: Is defined with TARGET_POSITION and corresponds to the range in which the end of the movement is expected.
 - See WORKPIECE_TOLERANCE in section "12 Acyclic data".
- START_POSITION: Corresponds to the current position of the rotary module (PDI.ACTUAL_POSITION).
- TARGET_VELOCITY: Is defined with PDO.VELOCITY. If PDO.VELOCITY = 0, then a default value is used and a warning is set.
- ACCELERATION: See ACCELERATION in section "12 Acyclic data".

11.6.7.1 Close at Position: Absolute Positioning (0x0208)

$TARGET_POSITION = PDO.POSITION$

The direction of movement depends on the current position:

- $PDI.ACTUAL_POSITION < PDO.POSITION$ = Clockwise
- $PDI.ACTUAL_POSITION > PDO.POSITION$ = Counterclockwise

11.6.7.2 Close at Position: Relative Positioning (0x0808)

$TARGET_POSITION = PDI.ACTUAL_POSITION + PDO.POSITION$

The direction of movement depends on the sign of PDO.POSITION.

- $PDO.POSITION > 0$ = Clockwise
- $PDO.POSITION < 0$ = Counterclockwise

11.6.8 Close up to position

These commands are used to close a cover with a defined torque if the exact number of turns required to close it is not known.

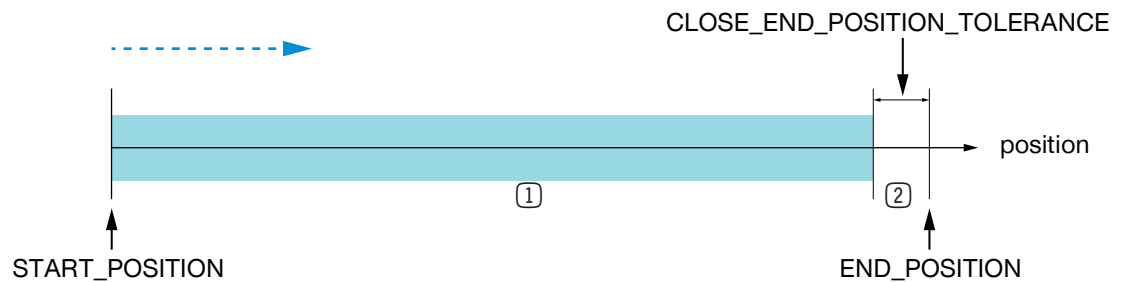
END_POSITION defines the position at which the movement stops if the cover has not been fully closed. If this position is reached, then the process was not successful.

The direction of movement is defined by START_POSITION and END_POSITION.

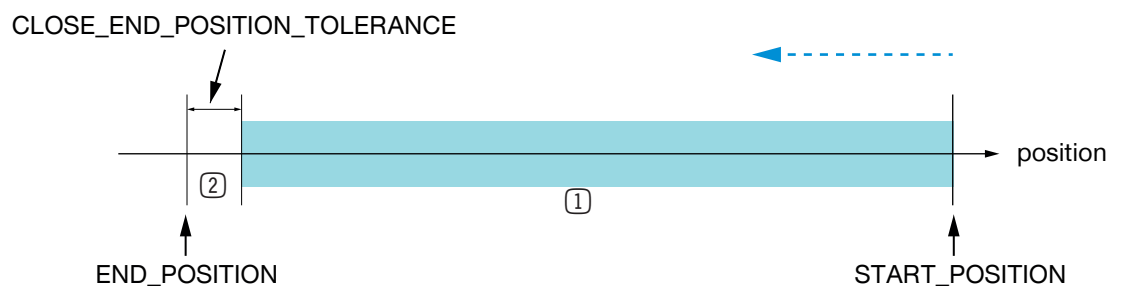
As long as the command is active, the module attempts to reach or maintain END_POSITION.

► For more information, refer to section "12 Acyclic data".

