Application Manual

Language	**English**
	Translation
Document No.	5.13013.04
Part No.	445132
Status	14-Jan-2018

be in motion be in motion

BAUMÜLLER

b maXX

POWERLINK
BM4-O-PLK-01

POWERLINK
Controlled Node

BM4400, BM4400 ES
BM4600, BM4600 ES
BM4700, BM4700 ES

Read the Operating Handbook before starting any work!

Copyright

This Application Manual may be copied by the owner in any quantity, but only for internal use. This Application Manual may not be copied or reproduced, in whole or in part, for any other purposes.
The use and disclosure of information contained in this Application Manual are not permitted. Designations and company marks contained in this Application Manual could be trademarks, the use of which by third parties for their own purposes could violate the rights of the rights holder.

Preliminary information

Warning Insofar as this document is identified as being preliminary information, the following applies:
This version is regarded as providing advance technical information to users of the described devices and their functions at an early enough time in order to adapt to any possible changes or expanded functionality.
This information must be regarded as being preliminary, as it has not yet passed through Baumüller's internal review process. In particular, this information is still subject to changes, thus no legal liability can be derived from this preliminary information. Baumüller assumes no liability for damages that might arise from this possibly faulty or incomplete version.
If you detect or suspect any content errors and/or major form errors in this preliminary information, we request that you notify the Baumüller support specialist responsible for you. Please provide us, via this employee, with your insights and comments so that we can take them into account and include them when transitioning from the preliminary information to the final information (as reviewed by Baumüller).
The conditions stipulated in the following section under "Obligatory" are invalid in case of preliminary information.

Obligatory

This Application Manual is a part of the equipment/machine. This Application Manual must be available to the operator at all times and must be in legible condition. If the equipment/machine is sold or moved another location, this Application Manual must be passed on by the owner together with the equipment/machine.
After any sale of the equipment/machine, this original and all copies must be handed over to the buyer. After disposal or any other end use, this original and all copies must be destroyed.

When the present Application Manual is handed over, corresponding sets of application manuals of a previous version are automatically invalidated.
Please note that the specifications/data/information **are current values according to the printing date**. These statements are **not legally binding** with regard to measurements, computation or calculations.
Baumüller Nürnberg GmbH reserves the right, in developing its products further, to change the technical specifications and handling of it products concerned without prior notice.

No liability can be accepted concerning the correctness of this Application Manual unless otherwise specified in the General Conditions of Sale and Delivery.

© **Baumüller Nürnberg GmbH**

Ostendstr. 80 - 90
90482 Nuremberg
Germany

Tel. +49 9 11 54 32 - 0
Fax: +49 9 11 54 32 - 1 30

Email : mail@baumueller.de
Internet: www.baumueller.de

Table of contents

Table of contents

1

INTRODUCTION

The application manual is an important part of the b maXX® 4400 (ES) device. Therefore this manual must be read completely, before starting any operation, last but not least on behalf of the own security. This manual describes how the company Baumüller Nürnberg GmbH implemented the option module **POWERLINK Controlled Node** for the series b maXX®4400.

The introduction contains general information regarding the option module **POWERLINK Controlled Node**.

1.1 General information

The **POWERLINK Controlled Node**option module connects the b maXX® 4400 (ES) via the POWERLINK bus with other CoE nodes (e. g. PC, PLC, further b maXX® 4400 (ES), I/O modules).

Information according option and function modules for the device series b maXX®4400 is found in the manual 5.12008.

Information according the programming of the **b maXX BM4000** controller is found in the parameter manual 5.03039.

1.2 Content of instruction manual to POWERLINK Controlled Node

Each person being in charge of working at or with the device, must have read and understood the instruction manual **POWERLINK Controlled Node** 5.12072 and the instruction manual for **b maXX BM4000** 5.12008 before starting to work with the device. This applies even if the person concerned has already worked with such or similar devices or has been trained by the manufacturer.

1.3 Mounting and installation

The mounting of the option module **POWERLINK Controlled Node**is described in the instruction manual **POWERLINK Controlled Node**5.12072.

1.4 Address Setting

The address setting and baud rate of the **POWERLINK Controlled Node** are described in the instruction manual **POWERLINK Controlled Node** 5.12072.

1.5 XDD file

The XDD file is a XML file and is for the description of the function range of a **POWERLINK** device. It is an electronic data sheet of the **POWERLINK** device. The XDD file is used by the POWERLINK Managed Node or the bus configurators. The XDD file contains information on all objects supported by the Controlled Node, the network management and further features.

The name extension of the XDD file is *.xdd.

The file can be downloaded from the download area on Baumüller's home page www.baumueller.de.

2

FUNDAMENTAL SAFETY INSTRUCTIONS

In this chapter the dangers are prescribed, which can arise during parameterization of the Baumüller **b maXX 4000** controller unit and the meaning of the information sign is explained.

2.1 Safety notes and mandatories

WARNING!

Danger from modification of the parameter settings!

The change of parameters affects the behavior of the Baumüller-unit and consequently the behavior of the construction and its components. If you change the adjustments of the parameters, you may cause a dangerous behavior of the construction and/or of its components.

Therefore:

- After each modification of the parameter settings, a commissioning with consideration to all safety instructions and safety regulations must be executed.

2.2 Information sign

NOTE!

This note is a very important information.

2.2 Information sign

3

BASICS POWERLINK

3.1 Literature concerning POWERLINK

On behalf of basic information with reference to **POWERLINK** the following literature is recommended:

- [1]
  Ethernet **POWERLINK** Communication Profile Specification
  EPSG Draft Standard 301
  Ethernet POWERLINK Standardization Group (EPSG)

- [2]
  Ethernet **POWERLINK** XML Device Description
  EPSG Draft Standard 311
  Ethernet POWERLINK Standardization Group (EPSG)

- [3]
  www.ethernet-powerlink.org
  Ethernet **POWERLINK** Standardization Group (EPSG)
  Schaperstraße 18
  D-10719 Berlin

- [4]
  CANopen Device Profile for Drives and Motion Control
  CiA Draft Standard Proposal DSP-402
  CAN in Automation e.V.

3.2 Basic principles POWERLINK

POWERLINK Version 2 (Ethernet type 0x88ab) is a published fieldbus system on the basis of real-time Ethernet, that integrates the mechanisms of CANopen completely.

Twisted pair cables (100Base-TX) serve as physical basis.

Net work

POWERLINK enables users to choose any topology. The network can be realized as line structure, tree structure, star structure or ring structure, whereas combinations are allowed, too. There is the possibility of adding and removing devices during run time (Hot-Plugging).

Bus access

The bus can be accessed via the CSMA/CD procedure (Carrier Sense Multiple Access / Collision Detection). Collisions may occur, as each participant is allowed to start sending his message, after recognizing the necessary idle bus. In this case, the collisions will be detected (Collision Detection) and the sending will be repeated after a random time interval. This ensures a transmission without data loss. For that reason, it is, of course, necessary that each participant can clearly be identified in the network by the respective MAC address.

The application of switches may lead to undefined conditions in the network.

MAC addressing

Each participant can send messages unrequested. Therefore, a clear sender and destination address is needed which is achieved by the MAC address.

IP addressing

Class Cv4 address 192.168.100.0 shall be used as net ID of a POWERLINK network. Each network supports 254 IP addresses whereby the last byte of the IP address (host ID) should correspond to the node number (node ID) of the participant.

192.168.100.POWERLINK Node ID

Net ID Host ID

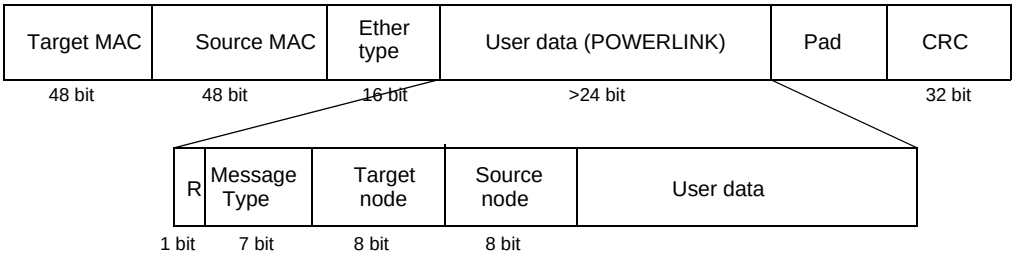
The following node IDs are reserved:

Address	Description
0x00	Invalid address
0xF0	**POWERLINK** default address of Managed Node
0xFB	Pseudo node address to be used by a node to adress itself
0xFC	**POWERLINK** dummy node address
0xFD	**POWERLINK** default address of diagnostic device
0xFE	Default address of **POWERLINK** gateways/router
0xFF	**POWERLINK** broadcast address

As this is a class C network, the subnet mask of the **POWERLINK** node shall be 255.255.255.0.

Ethernet frame

The Ethernet frame consists of a header and the data payload. The header consists of the destination and the source MAC address as well as of the Ethertype field that contains some control information. The Ethernet payload field, including the **POWERLINK** frame, contains at least 46 and up to 1500 bytes. Concluding, the correctness of the frame is ensured by means of a checksum.



POWERLINK frame

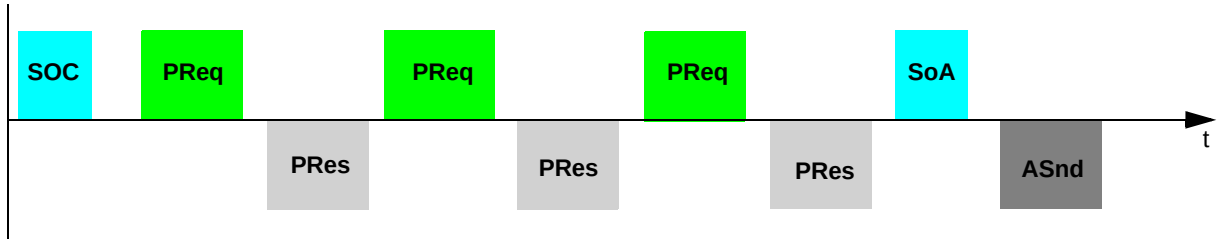
The **POWERLINK** frame consists of a header and the actual data payload as well. The header consists of a reserved-bit, the message type, the destination node and the source node. The following message types are defined.

Message type	ID	Designation	Use	Transfer type
SoC	0x01	Start of Cycle	Defines the start of a new cycle	Multicast
PReq	0x03	Poll Request	Ask for cyclic data of the CN	Unicast
PRes	0x04	Poll Response	Send current cyclic data of the CN	Multicast
SoA	0x05	Start of Asynchronous	Signalize the start of the asynchronous phase	Multicast
ASnd	0x06	Asynchronous Send	Sending of asynchronous data	Multicast

Determinism

The various participants in the network, the Controlled Nodes (CN), are controlled by a specific participant, the Managed Node (MN), and are only allowed to send, if they are asked to by the MN.

The **POWERLINK** cycle is divided into a synchronous and a asynchronous phase. At the beginning of the synchronous phase, the MN sends the SoC Frame. Subsequently, each single node is enquired by the MN with a PReq and responds with the PRes. After the cyclic phase, the MN starts the asynchronous phase with the sending of the SoA frame. A node determined by the MN can transmit acyclic data by means of a ASnd frame.



Device profile

POWERLINK supports the CANopen device profiles. These profiles describe application-specific and device-specific definitions, meaning of the data with regard to contents and device functionality. Amongst others, there are device profiles for drives, I/O modules, encoders or programmable devices. The option module **POWERLINK Controlled Node** for the **b maXX 4000** controller is implemented according to the device profile DSP402 (Drives and Motion Control).

Object directory

The main element of a device profile is the object directory.

The basis for this is the CANopen object directory:

Index (hex)	Object
0x0000	Not used
0x0001 - 0x001F	Static data types
0x0020 - 0x003F	Complex data types
0x0040 - 0x005F	Manufacturer specific complex data types
0x0060 - 0x007F	Device profile specific static data types
0x0080 - 0x009F	Device profile specific complex data types
0x00A0 - 0x03FF	Reserved
0x0400 - 0x041F	**POWERLINK** specific static data types
0x0420 - 0x04FF	**POWERLINK** specific complex data types
0x0500 - 0x0FFF	Reserved
0x1000 - 0x1FFF	Communication profile area
0x2000 - 0x5FFF	Manufacturer specific profile area
0x6000 - 0x9FFF	Standardized device profile area
0xA000 - 0xBFFF	Standardized interface profile area
0xC000 - 0xFFFF	Reserved

The objects are always addressed by means of an index (16 bit) and additionally a sub-index (8 bit).

3.3 Supported operation modes of the CANopen device profile DSP402

3.3.1 Short summery

The following operation modes are supported, i.e. all prescribed objects can be found on the option module .

Device Control	optional objects completely available
Homing Mode	optional objects completely available
Profile Position Mode	optional objects not available completely
Position Control Funktion	optional objects not available completely
Profile Velocity Mode	optional objects not available completely
Common Entries in the Object Dictionary (no prescribed objects available)	optional objects not available completely
Factor Group	optional objects not available completely

The following operation modes are not supported, i.e. at least one prescribed object is not available and optional objects may be available.

Velocity Mode	prescribed and optional objects are not available completely
Profile Torque Mode	three objects
Interpolated Position Mode	no objects

3.3.2 Operating modes and field bus objects

Operating modes

Device Control

All prescribed and all optional objects are supported.

Field bus object number	mandatory/ optional	Field bus object name
0x6040	mandatory	controlword
0x6041	mandatory	statusword
0x605A	optional	quick_stop_option_code
0x605B	optional	shutdown_option_code
0x605C	optional	disable_operation_option_code
0x605D	optional	stop_option_code
0x605E	optional	fault_reaction_option_code
0x6060	mandatory	modes_of_operation
0x6061	mandatory	modes_of_operation_display

Homing Mode

All prescribed and all optional objects are supported.

Field bus object number	mandatory/ optional	Field bus object name
0x607C	optional	home_offset
0x6098	mandatory	homing_method
0x6099 SIX 0 = 2	mandatory	homing_speed
0x609A	optional	homing_acceleration

Profile Position Mode

(**Positioning**)

All prescribed and partly all optional objects are supported.

Field bus object number	mandatory/ optional	Field bus object name
0x607A	mandatory	target_position
0x607D SIX 0 = 2	optional	software_position_limit
0x607F	optional	max_profile_velocity
0x6080	optional	max_motor_speed
0x6081	mandatory	profile_velocitiy
0x6083	mandatory	profile_acceleration

Field bus object number	mandatory/ optional	Field bus object name
0x6084	mandatory	profile_deceleration
0x6085	optional	quick_stop_deceleration
0x6086	mandatory	motion_profile _type

Position Control Function

All prescribed and partly all optional objects are supported.

Field bus object number	mandatory/ optional	Field bus object name
0x6062	optional	position_damand_value
0x6063	optional	position_actual_value*
0x6064	mandatory	position_actual_value
0x6066	optional	following_error_time_out
0x6067	optional	position_window
0x6068	optional	position_window_time
0x60FB SIX 0 =28	optional	position_control_parameter_set

Profile Velocity Mode

(**Speed control**)

All prescribed and partly all optional objects are supported.

Field bus object number	mandatory/ optional	Field bus object name
0x6069	mandatory	velocity_sensor_actual_value
0x606A	mandatory	sensor_selection_code
0x606B	mandatory	velocity_demand_value
0x606C	mandatory	velocity_actual_value
0x606F	optional	velocity_threshold
0x60F8	optional	max_slippage
0x60FF	mandatory	target_velocity

Common Entries in the Object Dictionary

No prescribed objects available, optional objects are supported partly.

Field bus object number	mandatory/ optional	Field bus object name
0x60FD	optional	digits_inputs
0x6510 SIX 0 = 8	optional	drive_date

Factor Group **(Group of user units)**

No prescribed objects available, optional objects are supported partly **.**

Field bus object number	mandatory/ optional	Field bus object name
0x608F SIX 0 = 2	optional	position_encoder_resolution
0x6090 SIX 0 = 2	optional	velocity_encoder_resolution
0x6092 SIX 0 = 2	optional	feed_constant

Velocity Mode (**Speed control**)

All prescribed and partly all optional objects are supported .

Field bus object number	mandatory/ optional	Field bus object name
0x6042	mandatory	vl_target_velocity
0x6043	mandatory	vl_velocity_demand
0x6044	mandatory	vl_control_effort
0x6045	optional	vl_manipulated_velocity
0x6046 SIX 0 = 2	mandatory	vl_velocity min_max_amount
0x604D	optional	vl_pole_number
0x604F	optional	vl_ramp_function_time
0x6050	optional	vl_slow_down_time
0x6051	optional	vl_quick_stop_time

Profile Torque Mode (**Torque control**)

One prescribed and two optional objects are supported .

Field bus object number	mandatory/ optional	Field bus object name
0x6071	mandatory	target_torque
0x6072	optional	max_torque
0x6077	optional	torque_actual_value

4

COMMUNICATION TO THE b maXX 4000 CONTROLLER

In this chapter the data communication between the **POWERLINK Controlled Node** and the **b maXX 4000** device is described.

4.1 Communication flow

The **POWERLINK Controlled Node**module exchanges via a Dual-Port-RAM in the FPGA data with the **b maXX 4000** controller. This data exchange is made with a defined time pattern via the BACI interface (Baumüller bus).

Therewith the option module activates the communication with the **b maXX 4000** controller. During communication, two different types of data are transferred:

- Process data
- Service data

Process data is always transferred cyclically. In the remaining time, service data is transferred. The transmission of the process data is made in a settable time pattern (Sync interval), transmitting the reference values and the actual values in the SYNC interval each with different offset. The cycle time of the SoC frame must be in accordance with the BACI cycle time.

4.2 Parameterizing the BACI communication times

Between the **POWERLINK Controlled Node**and the **b maXX 4000** controller 8 set values and 8 actual values can be exchanged as process data in a communication cycle. Which set values and actual values are exchanged is specified in the mapping objects on the **POWERLINK** Managed Node, see ▹Data Exchange and Parameterization◃ from page 27.

The setting of communication times between **POWERLINK Controlled Node**and **b maXX 4000** is automatically set by the option module. The SYNC interval of the controller, the BACI communication cycle time (rate set values, actual values), the cycle offset of the set values and the cycle offset of the actual values are set by means of the cycle time in object 0x1006 set in the Managed Node.

The **b maXX 4000** controller initiates a communication time slot every 125 µs, in which process data set values or process data actual values are transferred.

The communication cycle time is a multiple of the call of the communication time slice of the controller (every 125 µs).

The factor is calculated as follows:

$$\text{Cycle time set values, actual values} = \frac{\text{communication cycle time (in µs)}}{125\ \text{µs}}$$

The process date set values and the process data actual values are transmitted in different communication time slots. Therefore, different cycle offsets are stated for the set values and for the actual values. The cycle offset is nothing but the number of the communication time slot transmitting the data.

The configuration by the option card is as follows:

Cycle time (object 0x1006)	BACI cycle time (P0800)	Set value offset (P0818)	Actual value offset (P0819)
250 µs	2	0	1
500 µs	4	2	3
1000 µs	8	3	5
2000 µs	16	4	10
4000 µs	32	4	10
8000 µs	64	4	10

The settings must be stored in the data set of the **b maXX 4000** controller and the controller must be booted again.

NOTE!

If cyclic communication is interrupted, e. G.. at restart of the bus the error/warning Alive Counter or the error cyclic communication can occur.

4.3 Configuration possibilities of the POWERLINK Controlled Node in ProDrive.

ProDrive „option module G/H - configuration 1" (P0830 / P0840).

Depending on the slot the POWERLINK option card is plugged in.

NOTE!

Settings result in a modified behavior!

4.3.1 Changing the units of some fieldbus objects

Bit 2

0 Resolution for speed 1 rpm

1 New functions, alterations are quoted in the further course
Resolution for speed 1/10 rpm

The following objects are concerned:
0x6042, 0x6043, 0x6045, 0x606B, 0x606C, 0x607A, 0x607C, 0x6081, 0x6083, 0x6084, 0x6099 SIX 1, 0x6099 SIX 2, 0x609A, 0x60FF, 0x60FB SIX 29

4.3.2 CANopen offset

Bit 5

0 Conversion of the numerical scale from UINT32 to INT32; depending on the direction, an offset of 2^{31} is added/subtracted to the appropriate fieldbus object during the positioning.

1 No offset is added/subtracted

The following objects are concerned:
0x6062, 0x6064, 0x607A, 0x607C, 0x607D SIX1, 0x607D SIX2

4.3.3 BACI communication settings

Bit 10

0 Automatic configuration of the BACI communication times

1 No configuration of the BACI communication times by the option card

4.3.4 Standardization of speed 100% / 200%

Bit 11

0 Speed standardization related to 100% maximum speed.

1 Speed standardization related to 200% maximum speed

The following objects are concerned:
0x6042, 0x6046 SIX 2, 0x60FF, 0x60FB SIX 29

4.3.5 Gear factor

Bit 12

0 No gear.factor

1 Gear factor from controller parameter (numerator P3314, denominator P3315)

The following objects are concerned:
0x604C SIX 1, 0x604C SIX 2

4.3.6 Factor Group according to CiA®402

Bit 14

0 No calculation according to CiA®402 Factor Group

1 Calculation according to CiA®402 Factor Group

The following objects are concerned:
0x6064, 0x6067, 0x606C, 0x606F, 0x607A, 0x607C, 0x607D SIX1, 0x607D SIX2, 0x6081, 0x6099 SIX1, 0x6099 SIX2, 0x60FF

The calculation on the basis of the Factor Group occurs in the b maXX conroller (see also BM4400 Parameter Manual 5.03039).

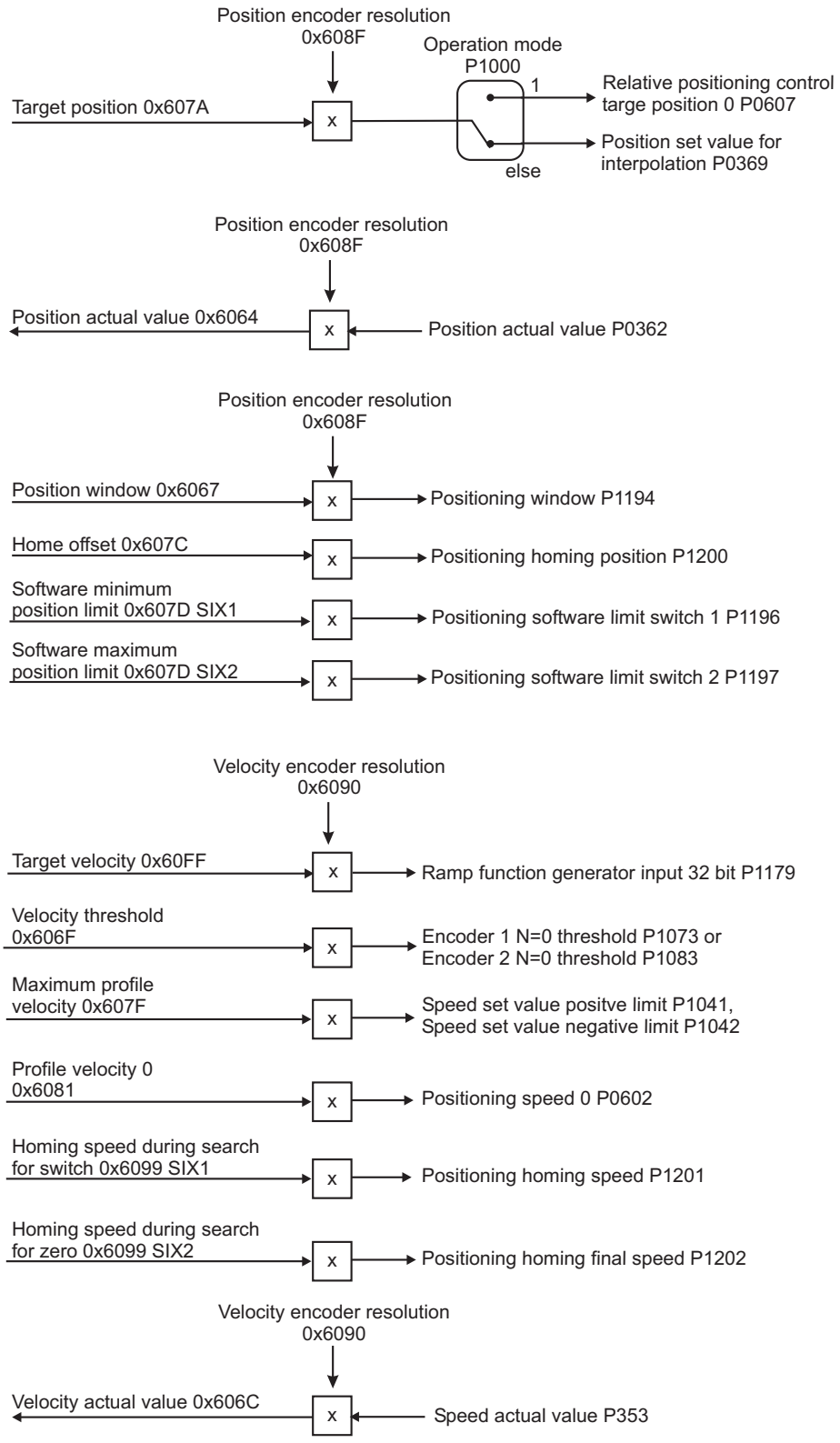


Figure 1: Calculation on the basis of the Factor Group

4.4 General notes according POWERLINK Controlled Node

NOTE!