Clockwise



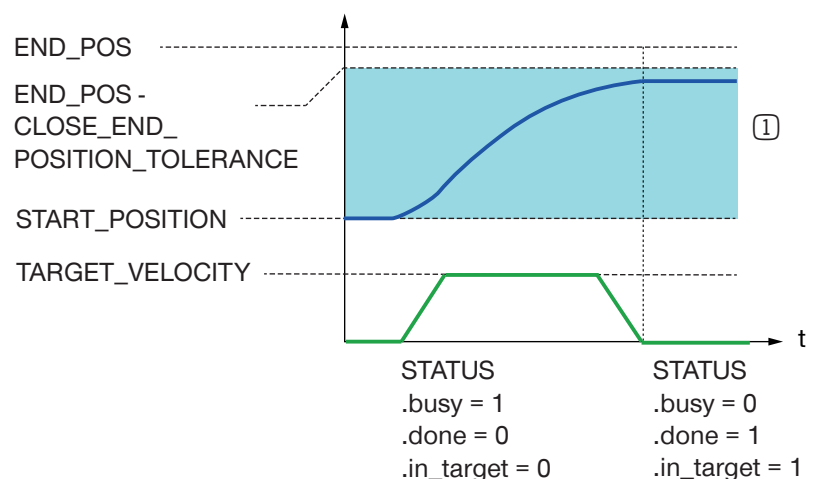
Counterclockwise



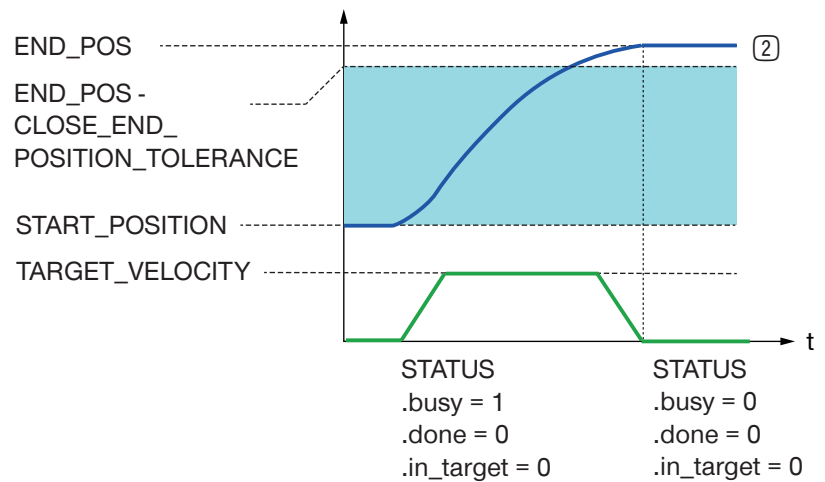
If the command is accepted, PDI.STATUS.busy is set to 1.

As soon as the movement stops, PDI.STATUS.busy is set to 0 and the module sets *in_target* and *done* according to the result of the operation.

- If the movement stops in area ①, the process was successful.



- If the movement stops in area ②, the process was not successful.



The following parameters define the movement:

- CLOSE_END_POSITION_TOLERANCE: See CLOSE_END_POSITION_TOLERANCE in section "12 Acyclic data".
- TARGET_TORQUE: Is defined with PDO.TORQUE.
- TARGET_POSITION: Is defined with PDO.POSITION.
 - See section "11.6.8.1 Close up to position: Absolute Positioning (0x0108)".
 - See section "11.6.8.2 Close up to position: Relative Positioning (0x0708)".
- START_POSITION: Corresponds to the current position of the rotary module (PDI.ACTUAL_POSITION).
- TARGET_VELOCITY: Is defined with PDO.VELOCITY. If PDO.VELOCITY = 0, then a default value is used and a warning is set.
- ACCELERATION: See ACCELERATION in section "12 Acyclic data".

11.6.8.1 Close up to position: Absolute Positioning (0x0108)

END_POSITION = PDO.POSITION

The direction of movement depends on the current position:

- PDI.ACTUAL_POSITION < PDO.POSITION = Clockwise
- PDI.ACTUAL_POSITION > PDO.POSITION = Counterclockwise

11.6.8.2 Close up to position: Relative Positioning (0x0708)

END_POSITION = PDI.ACTUAL_POSITION + PDO.POSITION

The direction of movement depends on the sign of PDO.POSITION.