Changes, which are executed via ProDrive, are not automatically updated on the **POWERLINK Controlled Node**. The access on the controller should be done with field bus objects via POWERLINK, if possible. Changes via ProDrive at the switching over between relative and absolute positioning modes are not noticed on the **POWERLINK Controlled Node** during the positioning operation. This also includes e. g. changes of the operation mode via ProDrive. The switchover/change must be made via the **POWERLINK**.

The following controller parameters are concerned:

P0830 / P0840	No object according to DSP 402, access only via the manufacturer-specific object possible (0x433E / 0x4348).
P1000	FBO 0x6060
P1031	FBO 0x6080
P3050	FBO 0x6092 SIX1
P3051	FBO 0x6092 SIX2
P0601	Internal switchover on the **POWERLINK Controlled Node** by the control word bit 6, operation mode Positioning, relative and absolute modes.
P0531	FBO 0x1006
P0532	FBO 0x1006
P0800	FBO 0x1006
P0818	FBO 0x1006
P0819	FBO 0x1006
P1190	FBO 0x6086

4.4.1 User units UU

The user units can be entered via ProDrive under „Scaling“. Save in the data set in order to save the user units even after a switching off.

If the desired user units have been set, they should be preserved even after subsequent updates of the controller. To be on the save side, check once again in ProDrive.

NOTE!

1 UU = 1 INC is set for the user units (UU) in the default data set.

The UU can be entered via **POWERLINK** by choosing object **0x6092.**

0x6092 feed constant = feed / driving shaft revolutions

Internally, „driving shaft revolutions“ is multiplied with 65536 on the **POWERLINK Controlled Node**. Maximum input for „feed“ (UU) is 0 ... 2^{24} - 1.

SIX 1 = feed [in User Units e.g. 360.00 degrees, 1/100 degree resolution]
Is displayed in the **b maXX 4000** controller on P3050 and can be stored in the data set.

SIX 2 = driving shaft revolutions [internally 1 revolution is multiplied with 65536 [INC] at **POWERLINK Controlled Node**].
Is displayed in the **b maXX 4000** controller on P3051 and can be stored in the data set.
The number of revolutions is limited to 255.

The input via the fieldbus 360.00 degrees is converted to 65536 increments for one revolution on the **POWERLINK Controlled Node**.

Example: position set value in UU = 36000 corresponds to 360.00 degrees (corresponds to 65536 increments).

The conversion for the position reference value **0x607A** on the **POWERLINK Controlled Node** is as follows:

0x607A Position set value [INC] in the **b maXX 4000**

= Fieldbus object [UU] * driving shaft revolutions * 65536 [INC] / feed [UU]

= 36000 * 1 * 65536 / 36000 [UU * INC / UU]

= 65536 [INC]

NOTE!

The calculation of the user units is very time-consuming and should not be effected in cyclic operation (position and synchronization). If the user units are set to 1:1, the calculation of the UU is not applicable.

The UU have an effect on the following fieldbus objects:

0x6062, 0x6063, 0x6064, 0x6067, 0x607A, 0x607C, 0x607D SIX1, 0x607D SIX2, 0x6081, 0x6083, 0x6084, 0x6085, 0x6099 SIX 1, 0x6099 SIX 2, 0x609A

4.4.2 CANopen offset

Display of the number scale UINT32 on INT32 (CANopen-Mode). On writing/reading on some FB objects, an offset of 2^{31} is added respectively subtracted internally on the **POWERLINK** option card depending on the direction.

If the position reference values and the target position shall also be displayed in ProDrive in the INT32 number scale, it is possible to activate a check box for the offset on the page „Scaling“.

The CANopen offset has an influence on the following fieldbus objects:

(0x6062, 0x6064, 0x607A, 0x607C, 0x607D SIX1, 0x607D SIX2) - 2^{31}

($0x607A_{hex}$, $0x607C_{hex}$,$0x\ 607D_{hex}$ Sub ½) + 2^{31}

4.4.3 Homing necessary for positioning

In ProDrive you can choose on page „Homing“ with the provided check box, whether the drive permits a positioning, if no first-time homing has been executed.

Deactivated:
No homing is necessary for operating in operating mode positioning.

Activated:
If the drive is activated in the operating mode ’positioning’ and no homing has been executed before,

an error message (controller message No. 200) is displayed and the drive remains position-controlled in the current position. Positioning jobs are not executed. They will only be executed, after a homing has been completed (at least once after switching on). The error message can only be acknowledged, if a homing has been executed. After the homing, the positioning can be started.

NOTE!

A homing is necessary, in case the CANopen mode is defined as standard!

4.4.4 Types of positioning, depending on the positioning mode (P0601)

NOTE!

Make sure that also the positioning data set 0 is set in ProDrive under positioning 0, otherwise the positioning will not be effected correctly in **POWERLINK**. The switching between the positioning modes „relative“, „negative/positive“ and „absolute“ only takes place by means of the control word. A homing should always precede the positioning in the CANopen mode (standard).

There are the following positioning modes:

Positioning mode P0601	Description
„Absolute/relative“ CANopen (standard value 9)	• Destination is displayed in P0607 (INT32) • Switching „absolute/relative“ via control word, only
„Relative, positive and negative“ (value 4)	• Destination is displayed in P0607 (INT32) • no switching „absolute/relative“ via control word
„Absolute relative“ (value 10)	• Destination is displayed in P0600 (UINT32) • switching „absolute/relative“ via control word, only
All other modes	• Destination is displayed in P0600 (UINT32) • no switching „absolute/relative“ via control word • no conversion (data type = UINT32)

Switching „absolute/relative“, via control word Bit 6

Bit 6

0 Absolute

1 Relative

The conversion of data type INT32 ⇔ UINT32 means that an offset of 2^{31} is added or substracted depending on the direction. This is necessary in order to achieve a standard presentation of the fieldbus object in data type INT, as some controllers parameters are realized for the positioning (see ▹CANopen offset◃ on page 19) as data type UINT. Thus, the existing fieldbus objects are displayed to the user in the positioning as data type INT.

The application of offset 2^{31} may be deactivated, if needed. Here in ProDrive under „Option module G, H - Configuration 1".

Bit 5

0 Number scale switching from UINT32 to data type INT32; on positioning, an offset of 2^{31} is added to the corresponding FBO depending on the direction.

1 No offset is used.

NOTE!

The conversion of the positioning mode to P0607 „Absolute/relative CANopen“ is not being deactivated.

5

DATA EXCHANGE AND PARAMETERIZATION

The access to data or parameter is made at **POWERLINK** via CANopen objects.

Accordant to profile structure it is differed between objects for communication control (indices 0x1XXX) and user- or device-specific objects. The latter are divided into objects according to profile DSP 402 (indices 0x6XXX) and manufacturer-specific objects (indices 0x4XXX.

A listing of the 4XXX and the 6XXX objects are to be found in ▹Appendix B - Quick reference◃ from page 47.

NOTE!

With manufacturer-specific objects (0x4XXX) the object index results from

$0x4000_{hex}$ + **b maXX 4000** parameter number in hexadecimal,

e. g. object 0x412C is transferred to **b maXX 4000** parameter P0300, the control word. These objects only have subindex 0x00.

Object = 0x4000 + parameter number in hex

5.1 Directory of objects for communication control

In this section all objects of the communication-specific area of the object directory are to be found, which are supported by the Baumüller **POWERLINK Controlled Node** in accordance with EPSG DS301.

Name	Index	Subindex	Data type	Default value
NMT_DeviceType_U32	0x1000	0x00	UINT32	0x00020192

This object is read-only and contains information on the related device (drive in accordance with DSP402).

Name	Index	Subindex	Data type	Default value
NMT_CycleLen_U32	0x1006	0x00	UINT32	0x1000

If the sync frame is activated, the sync interval has to be set in accordance with the time of the sync frame (1000 µs, 2000 µs, 4000 µs or 8000 µs). The set time has an effect on the parameter P0532 (sync interval) of the b maXX BM4000 controller.

Name	Index	Subindex	Data type	Default value
NMT_ManufactDevName_V	0x1008	0x00	VAR	-

This object is read-only. It contains the following character strings: „b maXX 4400“.

Name	Index	Subindex	Data type	Default value
NMT_ManufactHwVers_VS	0x1009	0x00	VAR	-

This object is read-only. It contains the present hardware version of the option module, e. g. the character string: „01.00“.

Name	Index	Subindex	Data type	Default value
NMT_ManufactSwVers_VS	0x100A	0x00	VAR	-

This object is read-only. It contains the present POWERLINK Stack version of the option module e. g. the character string „EPL V2 V1.8 r1“.

Name	Index	Subindex	Data type	Default value
NMT_IdentityObject_REC	0x1018	0x00	UINT8	0x04
VendorID_U32		0x01	UINT32	0x0000015A
ProductCode_U32		0x02	UINT32	0x00010000
RevisionNo_U32		0x03	UINT32	0x00010000
SerialNo_U32		0x04	UINT32	0x00000000

In this object there is information about the device.
RevisionNo_U32 contains the current version of firmware e. g. 00010002 for FW 01.02.

Name	Index	Subindex	Data type	Default value
CFM_VerifyConfiguration_REC	0x1020	0x00	UINT8	0x02
ConfDate_U32		0x01	UINT32	0x00000000
ConfTime_U32		0x02	UINT32	0x00000000

This object contains the information on the local configuration time of the device.

ConfDate_U32 contains the configuration time which including the days since 01.01.1984.

Name	Index	Subindex	Data type	Default value
NMT_InterfaceGroup_Xh_REC	0x1030	0x00	UINT8	0x09
InterfaceIndex_U16		0x01	UINT16	0x0001
InterfaceDescription_VSTR		0x02	VISIBLE_STRING	„interface 1“
InterfaceType_U8		0x03	UINT8	0x06
InterfaceMtu_U16		0x04	UINT16	1500
InterfacePhysAddress_OSTR		0x05	OCTET_STRING6	0x06
InterfaceName_VSTR		0x06	VISIBLE_STRING	„interface 1“
InterfaceOperStatus_U8		0x07	UINT8	0x01
InterfaceAdminState_U8		0x08	UINT8	0x01
Valid_BOOL		0x09	BOOL	0x1

Parameters of the network interface are configurated by means of this object via SDO.

Name	Index	Subindex	Data type	Default value
SDO_SequLayerTimeout_U32	0x1300	0x00	VAR	-

This object contains the timeout value in [ms] for the recognition of an SDO abort.

Name	Index	Subindex	Data type	Default value
PDO_RxCommParam_00h_REC	0x1400	0x00	UINT8	0x02
Node_ID_U8		0x01	UINT8	0x00
MappingVersion_U8		0x02	UINT8	0x00

This object contains information on the receive PDO.

Name	Index	Subindex	Data type	Default value
PDO_RxMappParam_00h_REC	0x1600	0x00	UINT8	0x01
		0x01	UINT64	0x0010000000006040
		:	:	
		n	UINT64	

This object contains the information of receive-PDO. The total number of the following entries is in subindex 0x00. The control word is entered according to default in subindex 0x01 (object 0x6040 subindex 0x00 length 0x10).

The total number of the mapped objects may not exceed the BACI reference value limits of a maximum of 8 objects. (also see ▹PDO Mapping◃ from page 38).

Name	Index	Subindex	Data type	Default value
PDO_TxCommParam_00h_REC	0x1800	0x00	UINT8	0x02
Node_ID_U8		0x01	UINT8	0x00
MappingVersion_U8		0x02	UINT8	0x00

This object contains information on the transmit PDO.

Name	Index	Subindex	Data type	Default value
PDO_TxMappParam_00h_REC	0x1A00	0x00	UINT8	0x02
		0x01	UINT64	0x0010000000006041
		:	:	
		n	UINT64	

This object contains the contents of transmit PDO. The total number of the following entries are in subindex 0x00. The status word is entered according to default in subindex 0x01 (object 0x6041 subindex 0x00 length 0x10).

The total number of mapped objects may not exceed the BACI reference value limits of a maximum of 8 objects. (also see ▹PDO Mapping◃ from page 38).

5.2 Net work management (NMT)

Commands of the network management serve mainly the control of the communication states in **POWERLINK** net.

The state diagram of the communication of the **POWERLINK Controlled Node**is illustrated here:

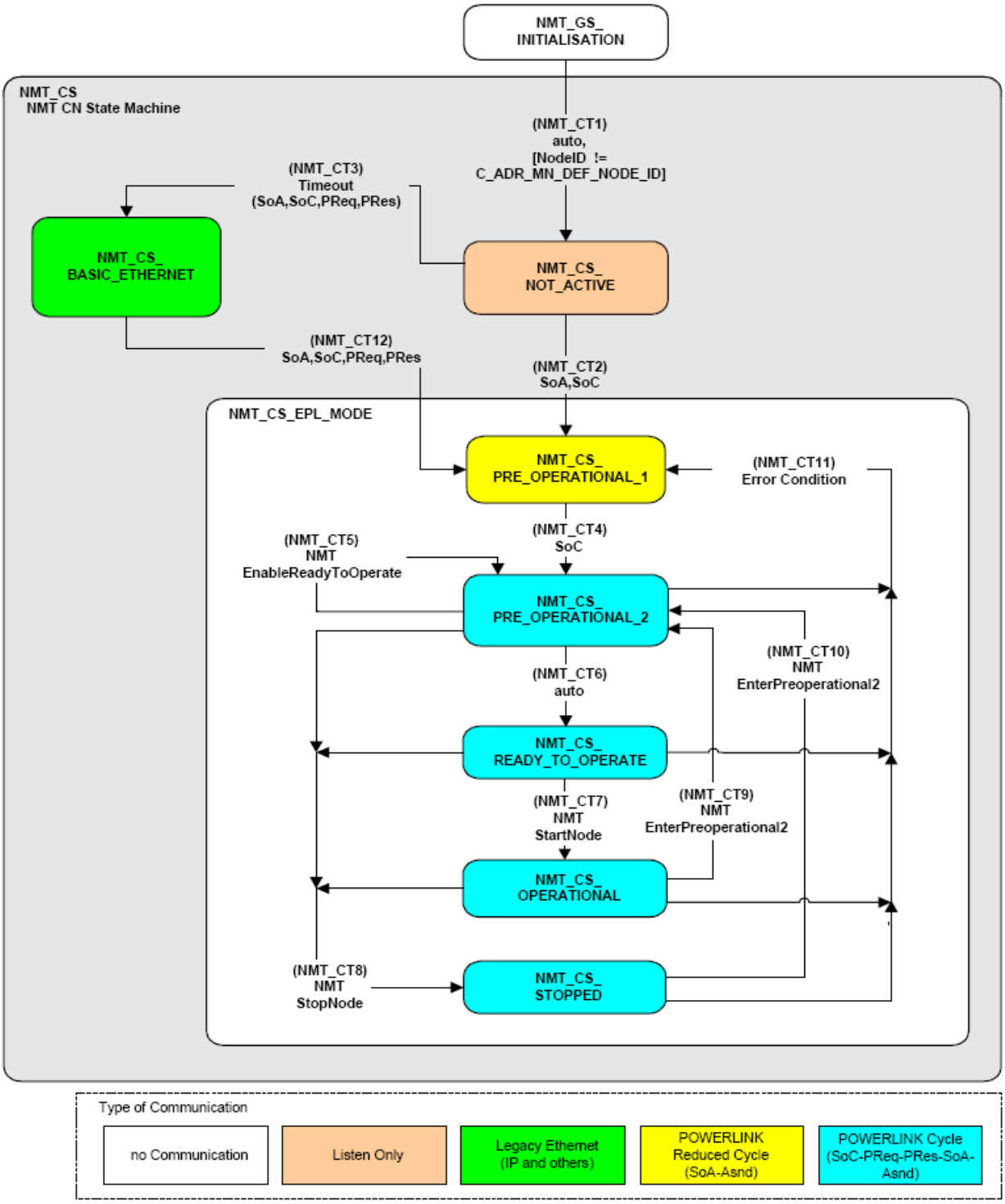


Figure 2: State diagram **POWERLINK Controlled Node**

State	Description
	Transient state for starting the CN and for identifying in the network
NMT_CS_PRE_OPERATIONAL_1	Identification of the node by the MN via IdentRequest
NMT_CS_PRE_OPERATIONAL_2	PDO configuration by the MN; configuration examination
NMT_CS_READY_TO_OPERATE	Signalizing the service readiness to the MN; Starting the PDO communication
NMT_CS_OPERATIONAL	Normal operating state of the CN; CN takes part at the cyclic data exchange
NMT_CS_STOPPED	CN is passive; controlled shut down of the CN; CN does not take part at the cyclic data exchange
NMT_CS_BASIC_ETHERNET	CN takes part at usual ethernet communication according to IEEE 802.3

5.3 Service data (SDO)

Service data objects (SDO) serve as an exchange of messages without real-time requests. SDOs are used for parameterizing CNs and for setting the communication references for PDOs. Access on data occurs only via the object list. SDOs are always acknowledged data, i. e. the transmitter receives an acknowledge from the receiver. The data exchange by means of SDOs can only be executed asynchronously (see also ▹Synchronization (SYNC)◃ from page 36).

SDOs follow the client-server-model. The client (MN) initiates the communication and the server (CN) responds to it. A server is not able to start a SDO communication. The Baumüller POWERLINK option module supports a server SDO but no client SDO.

The MN starts the asynchronous cycle by sending the Start of Asynchronous (SoA) frame. The SDO transfer is answered by the CN via ASnd.

5.3.1 Frame structure SoA

The structure of the Start of Asynchronous (SoA) frame is as follows:

	Bit offset							
Byte offset	7	6	5	4	3	2	1	0
0	Message Type = SoA (0x05)							
1	Destination = Broadcast address (0xFF)							
2	Source = Node ID of the MN (0xF0)							
3	NMTStatus							
4	res	res	res	res	res	EA/res	ER/res	res
5	reserved							
6	RequestedServiceID (see the following table)							
7	RequestedServiceTarget							
8	EPLVersion							
9 ... 45	reserved							

EA (Exception Acknowledge)
ER (Exception Reset)

RequestedService	**ID**	**Description**
	0x00	Asynchronous slot is not assigned to any node
IdentRequest	0x01	Identification of inactive nodes
StatusRequest	0x02	Request the status and error information of single nodes
NMTRequestInvite	0x03	Requesting a node to send an indicated NMT command.
Manufacturer specific	0xA0 .. 0xFE	Manufacturer specific purposes
UnspecificInvite	0xFF	Requesting a node to send an indicated transmit request.

5.3.2 Frame structure ASnd

The structure of the asynchronous Send (ASnd) frame is as follows:

	Bit offset							
Byte offset	7	6	5	4	3	2	1	0
0	Message Type = ASnd (0x06)							
1	Destination (Node ID of the addressed nodes)							
2	Source (Node ID of the transmitting node)							
3	Service ID (see the following table)							
4 .. 7	Sequence Layer Protocol							
8 .. k-1	Command Layer Protocol							
k .. 1472	SDO Payload Data							

RequestedService	**ID**	**Description**
	0x01	Response of a node to an IdentRequest via SoA
StatusResponse	0x02	Response of a node to a StatusRequest via SoA
NMTRequest	0x03	Response of a CN to a NMTRequestInvite via SoA
NMTCommand	0x04	Response of the MN to an internal request or an external request via NMTRequest
SDO	0x05	Response of a CN to an UnspecificInvite via SoA
Manufacturer specific	0xA0 .. 0xFE	Manufacturer specific service

5.3.3 Error reactions

Faulty SDO accesses are rejected by means of abort codes. The structure of these abort frames is identical to the SDO frame described in ▹Frame structure SoA◃ on page 33.

The data field contains an abort code with 4 bytes.

The following different messages may occur in case of faulty SDO accesses:

Abort code	Description
	SDO protocol timed out.
0x05040001	Unknown command ID
0x05040002	Invalid block size
0x05040003	Invalid sequence number
0x05040005	Out of memory
0x06010000	Unsupported access to an object
0x06010001	Attempt to read a write-only object
0x06010002	Attempt to write a read-only object
0x06020000	Object does not exist in the object dictionary
0x06040041	Object cannot be mapped to the PDO.
0x06040042	The number and length of the objects to be mapped would exceed PDO length
0x06040043	General parameter incompatibility
0x06040044	Invalid heartbeat declaration
0x06040047	General internal incompatibility in the device
0x06060000	Access failed due to an hardware error
0x06070010	Data type does not match, length of service parameter does not match
0x06070012	Data type does not match, length of service parameter too high
0x06070013	Data type does not match, length of service parameter too low
0x06090011	Sub-index does not exist
0x06090030	Value range of parameter exceeded (only for write access)
0x06090031	Value of parameter written too high
0x06090032	Value of parameter written too low
0x06090036	Maximum value is less than minimum value
0x08000000	General error
0x08000020	Data cannot be transferred or stored to the application.
0x08000021	Data cannot be transferred or stored to the application because of local control
0x08000022	Data cannot be transferred or stored to the application because of the present device state.
0x08000023	Object dictionary is not available
0x08000024	Configuration data set is empty

5.4 Synchronization (SYNC)

The Start of Cycle (SoC) frame is used for synchronizing the CNs. This frame is unconfirmed (multicast) and is sent by the MN. It does not contain any data. The **POWERLINK** Controlled Node is able to receive SoC frames.

The receipt of a SoC frame generates an interrupt on the **POWERLINK Controlled Node**; this interrupt is forwarded to the **b maXX 4000** controller. Thus, this signal can be used for the synchronization of the **b maXX 4000** controller. Within a communication cycle all corresponding frames have to be sent to all configured slaves and in doing so the number of nodes and the processing time have to be considered. The cycle time for the SoC frame is set in object 0x1006. For this see ▹Directory of objects for communication control◃ from page 28. Furthermore, the communication cycle time has to be stored in the data set of the controller.

The so-called multiplexing in **POWERLINK** allows not to request single nodes in each cycle. Thus, several nodes can share one time section in the transmission phase, so that the bandwidth of the synchronous phase can be used in an optimum way on the **POWERLINK** bus. The configuration of this assignment is effected by the Managed Node.

5.4.1 Frame structure SoC

The structure of the Start of Cycle (SoC) frame is as follows:

	Bit offset							
Byte offset	7	6	5	4	3	2	1	0
0	Message Type = SoC (0x01)							
1	Destination = Broadcast address (0xFF)							
2	Source = Node ID of the MN (0xF0)							
3	reserved							
4	MC	PS	res	res	res	res	res	res
5	reserved							
6 ... 13	NetTime (optional)							
14 ... 21	RelativeTime (optional)							
22 ... 45	reserved							

MC (Multiplexed Cycle Completed)
PS (Prescaled Slot)

5.5 Process data (PDO)

Process data objects are unconfirmed frames that are optimized for the exchange of data with real-time requirements. There are two kinds of PDOs, differing in the direction of the data transmission from the perspective of the device. The **POWERLINK Controlled Node** for **b maXX 4000** controllers supports a transmit PDO (TPDO) as well as a receive PDO (RPDO). Up to eight objects can be transmitted in each PDO.