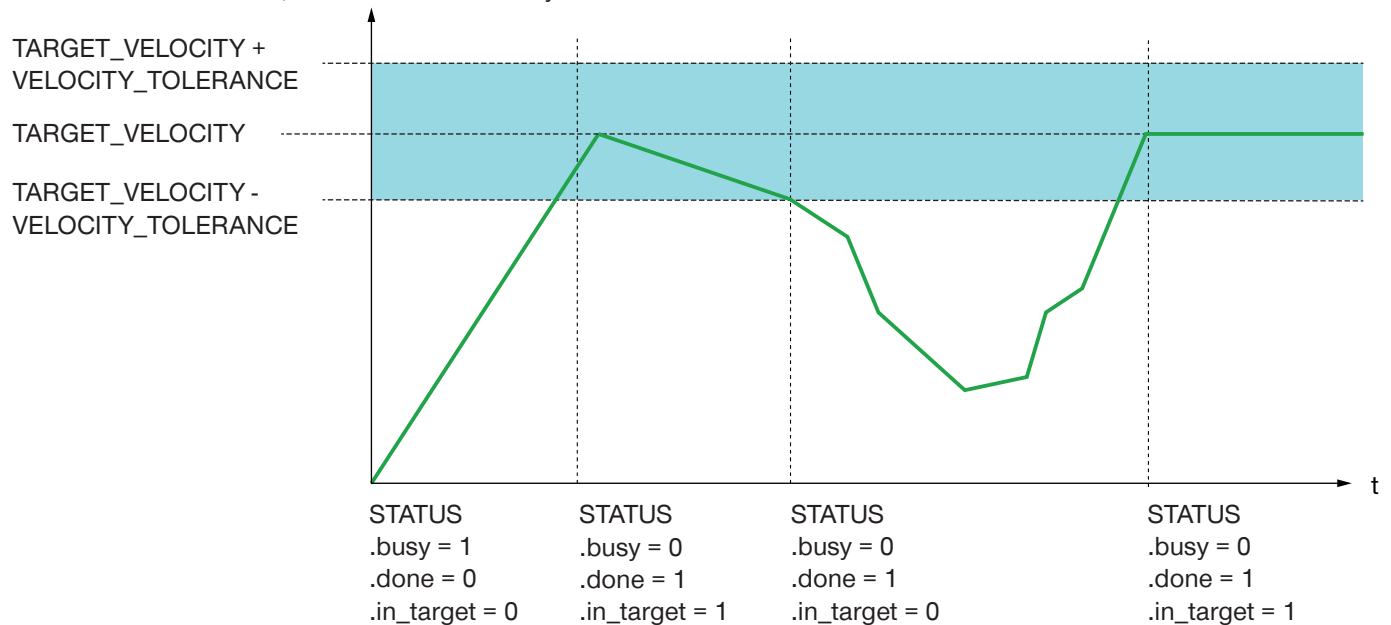
- PDO.POSITION > 0 = Clockwise
- PDO.POSITION < 0 = Counterclockwise

11.6.9 Velocity mode (0x0504, 0x0584)

These commands are used to rotate the rotary module at a specified speed.

If the movement has been stopped or slowed down, the rotary module attempts to reach TARGET_VELOCITY again.

► For more information, refer to section "12 Acyclic data".



The following parameters define the movement:

- **TARGET_VELOCITY:** Is defined with PDO.VELOCITY. If PDO.VELOCITY = 0, then the command is rejected and a warning is set.
- **VELOCITY_TOLERANCE:** Is defined with TARGET_VELOCITY and corresponds to the range in which the end of the acceleration is expected.
 - See VELOCITY_TOLERANCE in section "12 Acyclic data".
- **ACCELERATION:** See ACCELERATION in section "12 Acyclic data".

11.6.10 Stop

This command is used to stop the current move command and the current movement.

If the rotary module accepts the command, the speed is reduced to 0 and the rotary module retains the position reached.

If the command is accepted, PDI.STATUS.busy is set to 1. As soon as the movement stops, PDI.STATUS.busy is set to 0. As soon as PDI.STATUS.busy = 0, other move commands can be accepted.

INFORMATION



The *Stop* command cannot be interrupted by another command.

If another command is sent during the execution of *Stop*, it is rejected and the warning 0x0B03 is reported.

Program example:

// The device is in the ENABLED state.

// The device is currently executing a movement order (such as velocity or absolute positioning, for example).

CASE iStep OF

```
80:
  order_flag := PDI..STATUS.new_order;           // Order_flag is used to detect that the device has accepted a new order
  PDO.COMMAND := 16#8000;                         // Stop
  iStep := 90;
```

```
90:
  IF order_flag <> PDI.STATUS.new_order THEN      // The device has accepted the order
    IF NOT PDI.STATUS.order_rejected THEN
      iStep := 100;
    ELSE
      iStep := 500;
    END_IF
  END_IF
```

```
100:
  IF NOT PDI.STATUS.busy THEN                     // Unit stopped
    iStep := 400;
  END_IF
```

```
400:                                             // Done (OK)
```

```
500:                                             // Something wrong has happened
```

END_CASE;

11.6.11 Reset Multiturn Position (0x0F10)

This command is used to convert the Multiturn value in PDO.ACTUAL_POSITION to the corresponding Singleturn value. The mechanical position remains unchanged.

Example:

PDO.ACTUAL_POSITION before Reset Multiturn position	PDO.ACTUAL_POSITION after Reset Multiturn position
75000 (750°)	3000 (30°)
5000 (50°)	5000 (50°)
-364800 (-3648°)	31200 (312°)

Prerequisite for execution:

- Rotary module in ENABLED state
- No other command in execution (PDO.STATUS.busy = 0)
- Rotary module at a standstill (PDO.STATUS.in_motion = 0)

Changes in process data:

If the command is accepted, the process data changes as follows:

PDO.STATUS.new_order	changes
PDO.STATUS.order_rejected	0
PDO.done	1
PDO.in_target	0
PDO.ACTUAL_POSITION	Singleturn value

If the command is not accepted, the process data changes as follows:

PDO.STATUS.new_order	changes
PDO.STATUS.order_rejected	1

Program example:

```
// The device is in the ENABLED state.
// In this example, the actual position will be set back to its Singleturn value.

CASE iStep OF

  100:
    IF NOT PDI.STATUS.busy AND NOT PDI.STATUS.in_motion THEN
      order_flag := PDI.STATUS.new_order;
      PDO.COMMAND := 16#0F10;
      iStep := 110;
    END_IF
    // The device is not in motion and no orders are being executed
    // Order_flag is used to detect that the device has accepted a
    // new order
    // Reset Multiturn Position

  110:
    IF order_flag <> PDI.STATUS.new_order THEN
      IF NOT PDI.STATUS.order_rejected THEN
        iStep := 120;
      ELSE
        iStep := 500;
      END_IF
    END_IF
    // The device has accepted the order

  120:
    IF NOT PDI.STATUS.busy AND PDI.STATUS.done THEN
      iStep := 400;
    END_IF
    // The position is set back to its singleturn value

  400:
    // Done (OK)

  500:
    // Something wrong has happened

END_CASE;
```

12 Acyclic data

12.1 General

INFORMATION



A restart is required to change BAUDRATE and STATION_ADDRESS.

► Restart the product by disconnecting it from the power supply and reconnecting it.

Modbus Register address	Amount Registers	Data type	Access	NVM	Value Range	Default Value	Unit
PRODUCT_NAME							
Product name							
0x0100 - 0x011F (256 - 287)	32	STRING	RO	-	-	-	-
BAUDRATE							
Modbus RTU baud rate							
0: 9.6 kbaud 1: 19.2 kbaud 2: 38.4 kbaud 3: 57.6 kbaud 4: 115.2 kbaud 5: 230.4 kbaud 6: 460.8 kbaud							
0x3B01 (15105)	1	UINT8	RW	Yes	0 – 6	4	-
STATION_ADDRESS							
Station address							
0x3B00 (15104)	1	UINT8	RW	Yes	1 – 247	1	-
FIRMWARE_VERSION							
Software version							
0x0180 - 0x019F (384 - 415)	32	STRING	RO	-	-	-	-

12.2 Rotary module

Modbus Register address	Amount Registers	Data type	Access	NVM	Value Range	Default Value	Unit
RU_ACCELERATION							
Defines the acceleration.							
0x151D - 0x151E (5405 - 5406)	2	UINT32	RW	Yes	100 – 12000	6000	°/s ²
RU_CLOSE_STOP							
Defines the position at which the rotary module stops.							
0x0150C - 0x0150D (5388 - 5389)	2	UINT32	RW	Yes	RU_WORKPIECE_TOLERANCE ≤ RU_CLOSE_STOP	10000 [100.0°]	0.01°
RU_LIMIT_VEL_MAX							
Defines the absolute maximum speed.							
0x2086 - 0x2087 (8326 - 8327)	2	INT32	RO	-	-	-	°/s
RU_POSITION_TOLERANCE							
Defines a tolerance range for the position.							
0x1504 - 0x1505 (5380 - 5381)	2	UINT32	RW	Yes	20 – 36000 [0.2°–360.0°]	100 [1.0°]	0.01°
RU_VELOCITY_TOLERANCE							
Defines a tolerance range for the speed.							
0x150E - 0x150F (5390 - 5391)	2	UINT32	RW	Yes	1 – 1500	100 [1 °/s]	°/s
RU_WORKPIECE_TOLERANCE							
Defines a tolerance range for the workpiece position for the Close command.							
0x1506 - 0x1507 (5382 - 5383)	2	UINT32	RW	Yes	20 – 36000 [0.2°–360.0°]	5000 [50.0°]	0.01°
RU_CLOSE_END_POSITION_TOLERANCE							
Defines a tolerance range for the end position when using the Close up command.							
0x1525 - 0x1526 (5413 - 5414)	2	UINT32	RW	Yes	20 – 36000 [0.2°–360.0°]	100 [1.0°]	0.01°