The PDO communication in **POWERLINK** is executed by synchronous PReq respectively PRes frames. In the synchronous phase the MN sends the PollRequest (PReq) as unicast frame. The corresponding CN sends the PollResponse (PRes) as broadcast.

The format of the data exchange has to be defined before starting the communication between sender and receiver (mapping). The sending and receiving of PDOs can be initiated in different kind of ways.

NOTE!

All objects, which were configured in the PDOs are transmitted between the **POWERLINK Controlled Node** and the **b maXX 4000** controller as cyclic data (also see ▹Communication flow◃ from page 17). As the cyclic data transmission is only made in the state of NMT_CS_OPERATIONAL, the communication monitoring in ProDrive BACI should be only in this status be activated (timeout for cyclic communication **P0836** BACI).

5.5.1 Frame structure PReq and PRes

The structure of the PollRequest (PReq) frame is as follows:

	Bit offset							
Byte offset	7	6	5	4	3	2	1	0
0	Message Type = PReq (0x03)							
1	Destination = Node ID of the CN							
2	Source = Node ID of the MN (0xF0)							
3	reserved							
4	res	res	MS	res	res	EA	res	RD
5	reserved							
6	PDOVersion							
7	reserved							
8 ... 9	Size (Size of the process data in byte)							
10 ... n	Payload							

MS (Multiplexed Slot)
EA (Exception Acknowledge)
RD (Ready)

The structure of the PollResponse (PRes) frame is as follows:

	Bit offset							
Byte offset	7	6	5	4	3	2	1	0
0	Message Type = PRes (0x04)							
1	Destination = Broadcast Address (0xFF)							
2	Source = Node ID of the CN							
3	NMT status							
4	res	res	MS	EN	res	res	res	RD
5	res	res	PR			RS		
6	PDOVersion							
7	reserved							
8 ... 9	Size (Size of the process data in byte)							
10 ... n	Payload							

MS (Multiplexed Slot)
EN (Exception New)
RD (Ready)
PR (Priority)
RS (Request To Send)

5.5.2 PDO Mapping

Mapping is a method of assigning variables/objects to PDOS. With these PDOs these variables/objects are transmitted via **POWERLINK**. Due to mapping the cyclic data exchange is configured. SDOs are used for the parameterization. The mapping is set via addressable objects in the object library.

Two objects each exist for the TPDO and the RPDO (see also ▹Directory of objects for communication control◃ from page 28). One of these objects determines the contents of the PDO, the second one determines the communication relation respectively triggering.

Process data object	Object for content	Object for the communication relation
TPDO	0x1A00	0x1800
RPDO	0x1600	0x1400

NOTE!

The mapping **cannot** be changed in state NMT_CS_OPERATIONAL. A new mapping will be activated only after transition to NMT_CS_READY_TO_OPERATE.

A maximum of 1490 bytes are provided by the PReq respectively the PRes data frame for the payload data transmission. The **POWERLINK Controlled Node** is able to transmit the contents of up to eight variables / objects in each direction. The logic content of the user data is determined by the mapping.

Specific information about the objects to be mapped are needed for this determination:

Object index, subindex and the length of the date as well as the sequence of the objects to be mapped. The objects to be mapped are written from the object directory to the subindices starting with 0x01 of the mapping object (0x1600, 0x1A00), e. g. the value 0x0010.0000.0000.6040 is entered in object 0x1600 subindex 0x01. This means that the first two bytes (length 0x0010, offset 0x0000) of the data received in RXPD are written on the control word (object 0x6040, Subindex 0x00). The object 0x6040 is transferred to the **b maXX 4000** parameter **P0300** control word (see also ▹Appendix C - Conversion tables◃ from page 53). This means that the first word of the frame received in RPDO is written on the control word of the **b maXX 4000**. The number of objects to be mapped (number of subindices occupied with valid objects) has to be entered in subindex 0x00.

The structure of the subindices for the mapping objects 0x1600 and 0x1A00 is as follows:

Byte offset	Name	Description
	Index	Index of the object to be mapped
2	Subindex	Subindex of the object to be mapped
3	Reserved	
4 ... 5	Offset	Offset related to start of PDO payload (Bit count)
6 ... 7	Length	Length of the object to be mapped (Bit count)

NOTE!

On setting the mapping in the mapping parameters (0x1600, 0x1A00), it is necessary to describe the respective subindex 0x00 with the correct number of mapped objects.

Set values:

The permissible cyclical set values are marked in a table with the column 'PDO mapping' as 'RX', see table ▹B.2 6000 object numbers (device profile DSP 402)◃ from page 48. The manufacturer-specific parameters (four thousands objects) must be checked up in the parameter manual **b maXX 4000** (5.03039), chapter 7.1.4 attributes.

Actual values

The permitted cyclic actual values are marked in a table in column „PDO mapping“ as „TX“, see table ▹B.2 6000 object numbers (device profile DSP 402)◃ from page 48. The manufacturer-specific parameters (four thousands objects) must be checked up in the parameter manual **b maXX 4000** (5.03039), chapter 7.1.4 attributes.

Incorrect mapping configurations (invalid objects in 1600_{hex}, $1A00_{hex}$) are signalled with abort codes via SDO.

The cyclic set-/actual values are continuously initialized into the BACI, i. e. the first setpoint of RPDO is on first position in the BACI, the second setpoint on second position a.s.o. Analog for the actual value initialization the first actual value of TPDO is on first position in the BACI, the second actual value of TPDO on second position a.s.o.

NOTE!

In case the controller status word (0x6041) is requested for the cyclic communication, it must be entered in the TPDO in the first position!

In case the status word is not requested, it is entered in the BACI nevertheless, but in this case it is not considered regarding the field bus transmission. Here the status word is necessary for internal requests and then only 7 further actual values can be used.

Writing of similar field bus objects (FBO) via service data SD and process data PD.

Normally, PD write accesses overwrite SD write accesses cyclically on the same FBO. In single cases a write access via SD may have been successful, but this is not certain.

NOTE!

In this context you please avoid access to the same field bus object via SD and via PD.

6

CONFIGURATION EXAMPLE WITH B&R X20 PLC

The following chapter describes the configuration of the **POWERLINK Controlled Node** for **b maXX 4000** controller with a B&R X20 PLC by means of Automation Studio (V3.0.90.22).

To integrate the **POWERLINK Controlled Node** for **b maXX 4000**, the XDD-file must be imported into the Automation Studio Project.
In this connection the device description file BM_POWERLINK_CiA402_CN.xdd must be chosen in the menu under *Tools* ⇒ *Import Fieldbus Device*…. The file can be downloaded from the download area on Baumüller's homepage.

As the device description is store in the Automation Studio project file, this process must be repeated on creating a new project.

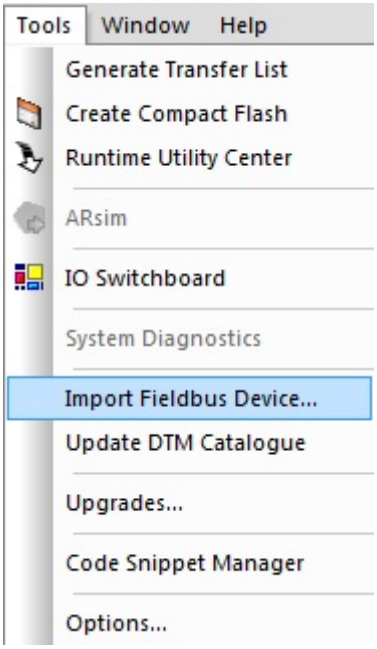


Figure 3: Configuration - Import fieldbus device

Now the option module can be inserted into the POWERLINK interface of the PLC under *Physical View ⇒ Open POWERLINK:*

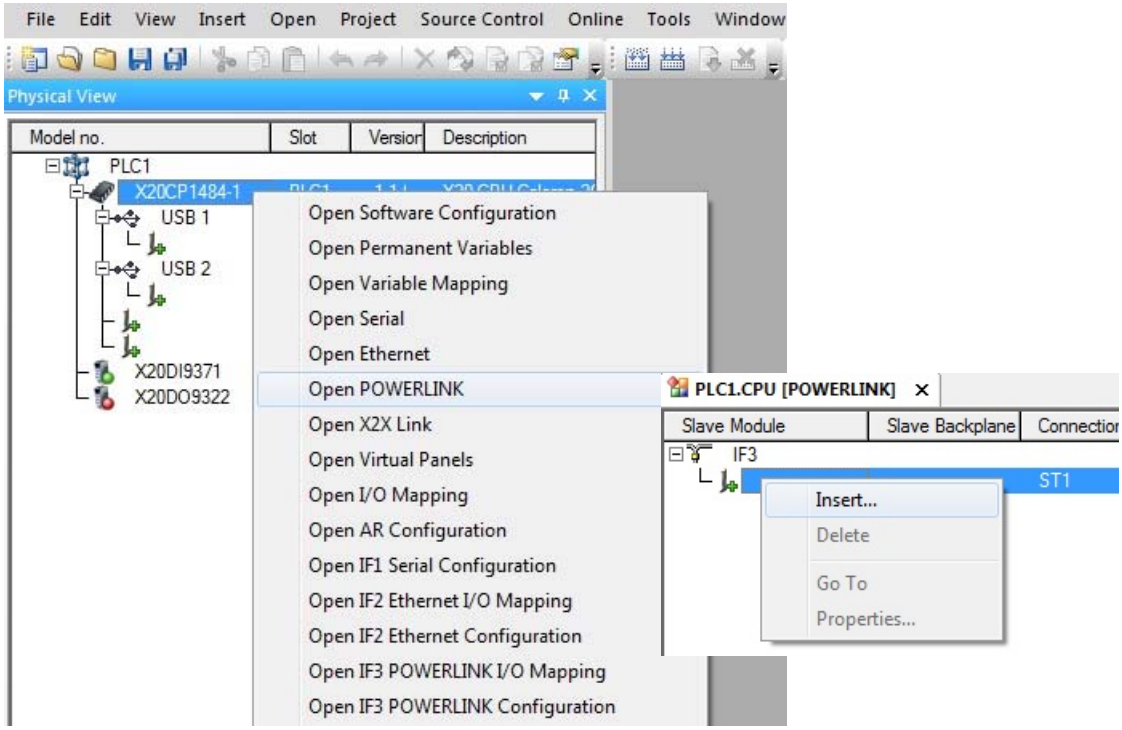


Figure 4: Configuration - Physical View - open POWERLINK

The option module is displayed as ***b maXX 4400*** **Powerlink controlled node** under POWERLINK devices.