13 Error diagnosis

13.1 Error

Errors lead to a standstill of the motor and the power level. When an error occurs, the respective module is set to the *FAULT* or *LOCKED* state and the module can no longer move.

Errors remain visible until they are acknowledged, even if the cause has been eliminated.

Error code	Error	Possible cause	Measure
0x0000 (0)	No error	-	-
0x0101 (257)	Logic supply voltage low	Logic supply voltage is below the permissible range.	► Check the supply voltage.
0x0102 (258)	Logic supply voltage high	Logic supply voltage is above the permissible range.	► Check the supply voltage.
0x0103 (259)	Actuator supply voltage low	Actuator supply voltage is below the permissible range.	► Check the supply voltage.
0x0104 (260)	Actuator supply voltage high	Actuator supply voltage is above the permissible range.	► Check the supply voltage.
0x0302 (770)	Logic temperature high	The temperature of the circuit board exceeds the permissible range.	► Provide sufficient ventilation and cooling. ► Make sure that the module moves freely. ► Ensure the appropriate operating temperature. ► Reduce the on-time of the application.
0x0304 (772)	Motor temperature high	Motor temperature exceeds the permissible range.	► Provide sufficient ventilation and cooling. ► Make sure that the module moves freely. ► Ensure the appropriate operating temperature. ► Reduce the on-time of the application.
0x0305 (773)	Rectifier temperature high	Rectifier temperature exceeds the permissible range.	► Provide sufficient ventilation and cooling. ► Make sure that the module moves freely. ► Ensure the appropriate operating temperature. ► Reduce the on-time of the application.
0x0401 (1025)	Overcurrent	Overcurrent protection is active.	► Set <i>PDO.CONTROL</i> = 0. ► Set <i>PDO.COMMAND</i> = 0. ► Execute the ACK command (<i>PDO.CONTROL.error_reset</i>).
0x0801 (2049)	Finite state machine	Invalid value in <i>PDO.COMMAND</i> in the READY status	► Set <i>PDO.CONTROL.enable</i> = 0. ► Execute the ACK command (<i>PDO.CONTROL.error_reset</i>).
0x0803 (2051)	Finite state machine	<i>PDO.ENABLE</i> = 1 in FAULT status	► Set <i>PDO.CONTROL.enable</i> = 0. ► Execute the ACK command (<i>PDO.CONTROL.error_reset</i>).

Error code	Error	Possible cause	Measure
0x0805 (2053)	Finite state machine	Invalid value in PDO.CONTROL or PDO.COMMAND when exiting the START status	▶ Set <i>PDO.CONTROL</i> = 0. ▶ Set <i>PDO.COMMAND</i> = 0. ▶ Execute the ACK command (<i>PDO.CONTROL.error_reset</i>).

13.2 Warnings

Warnings serve as an indication of possible discrepancies. The module can be operated during a warning.

As soon as the cause of the warning has been eliminated, it is automatically canceled by the module and does not need to be acknowledged.

Error code	Warning	Possible cause	Measure
0x0000 (0)	No warning	-	-
0x0A01 (2561)	Position following error	Range of movement not unrestricted	▶ Make sure that there are no obstacles in the range of movement.
		Movement and dynamic requirements unsuitable for the workpiece	▶ Perform one or more of the following actions: ▶ Increase <i>PDO.TORQUE</i>. ▶ Please note section "11.3.7.1 PDO.TORQUE". ▶ Reduce <i>PDO.VELOCITY</i>. ▶ Please note section "11.3.7.2 PDO.VELOCITY". ▶ Reduce <i>ACCELERATION</i>. ▶ Please note section "12 Acyclic data".
		Off-center mass distribution of the workpiece (imbalance)	▶ Optimize the position of the workpiece while it is in the clamped state. ▶ If the position of the workpiece cannot be changed, perform one or more of the following actions: ▶ Increase <i>PDO.TORQUE</i>. ▶ Please note section "11.3.7.1 PDO.TORQUE". ▶ Reduce <i>PDO.VELOCITY</i>. ▶ Please note section "11.3.7.2 PDO.VELOCITY". ▶ Reduce <i>ACCELERATION</i>. ▶ Please note section "12 Acyclic data".
0x0B01 (2817)	Movement not possible/invalid command	Invalid value in PDO.COMMAND	▶ Issue a valid command.
0x0B02 (2818)	Movement not possible/invalid parameter	Invalid parameter	▶ Check the value range of the command.
0x0B03 (2819)	Movement not possible/stop process running	Stop process running	▶ Wait until the stop process is complete.
0x0B04 (2820)	Movement not possible	Jog command running	▶ First interrupt the jog command.
0x0B06 (2822)	Movement not possible	Movement running	▶ First interrupt the movement.

Error code	Warning	Possible cause	Measure
0x0C01 (3073)	Force parameter	- Force or torque target value is outside the valid range. - Default value is used.	► Use a valid value.
0x0C02 (3074)	Parameters Speed	- Speed target value is outside the valid range - Default value is used.	► Use a valid value.
0x0C03 (3075)	Position parameter	- Position target value is outside the valid range. - Default value is used.	► Use a valid value.

14 Maintenance

NOTICE



Material damage resulting from blowing out with compressed air

Blowing out the product with compressed air can cause malfunctions and pose a risk of accidents.

- ▶ Never purge the product with compressed air.

NOTICE



Material damage caused by unsuitable cleaning materials

Liquid and solvent-based cleaning agents can cause malfunctions.

- ▶ Do not clean the product with any cleaning agents that are liquid or contain solvents.

Maintenance-free operation of the product is guaranteed for up to 5 million cycles.

▶ Note that the product could become damaged under the following circumstances:

- Dirty environment
- Improper use and use that does not comply with the performance data
- Permissible temperature range not observed

▶ Have maintenance work that requires disassembly of the product performed by customer service only.

⇒ Dismantling and reassembling the product without authorization may result in complications, as special installation equipment is required in some cases. Zimmer GmbH accepts no liability for any resulting malfunctions or damage.