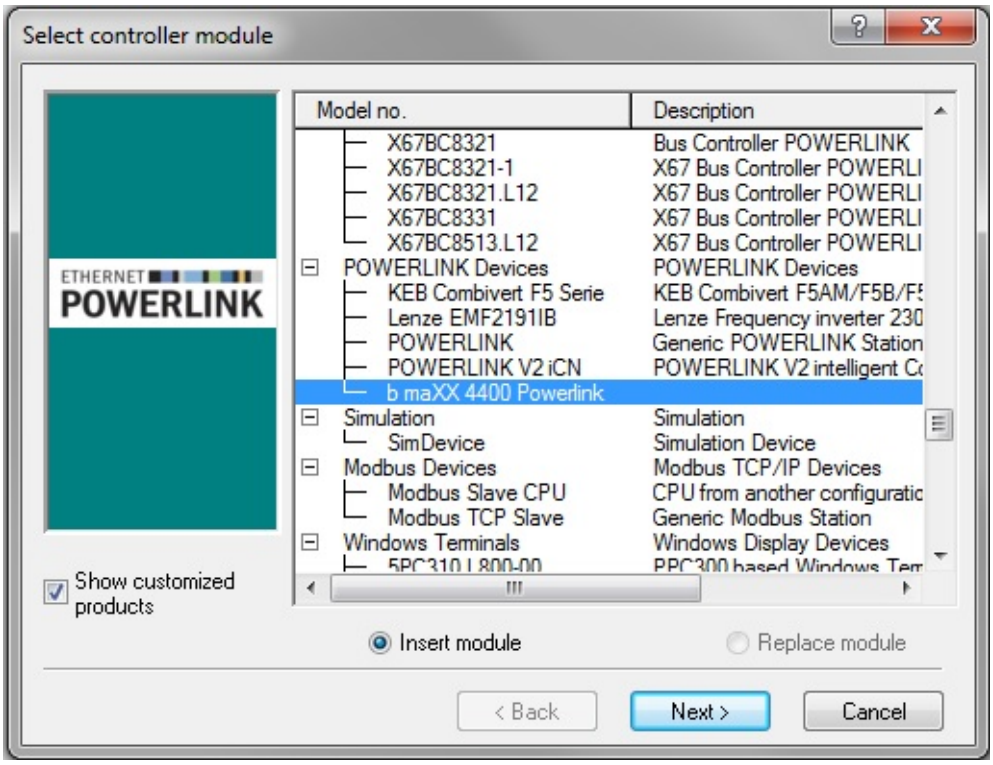


Figure 5: Configuration - Select controller module

Following a successful import, the option module appears in the Automation Studio *Physical View*.

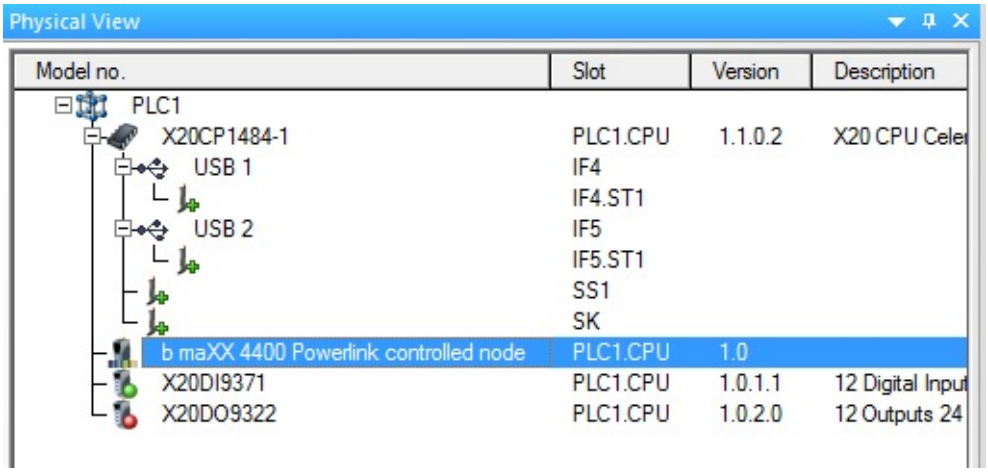


Figure 6: Configuration - option module in Physical View

APPENDIX A - ABBREVIATIONS

BACI	Baumüller drives serial interface
CD	Collision Detection
CN	Controlled Node
CSMA	Carrier Sense Multiple Access
DSP	Draft Standard Proposal
FBO	Field bus object
ID	Ident number
MAC	Media Access Control
MN	Managed Node
NMT	Net work management
PC	Personal Computer
PDO	Process Data Object
SDO	Service Data Object
SIX	Subindex
SPS	Storage-programmable logic controller
SYNC	Synchronization
XDD	XML Device Description

APPENDIX B - QUICK REFERENCE

The following quick reference shows the connection between CANopen object numbers and the **b maXX 4000** controller parameter numbers (see parameter manual **b maXX 4000** 5.03039).

B.1 4000 object numbers (manufacturer-specific objects)

Manufacturer-specific objects result from
0x4000 + parameter number (hex).
The subindex for all 4000-parameters always is 0x00.

Example — Parameter **P0053** ⇒ object index 0x4035 subindex 0x00

B.2 6000 object numbers (device profile DSP 402)

It is possible to access some parameters of the controller via 4000-objects **as well as** via one or several 6000s.

There are only a few parameters, that can exclusively be accessed by means of a 6000 parameter (0x606A).

NOTE!

There may be different standardizations for the 6000 objects and the 4000 objects!

TX: Transmit; RX: Receive; r: read; w: write; ro: read only; wo: write only

CANopen object number		Parameter No.	PDO mapping	Access type	Operating mode acc. to DSP 402
Index	Sub index				
0x6040	0x00	**P0300**	TX / RX	rw	Device control
0x6041	0x00	**P0301**	TX	ro	Device control
0x6042	0x00	**P1179**	TX / RX	rw	Velocity mode
0x6043	0x00	**P0351**	TX	ro	Velocity mode
0x6044	0x00	**P0353**	TX	ro	Velocity mode
0x6045	0x00	**P0352**	TX	ro	Velocity mode
0x6046	01	**P1041**	TX	ro	Velocity mode
0x6046	02	**P1041**, **P1042**	TX / RX	rw	Velocity mode
0x604D	01	**P0065**	TX	rw	Velocity mode
0x604F	0x00	**P1172**	TX / RX	rw	Velocity mode
0x6050	0x00	**P1173**	TX / RX	rw	Velocity mode
0x6051	0x00	**P1174**	TX / RX	rw	Velocity mode
0x605A	0x00	**P1004**	TX	rw	Device control
60x05B	0x00	**P1005**	TX	rw	Device control
0x605C	0x00	**P1006**	TX	rw	Device control
0x605D	0x00	**P1003**	TX	rw	Device control
0x605E	0x00	**P1007**	TX	rw	Device control
0x6060	0x00	**P1000**	TX / RX	rw	Device control

CANopen object number		Parameter No.	PDO mapping	Access type	Operating mode acc. to DSP 402
Index	Sub index				
0x6061	0x00	**P0304**	TX	ro	Device control
0x6062	0x00	**P0463**	TX	ro	Position control function
0x6063	0x00	**P0362**	TX	ro	Position control function
0x6064	0x00	**P0462**	TX	ro	Position control function
0x6066	0x00	**P1056**	TX	rw	Position control function
0x6067	0x00	**P1194**	TX / RX	rw	Position control function
0x6068	0x00	**P1195**	TX / RX	rw	Position control function
0x6069	0x00	**P0391**	TX	ro	Profile velocity mode
0x606A	0x00	-	-	ro	Profile velocity mode
0x606B	0x00	**P0352**	TX	ro	Profile velocity mode
0x606C	0x00	**P0353**	TX	ro	Profile velocity mode
0x606F	0x00	**P1073**	TX / RX	rw	Profile velocity mode
0x6071	0x00	**P0331**	TX / RX	rw	Profile torque mode
0x6072	0x00	**P0357**	TX / RX	rw	Profile torque mode
0x6077	0x00	**P0508**	TX	ro	Profile torque mode
0x607A	0x00	**P0369/ P0607**	TX / RX	rw	Profile position mode
0x607C	0x00	**P1200**	TX / RX	rw	Homing mode
0x607D	0x01	**P1196**	TX	rw	Profile position mode
0x607D	0x02	**P1197**	TX	rw	Profile position mode
0x607F	0x00	**P0057**	TX	rw	Profile position mode
0x6080	0x00	**P1031**	TX	rw	Profile position mode
0x6081	0x00	**P0602**	TX	rw	Profile position mode
0x6083	0x00	**P0603**	TX	rw	Profile position mode
0x6084	0x00	**P0604**	TX	rw	Profile position mode
0x6085	0x00	**P1213**	TX	rw	Profile position mode
0x6086	0x00	**P1190**	TX	rw	Profile position mode
0x6092	0x01	**P3050**	TX	rw	Factor group
0x6092	0x02	**P3051**	TX	rw	Factor group
0x6098	0x00	**P1205**	TX	rw	Homing mode
0x6099	0x01	**P1201**	TX / RX	rw	Homing mode

CANopen object number		Parameter No.	PDO mapping	Access type	Operating mode acc. to DSP 402
Index	Sub index				
0x6099	0x02	**P1202**	TX / RX	rw	Homing mode
0x609A	0x00	**P1203**	TX / RX	rw	Homing mode
0x60F8	0x00	**P1054**	TX / RX	rw	Profile velocity mode
0x60FB	0x01	**P0360**	TX	ro	Position control function
0x60FB	0x02	**P1050**	TX	rw	Position control function
0x60FB	0x03	**P1051**	TX	rw	Position control function
0x60FB	0x04	**P0364**	TX / RX	rw	Position control function
0x60FB	0x05	**P0363**	TX / RX	rw	Position control function
0x60FB	0x06	**P1053**	TX	rw	Position control function
0x60FB	0x07	**P0367**	TX	ro	Position control function
0x60FB	0x08	**P0362**	TX / RX	rw	Position control function
0x60FB	0x09	**P0392**	TX	ro	Position control function
0x60FB	0x0A	**P0391**	TX	ro	Position control function
0x60FB	0x0B	**P0365**	TX	ro	Position control function
0x60FB	0x0C	**P0460**	TX	ro	Position control function
0x60FB	0x0D	**P1191**	TX / RX	rw	Position control function
0x60FB	0x0E	**P1190**	TX	rw	Position control function
0x60FB	0x0F	**P1200**	TX / RX	rw	Position control function
0x60FB	0x10	**P1208**	TX / RX	rw	Position control function
0x60FB	0x11	**P0464**	TX	ro	Position control function
0x60FB	0x12	**P0605**	TX / RX	rw	Position control function
0x60FB	0x13	**P1198**	TX / RX	rw	Position control function
0x60FB	0x14	**P1199**	TX / RX	rw	Position control function
0x60FB	0x15	**P0601**	TX / RX	rw	Position control function
0x60FB	0x16	**P0611**	TX / RX	rw	Position control function
0x60FB	0x17	**P0370**	TX / RX	rw	Position control function
0x60FB	0x18	**P1209**	TX / RX	rw	Position control function
0x60FB	0x19	**P1204**	TX / RX	rw	Position control function
0x60FB	0x1A	**P0353**	TX	ro	Position control function
0x60FB	0x1B	**P0262** **P0263**	TX	ro	Position control function

CANopen object number		Parameter No.	PDO mapping	Access type	Operating mode acc. to DSP 402
Index	**Sub index**				
0x60FB	0x1C	**P0344**	TX	ro	Position control function
0x60FB	0x1D	**P1040**	TX/RX	rw	Position control functionn
0x60FD	0x00	**P0461**	TX	ro	Common entries
0x60FF	0x00	**P1179**	TX / RX	rw	Profile velocity mode
0x6510	0x01	**P0001**	TX	ro	Info
0x6510	0x02	**P0002**	TX	ro	Info
0x6510	0x03	**P0003**	TX	ro	Info
0x6510	0x04	**P0004**	TX	ro	Info
0x6510	0x05	**P0005**	TX	ro	Info
0x6510	0x06	**P0009**	TX	ro	Info
0x6510	0x07	**P0555**	TX	ro	Info
0x6510	0x08	**P0556**	TX	ro	Info

C

APPENDIX C - CONVERSION TABLES

This chapter contains tables specifying the conversion of CANopen communication objects into **b maXX 4000** controller communication parameters and vice versa. Conversion is performed by giving the value ranges ($x = x_{min} .. x_{max.}$) and the representation function x = f(x) (in the most simple case, the value is just passed through: y = x).