15 Decommissioning/disposal

INFORMATION



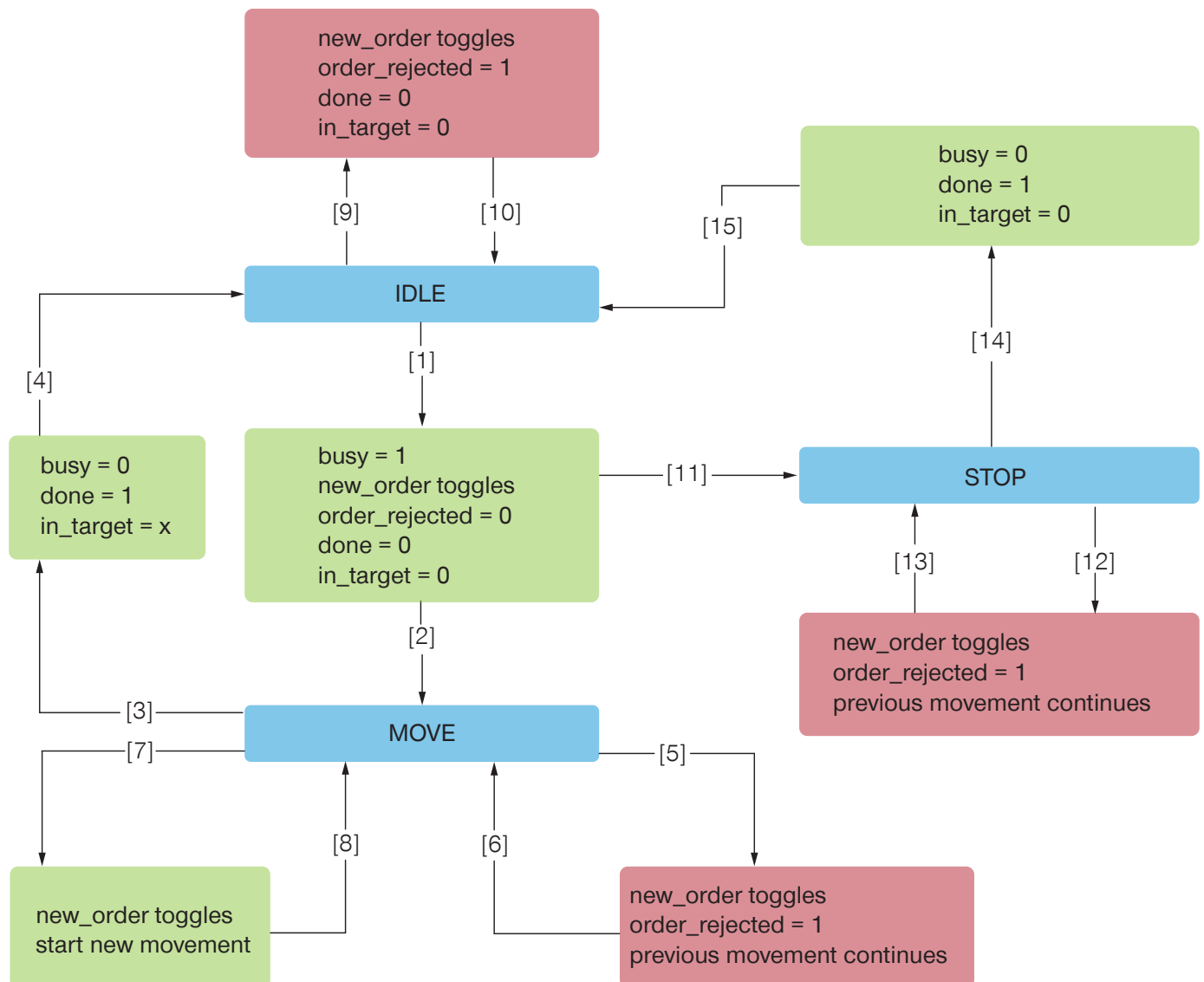
When the product reaches the end of its useful life, it can be disposed of.

- ▶ Disconnect the product completely from the power supply.
- ▶ Comply with the locally applicable environmental and disposal regulations.

16 Appendix

16.1 Movement finite state machine

This finite state machine demonstrates the changes in *PDI.STATUS* when a command is performed.



State	Description	Possibility to leave the state
IDLE	The module is ready to receive a new movement command.	A new command has been received; it can either be valid or invalid.
MOVE	The module performs the movement command.	- The movement was completed successfully or not as expected. - The <i>STOP</i> command was sent. - A new valid or invalid command was received during the movement.
STOP	The movement was stopped using the <i>STOP</i> command in <i>PDO.COMMAND</i> .	The module has stopped moving and is at a standstill.

Transition	Description
[1]	A new valid command has been set.
[2]	Automatic transition
[3]	The movement ended as expected.
[4]	Automatic transition
[5]	The movement did not end as expected.
[6]	Automatic transition
[7]	A new valid command has been set.
[8]	Automatic transition
[9]	A new invalid command has been set.
[10]	Automatic transition
[11]	<i>PDO.COMMAND</i> is set to 0x8000 (STOP).
[12]	A new invalid command has been set.
[13]	Automatic transition
[14]	The movement was completed.
[15]	Automatic transition

- To execute a movement command, the module must be in the *ENABLED* state and must not be moving (*PDI.STATUS.in_motion* = 0).
- To execute a movement, *PDO.COMMAND* must be set or a movement parameter, e.g., the target position, must be changed.
- If the module accepts a movement command, the value in *PDI.STATUS.new_order* is set to 1 and *PDI.STATUS.order_rejected* is set to 0.
- If a new movement command is not accepted, e.g., due to an incorrect command or an invalid parameter, the value in *PDI.STATUS.new_order* is set to 0 and *PDI.STATUS.order_rejected* is set to 1.

16.2 Software copyright notices

16.2.1 FreeModbus

This product includes software derived from the FreeModbus project.

FreeModbus Library: A portable Modbus implementation for Modbus ASCII/RTU.

Copyright (c) 2006-2018 Christian Walter. All rights reserved.

BSD license

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

Disclaimer

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.

IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

17 RoHS declaration

In terms of the EU Directive 2011/65/EU

Name and address of the manufacturer:

Zimmer Group GmbH

📍 Am Glockenloch 2
77866 Rheinau, Germany

☎ +49 7844 9138 0

✉ info@zimmer-group.com

💻 www.zimmer-group.com

We hereby declare that the incomplete machine described below

Product designation: Rotary module

Type designation: REH2000

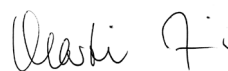
conforms to the requirements of the directive in its design and the version we put on the market.

Michael Hoch

Authorized representative for
compiling the relevant technical
documents

Rheinau, Germany, 2026-07-10

(Place and date of issue)



Martin Zimmer
(Legally binding signature)
Managing Partner

18 Declaration of Incorporation (Machinery Directive)

In terms of EU Machinery Directive 2006/42/EC (Annex II 1 B)

Name and address of the manufacturer:

Zimmer Group GmbH

📍 Am Glockenloch 2
77866 Rheinau, Germany

☎ +49 7844 9138 0

✉ info@zimmer-group.com

🌐 www.zimmer-group.com

We hereby declare that the incomplete machine described below

Product designation: Rotary module

Type designation: REH2000

conforms to the requirements of the Machinery Directive, 2006/42/EC, Article 2g, Annex VII, b – Annex II, b, in its design and the version we put on the market.

We hereby confirm that all the relevant basic health and safety requirements for the product have been observed and implemented.

A full list of applied standards can be obtained from the manufacturer.

We also declare that the specific technical documents were produced in accordance with Annex VII Part B of this Directive. We undertake to provide the market supervisory bodies with electronic versions of special documents for the incomplete machine through our documentation department, should they have reason to request them.

Commissioning of the incomplete machine is prohibited until it has been found that – where applicable – the machine in which the above-mentioned incomplete machine is to be installed complies with the Machinery Directive (2006/42/EC) and an EC Declaration of Conformity has been drawn up in accordance with Annex II 1 A.

Kurt Ross

Authorized representative for
compiling the relevant technical
documents

Rheinau, Germany, 2026-07-10

(Place and date of issue)



Martin Zimmer
(Legally binding signature)
Managing Partner

19 Declaration of Incorporation (Machinery Regulation)

In terms of EU Machinery Regulation 2023/1230, Annex V, Part B

Name and address of the manufacturer:

Zimmer Group GmbH

📍 Am Glockenloch 2
77866 Rheinau, Germany

☎ +49 7844 9138 0

✉ info@zimmer-group.com

🌐 www.zimmer-group.com

We hereby declare that the incomplete machine described below

Product designation: Rotary module

Type designation: REH2000

in its design and in the version we have placed on the market, it complies with the requirements of Regulation 2023/1230, Article 3.10, Annex III, Annex IV, Part B, Annex V, Part B.

A complete list of the standards applied and the essential health and safety requirements is available from the manufacturer.

The incomplete machine must not be commissioned until it has been determined that – where applicable – the machine into which the incomplete machine is to be incorporated complies with the provisions of EU Machinery Regulation 2023/1230 and the EC Declaration of Conformity has been issued in accordance with Annex V, Part A.

Christian Bosse

Authorized representative for
compiling the relevant technical
documents

Rheinau, Germany, 2027-01-20

(Place and date of issue)



Martin Zimmer
(Legally binding signature)
Managing Partner

20 Declaration of Conformity

In terms of the EC Directive 2014/30/EU on electromagnetic compatibility

Name and address of the manufacturer:

Zimmer Group GmbH

📍 Am Glockenloch 2
77866 Rheinau, Germany

☎ +49 7844 9138 0

✉ info@zimmer-group.com

🌐 www.zimmer-group.com

We hereby declare that the products described below

Product designation: Rotary module

Type designation: REH2000

conforms to the requirements of the Electromagnetic Compatibility Directive 2014/30/EU in their design and the version we put on the market.

The following harmonized standards have been used:

DIN EN ISO 12100	Safety of machinery – General principles for design – Risk assessment and risk reduction
DIN EN 61000-6-3	EMC Generic standard, Emission standard for residential, commercial and light-industrial environments
DIN EN 61000-6-2	EMC generic standard, Immunity for industrial environments
DIN EN 61000-6-4	EMC Generic standards, Emission standard for industrial environments

A full list of applied standards can be obtained from the manufacturer.

Kurt Ross

Authorized representative for
compiling the relevant technical
documents

Rheinau, Germany, 2026-07-10

(Place and date of issue)



Martin Zimmer
(Legally binding signature)
Managing Partner