The tables contain the following entries:

CANopen object:	Identification of the CANopen object from DS402
Index ▸ P. No.:	Representation of the CANopen object indices on **b maXX 4000** controller parameter
Controller parameters:	Identification of the controller parameters
P. No. ▸ index:	Conversion of the **b maXX 4000** controller parameters to CANopen object indices

CANopen object	Index Value range	▸	P. no. Scaling		Controller parameters	P. no. Value range	▸	Index re-scaling	Comment
Controlword	0x6040	▸	**P0300**		Control word	**P0300**	▸	0x6040	Bit 6 in the control word is now supported; Bit 6 = 0 : Positioning mode „absolute“ Bit 6 = 1 : Positioning mode „relative, negative, positive“ No other position mode is supported via CoE and the control word.
	x = 0 .. 0xFFFF	▸	y = x			x = 0 .. 0xFFFF	▸	y = x	
Switch on	Bit 0	▸	unchanged		Switch on	Bit 0	▸	unchanged	
Disable voltage	Bit 1	▸	unchanged		Inhibit voltage	Bit 1	▸	unchanged	
Quick stop	Bit 2	▸	unchanged		Quickstop	Bit 2	▸	unchanged	
Enable operation	Bit 3	▸	unchanged		Operation enabled	Bit 3	▸	unchanged	
Operation mode specific	Bit 4	▸	unchanged		Depending on operation mode	Bit 4	▸	unchanged	
Operation mode specific	Bit 5	▸	unchanged		Depending on operation mode	Bit 5	▸	unchanged	
Operation mode specific	Bit 6	▸	unchanged		Depending on operation mode	Bit 6	▸	unchanged	
Reset fault	Bit 7	▸	unchanged		Reset error	Bit 7	▸	unchanged	
Operation mode specific	Bit 8	▸	unchanged		Depending on operation mode	Bit 8	▸	unchanged	
reserved	Bit 9	▸	unchanged		reserved (always 0)	Bit 9	▸	unchanged	
reserved	Bit 10	▸	unchanged		reserved (always 0)	Bit 10	▸	unchanged	
Manufacturer specific	Bit 11	▸	unchanged		Depending on operation mode	Bit 11	▸	unchanged	
Manufacturer specific	Bit 12	▸	unchanged		Depending on operation mode	Bit 12	▸	unchanged	
Manufacturer specific	Bit 13	▸	unchanged		Depending on operation mode	Bit 13	▸	unchanged	
Manufacturer specific	Bit 14	▸	unchanged		Depending on operation mode	Bit 14	▸	unchanged	
Manufacturer specific	Bit 15	▸	unchanged		Write protection	Bit 15	▸	unchanged	

CANopen object	Index Value range	►	P. no. Scaling		Controller parameters	P. no. Value range	►	Index re-scaling	Comment
Statusword	0x6041/ro				Status word	**P0301**	►	0x6041	In operation mode = 7: IP-Mode If the control word bit 4 is being set, bit 12 is set.
	x = 0 .. 0xFFFF					x = 0 .. 0xFFFF	►	y = x	
Ready to switch on					Ready-to-start	Bit 0	►	unchanged	
Switched on					Switched on	Bit 1	►	unchanged	
Operation enabled					Operation enabled	Bit 2	►	unchanged	
Fault					Error	Bit 3	►	unchanged	
Voltage disabled					Voltage disabled	Bit 4	►	unchanged	
Quick stop					Quickstop	Bit 5	►	unchanged	
Switched on enabled					Inhibit start	Bit 6	►	unchanged	
Warning					Warning	Bit 7	►	unchanged	Adjustable in ProDrive via drive manager
Manufacturer specific					Depending on operation mode	Bit 8	►	unchanged	
Remote					Remote	Bit 9	►	unchanged	
Targed reached					Set value reached	Bit 10	►	unchanged	
Internal limit active					Depending on operation mode	Bit 11	►	unchanged	Adjustable in ProDrive via drive manager
Operation mode specific					Depending on operation mode	Bit 12	►	unchanged	
Operation mode specific					Depending on operation mode	Bit 13	►	unchanged	
Manufacturer specific					Conf. status bits	Bit 14	►	unchanged	
Manufacturer specific					Conf. status bits	Bit 15	►	unchanged	

CANopen object	Index Value range	▸	P. no. Scaling		Controller parameters	P. no. Value range	▸	Index re-scaling	Comment
vl_target_velocity	0x6042	▸	**P1179**		RFG1Input32	**P1179**	▸	0x6042	The user-defined unit (speed units) is interpreted in the **b maXX 4000** controller as rpm. In case of changes in „Option module G/H configuration 1" bit 2 = 1: Internal set value for the speed in 1/10 rpm, same unit as object 0x6042, e. g.: 200.0 revolutions per min ⇒ input 2000.
	x = -32768 .. 32767	▸	y = x * 0x40000000/ MotorMaxSpeed			x = -2^{31} .. 2^{31} -1	▸	y = x*Motor MaxSpeed/ 0x40000000	
vl_velocity_demand	0x6043 /ro				SpeedSetValue	**P0351**	▸	0x6043	
						x = -2^{31} .. 2^{31} -1	▸	y = x*Motor MaxSpeed/ 0x40000000	
vl_control_effort	0x6044 /ro				SpeedActValue	**P0353**	▸	0x6044	
						x = -2^{31} .. 2^{31} -1	▸	y = x*Motor MaxSpeed/ 0x40000000	
vl_manipulated_velocity	0x6045 /ro				SpeedSetValueTotal	**P0352**	▸	0x6045	
						x = -2^{31} .. 2^{31} -1	▸	y = x*Motor MaxSpeed/ 0x40000000	
vl_velocity_min_max_ amount	0x6046								
vl_velocity_min_amount	SIX. 0x01		-		-	-	▸	SIX 0x01	SIX. 1 is always zero, the minimum limit is set at zero.
						x = 0	▸	y = x	
vl_velocity_max_amount	SIX. 0x02	▸	**P1042** / **P1041**		SpeedSet_LLim / Speed-Set_ULim	**P1042**/ **P1041**	▸	SIX 0x02	The maximum limit has a symmetrical effect on both directions of rotation in the **b maXX 4000**. The user-defined unit (speed units) is interpreted in the **b maXX 4000** cotroller as rpm
	x = 0 .. 0xFFFFFFFF	▸	y = x * 0x40000000/ MotorMaxSpeed		**P1041**: x = 0 .. 0x40000000 **P1042**: x = 0xC0000000 .. 0		▸	y = x*Motor MaxSpeed/ 0x40000000	
vl_pole_number	0x604D	▸	**P0065**		MotorPolePairs	**P0065**	▸	0x604D	
	x = 0 .. 255	▸	y = x / 2			x = 1..120	▸	y = x*2	
vl_ramp_function_time	0x604F	▸	**P1172**		RFG1RampUpTime	**P1172**	▸	0x604F	Ramp function generator ramp function time (1 = 1/1000 s ⇒ 1s = 1000). The resolution is 10 ms
	x = 0 .. 0xFFFFFFFF	▸				x = 0 .. 65000	▸		
vl_slow_down_time	0x6050	▸	**P1173**		RFG1RampDownTime	**P1173**	▸	0x6050	
	x = 0 .. 0xFFFFFFFF	▸				x = 0..65000	▸		
vl_quick_stop_time	0x6051	▸	**P1174**		RFG1StopTime	**P1174**	▸	0x6051	
	x = 0 .. 0xFFFFFFFF	▸				x = 0..65000	▸	y = x	

CANopen object	Index Value range	▸	P. no. Scaling		Controller parameters	P. no. Value range	▸	Index re-scaling	Comment
quick_stop_option_code	0x605A	▸	**P1004**		QuickstopCode (Schnellhalt)	**P1004**	▸	0x605A	
Conversion formalism	x = -32768 .. 32767	▸	y = x			x = 0 .. 3	▸	y = x	
Manufacturer specific	x = -32768 .. -1	▸	y = x		not used	x = -32768 .. -1			
Disable drive	x = 0	▸	y = x		Drive inhibited	x = 0	▸	y = x	
Slow down on slow down ramp	x = 1	▸	y = x		Ramp down on deceleration ramp	x = 1	▸	y = x	
Slow down on quickstop ramp	x = 2	▸	y = x		Ramp down on quickstop ramp	x = 2	▸	y = x	
Slow down on current ramp	x = 3	▸	y = x		Ramp down on current limit	x = 3	▸	y = x	
Slow down on voltage limit	x = 4	▸	y = x		Ramp down on voltage limit			y = 4	
Slow down on slow down ramp and stay in quickstop	x = 5	▸	y = x		Ramp down on deceleration ramp and stay in quickstop			y = 5	
Slow down on quickstop ramp and stay in quickstop	x = 6	▸	y = x		Ramp down on quickstop ramp and stay in quickstop			y = 6	
Slow down on current ramp and stay in quickstop	x = 7	▸	y = x		Ramp down on current limit and stay in quickstop.			y = 7	
Slow down on voltage limit and stay in quickstop	x = 8	▸	y = x		Ramp down on voltage limit and stay in quickstop			y = 8	
reserved	x = 9 .. 32767				not used			y = 9 ..32767	
shutdown_option_code	0x605B	▸	**P1005**		ShutDownCode	**P1005**	▸	0x605B	
Manufacturer specific	x = -32768 .. -3	▸	y = x		not used	x = -32768..-3			
Manufacturer specific	x = -2	▸	y = 3		Ramp down at current limit	x = 3	▸	y = -2	
Manufacturer specific	x = -1	▸	y = 2		Ramp down at quickstop ramp	x = 2	▸	y = -1	
Disable drive	x = 0	▸	y = x		Drive inhibited	x = 0	▸	y = x	
Slow down on slow down ramp	x = 1	▸	y = x		Ramp down on deceleration ramp	x = 1	▸	y = x	The selected ramp function generators are adjustable via **P1174** ramp function generator hold time or in 0x6051.
reserved	x = 2 .. 32767				not used			y = .. 32767	

CANopen object	Index Value range	▸	P. no. Scaling		Controller parameters	P. no. Value range	▸	Index re-scaling	Comment
disable_operation_ option_code	0x605C	▸	**P1006**		DisableOpCode	**P1006**	▸	0x605C	
Manufacturer specific	x = -32768 .. -3	▸	y = x		not used	x = -32768..-3			
Manufacturer specific	x = -2	▸	y = 3		not used			y = -2	
Manufacturer specific	x = -1	▸	y = 2		not used			y = -1	
Disable drive	x = 0	▸	y = x		Drive inhibited		▸	y = 0	
Slow down ...	x = 1	▸	y = x		Ramp down on deceleration ramp	x = 1	▸	y = x	
reserved	x = 2				Ramp down on quickstop ramp	x = 2	▸	y = -1	
reserved	x = 3				Ramp down on current limit	x = 3	▸	y = -2	
reserved	x = 4 .. 32767				not used			y = 4..32767	
stop_option_code	0x605D	▸	**P1003**		StopOptionCode (stop)	**P1003**	▸	0x605D	
Conversion formalism	x = -32768 .. 32767	▸	y = x			x = 0 .. 3	▸	y = x	
Manufacturer specific	x = -32768 .. -1	▸	y = x		not used				
Disable drive	x = 0	▸	y = x		Drive inhibited	x = 0	▸	y = x	
Slow down on slow down ramp	x = 1	▸	y = x		Ramp down on deceleration ramp	x = 1	▸	y = x	The selected ramp function generators are adjustable via **P1174** ramp function generator hold time or in 0x6051.
Slow down on quick Stop Ramp	x = 2	▸	y = x		Ramp down on quickstop ramp	x = 2	▸	y = x	
Slow Down On Current Ramp	x = 3	▸	y = x		Ramp down on current limit	x = 3	▸	y = x	
Slow Down On Voltage Limit	x = 4	▸	y = x		not used			y = 4	
reserved	x = 5 .. 32767				not used			y = 5..32767	

CANopen object	Index Value range	►	P. no. Scaling		Controller parameters	P. no. Value range	►	Index re-scaling	Comment
fault_reaction_option_ code	0x605E	►	**P1007**		ErrorReactionCode	**P1007**	►	0x605E	Presently for statical and dynamical position deviations adjustable
Conversion formalism	x = -32768 .. 32767	►	y = x			x = 0 .. 3	►	y = x	
Manufacturer specific	x = -32768 .. -1	►	y = x		not used	x = -32768 .. -1			
Disable Drive, motor is free to rotate	x = 0	►	y = x		Drive inhibited	x = 0	►	y = x	
Slow down on slow down ramp	x = 1	►	y = x		Ramp down at deceleration ramp	x = 1	►	y = x	
Slow down on quick stop ramp	x = 2	►	y = x		Ramp down on quickstop ramp	x = 2	►	y = x	
Slow down on current ramp	x = 3	►	y = x		Ramp down at current limit	x = 3	►	y = x	
Slow down on voltage limit	x = 4	►	y = x		not used			y = 4	
reserved	x = 5 .. 32767				not used			y = 5 .. 32767	

CANopen object	Index Value range	▸	P. no. Scaling		Controller parameters	P. no. Value range	▸	Index re-scaling	Comment
modes_of_operation	0x6060	▸	**P1000**		OperationModeSet	**P1000**	▸	0x6060	
Conversion formalism	x = -128 .. 127	▸	y = x			x = -128 .. 127	▸	y = x	
Manufacturer specific	x = -106		y' = -6						y' only internal in the controller
Manufacturer specific	x = -7	▸	y = x		Autotuning	x = -7	▸	y = x	
Manufacturer specific	x = -6	▸	y = 5		Spindle positioning	x = -6	▸	x = -106	
Manufacturer specific	x = -5	▸	y = x		Synchronous operation with electronic gearing	x = -5	▸	y = x	
Manufacturer specific	x = -4	▸	y = x		Position control or Interpolated position mode	x = -4	▸	y = -4 or y = -7	Depending on the adjusted mode: mode = -4 position control mode = 7 IP mode (internally the controller is switched to mode = -4)
Manufacturer specific	x = -3	▸	y = x		Speed control	x = -3	▸	y = 3	
Manufacturer specific	x = -2	▸	y = x		Current control	x = -2	▸	y = x	
Manufacturer specific	x = -1	▸	y = x		Find notch position	x = -1	▸	y = x	
reserved	x = 0				not used	x = 0			
Profile position mode	x = 1	▸	y = x		Target position set value	x = 1	▸	y = x	
Velocity mode	x = 2	▸	y = x		Speed setting 1	x = 2	▸	y = x	
Profile velocity mode	x = 3	▸	y = -3		not used	x = 3	▸	y = -3	
Torque profile mode	x = 4	▸	y = x		not used	x = 4			
reserved	x = 5				Jog operation	x = 5	▸	y = -6	
Homing mode	x = 6	▸	y = x		Homing operation	x = 6	▸	y = x	
Interpolated position mode	x = 7	▸	y = -4		not used	x = 7			
reserved	x = 8 .. 127				not used	x = 8 .. 127			

CANopen object	Index Value range	▸	P. no. Scaling		Controller parameters	P. no. Value range	▸	Index re-scaling	Comment
modes_of_operation_ display	0x6061/ro				OperationModeAct (actual operating mode)	**P0304**	▸	0x6061	The CANopen standard designation, see 0x6060 ○ Jog operation is set in the controller at value 5, in CANopen it is set at value -6 ○ Spindle positioning is set in the controller at value -6, in CANopen it is set at value -106
Conversion formalism						x = -128 .. 127	▸	y = x	
					Autotuning	x = -7	▸	y = x	
					Spindle positioning	x = -6	▸	y = -106	
					Synchronous operation el. gear	x = -5	▸	y = x	
					Position control or Interpolated Position Mode	x = -4	▸	y = x	
					Speed control	x = -3	▸	y = 3	
					Current control	x = -2	▸	y = x	
					Notch position	x = -1	▸	y = x	
					not used	x = 0			
					Target position set value	x = 1	▸	y = x	
					Speed setting 1	x = 2	▸	y = x	
					not used	x = 3		y = -3	
					not used	x = 4			
					Jog operation	x = 5	▸	y = -6	
					Homing operation	x = 6	▸	y = x	
					not used	x = 7			
					not used	x = 8 .. 127			
position_demand_value	0x6062/ro				PPosSetValue (actual position value)	**P0463**	▸	0x6062	UINT32 is provided on the option card with an offset of 2^{31} (UINT32 ⇒ INT32). Only in case of changes in option module G/H configuration 1 bit 5 = 1, no offset of -2^{31}.
						x = 0 .. 2^{32} - 1	▸	y = x - 2^{31}	
position_actual_value*	0x6063 /ro				PosActValue (actual position value)	**P0462**	▸	0x6063	
						x = 0 .. 2^{32} - 1	▸	y = x	
position_actual_value	0x6064 /ro				PPosActValue (actual position value)	**P0462**	▸	0x6064	UINT32 is provided on the option card with an offset of 2^{31} (UINT32 ⇒ INT32). Only in case of changes in option module G/H configuration 1 bit 5 = 1, no offset of -2^{31}.
						x = 0 .. 2^{32} - 1	▸	y = x - 2^{31}	

CANopen object	Index Value range	▸	P. no. Scaling		Controller parameters	P. no. Value range	▸	Index re-scaling	Comment
following_error_time_out	0x6066/ro				PosDevTime (pos. error time)	**P1056**	▸	0x6066	In the CANopen object and in the **b maXX 4000** controller parameter the unit is ms.
						x = 0 .. 65000	▸	y = x	
position_window	0x6067	▸	**P1194**		PPosWindow (pos. window)	**P1194**	▸	0x6067	
	x = 0 .. 0xFFFFFFFF	▸	y = x			x = 0 .. 0xFFFFFFFF	▸	y = x	
position_window_time	0x6068	▸	**P1195**		PPosWindow Time (pos. window time)	**P1195**	▸	0x6068	
	x = 0 .. 65535	▸	y = x			x = 1 .. 0xFFFF	▸	y = x	
velocity_sensor_actual_value	0x6069 /ro				Enc1ActAngle	**P0391**	▸	0x6069	
						x = 0 .. 0xFFFFFFFF	▸	y = x	
sensor_selection_code	0x606A /ro				-	-			The **b maXX 4000** controller only supports position encoders, thus display only.
velocity_actual_value_-from_position_encoder						x = 0	▸	y = x	
velocity_actual_value_-from_velocity_encoder					not supported				
velocity_demand_value	0x606B /ro				SpeedSetValueTotal	**P0352**	▸	0x606B	The user-defined unit (speed units) is interpreted in the controller as rpm. Only in case if changes in option module G/H configuration 1 bit 2 = 1: Indication of the current speed in 1/10 rpm, e. g.: 200.0 revolutions ⇒ input 2000.
						x = -2^{31} .. 2^{31} -1	▸	y = x*Motor-MaxSpeed / 0x40000000	
velocity_actual_value	0x606C /ro				SpeedActValue	**P0353**	▸	0x606C	Only in case of changes in option module G/H configuration 1 bit 2 = 1: Indication of the current speed in 1/10 rpm. e.g.: 200.0 revolutions ⇒ input 2000.
						x = -2^{31} .. 2^{31} -1	▸	y = x*Motor-MaxSpeed / 0x40000000	
velocity_threshold	0x606F	▸	**P1073**		Enc1Mon_LLim	**P1073**	▸	0x606F	The threshold can be raised in the **b maXX 4000** controller up to 25 % of the maximum speed of the controller. The input is then effected in rpm e.g. max: 1000 rpm input for 25 % = 250 rpm
	x = -2^{31} .. 2^{31}-1	▸	y = x*0x4000 / 10000			x = -0 .. 0x1000	▸	y = x*10000 / 0x4000	

CANopen object	Index Value range	▸	P. no. Scaling		Controller parameters	P. no. Value range	▸	Index re-scaling	Comment
target_torque	0x6071	▸	**P0331**		TrqSetValue	**P0331**	▸	0x6071	1000 corresponds to 100.0 % related to the rated torque P1036
	$x = -2^{15} ... 2^{15} -1$	▸	y = x*0x4000 / 1000			$x = -2^{15} .. 2^{15} -1$	▸	y = x*1000 / 0x4000	
max_torque	0x6072	▸	**P0357**		TrqSynDirect	**P0357**	▸	0x6072	1000 corresponds to 100.0% related to the rated torque P1036
	$x = 0 ... 2^{16} - 1$	▸	y = x*0x4000 / 1000			$x = 0 ... 2^{16} - 1$	▸	y = x*1000 / 0x4000	
torque_actual_value	0x6077 /ro					**P0508**	▸	0x6077	1000 corresponds to 100.0% related to the rated torque P1036
						$x = -2^{31} .. 2^{31} -1$	▸	y = x	
target_position	0x607A	▸	**P0369 / P0607**		PosIpSetValue / PPosRelTarget0	**P0369 / P0607**	▸	0x607A	UINT32 is provided on the option card with an offset of 2^{31} (UINT32 ⇒ INT32). Only in case of changes in option module G/H configuration 1 bit 5 = 1, no offset of 2^{31}.
	$x = -2^{31} .. 2^{31}-1$	▸	$y = x + 2^{31}$			$x = 0 ... 2^{32} - 1$	▸	$y = x - 2^{31}$	
home_offset	0x607C	▸	**P1200**		PPosEncoderOffset	**P1200**	▸	0x607C	Deviation of the home position from the reference switch respectively limit switch UU transmission and an offset of 2^{31} is added. Only in case of changes in option module G/H configuration 1 bit 2 = 1: Indication in UU e. g. 1/100° for rotations. No limit value monitoring.
	$x = -2^{31} .. 2^{31}-1$	▸	$y = x + 2^{31}$			$x = 0 ... 2^{32} - 1$	▸	$y = x - 2^{31}$	
software_position_limit	0x607D				SW limit switch			0x607D	UINT32 is provided on the option card with an offset of 2^{31} (UINT32 ⇒ INT32). Only in case of changes in option module G/H configuration 1 bit 5 = 1, no offset of 2^{31}.
	SIX 0x01	▸	**P1196**		PPosSWLimitSwitch1	**P1196**	▸	SIX 0x01	
	$x = -2^{31} .. 2^{31}-1$	▸	$y = x + 2^{31}$			$x = 0 ... 2^{32} - 1$	▸	$y = x - 2^{31}$	
	SIX. 0x02	▸	**P1197**		PPosSWLimitSwitch2	**P1197**	▸	SIX 0x02	
	$x = -2^{31} .. 2^{31}-1$	▸	$y = x + 2^{31}$			$x = 0 ... 2^{32} - 1$	▸	$y = x - 2^{31}$	
max_profile_velocity	0x607F	▸	**P0057**		MotorNomSpeed	**P0057**	▸	0x607F	The user-defined unit (speed units) is interpreted in the controller as rpm
	$x = 0 .. 2^{32}-1$	▸	y = x			x = 1 .. 24000	▸	y = x	
max_motor_speed	0x6080	▸	**P1031**		SpeedMax	**P1031**	▸	0x6080	The user-defined unit (speed units) is interpreted in the controller as rpm
	$x = 0 .. 2^{16}-1$	▸	y = x			x = 20 .. 24000	▸	y = x	

CANopen object	Index Value range	▸	P. no. Scaling		Controller parameters	P. no. Value range	▸	Index re-scaling	Comment
profile velocity	0x6081	▸	**P0602**		PPosSetSpeed1	**P0602**	▸	0x6081	Only in case of changes in option module G/H configuration 1 bit 2 = 1: Indication of the desired driving speed of the motion task in UU e.g. 1/100°/s for rotations. s ⇒ ms [1/1000]. In case limit values of **b maXX 4000** parameters has exceeded , the minimum and maximum values are set without transmitting an error message.
	x = 0 .. 2^{32}-1	▸	y = x			x = 1 .. 13200	▸	y = x	
profile acceleration	0x6083	▸	**P0603**		PPosAcceleraton0	**P0603**	▸	0x6083	Only in case of changes in option module G/H configuration 1 bit 2 = 1: starting/braking acceleration of the motion task indicated in UU e. g. 10°/s^2 for rotations. s^2 ⇒ ms^2 [1/1000000]. In case limit values of **b maXX 4000** parameters has exceeded , the minimum and maximum values are set without transmitting an error message.
	x = 0 .. 2^{32}-1	▸	y = x			x = 25 .. 45000	▸	y = x	
profile deceleration	0x6084	▸	**P0604**		PPosDeceleraton0	**P0604**	▸	0x6084	
	x = 0 .. 2^{32}-1	▸	y = x			x = 25 .. 45000	▸	y = x	
quick_stop_deceleration	0x6085	▸	**P1213**		PPosStopDeceleraton	**P1213**	▸	0x6085	
	x = 0 .. 2^{32}-1	▸	y = x			x = 25 .. 45000	▸	y = x	
motion profile type	0x6086	▸	**P1190**		PPosMode	**P1190**	▸	0x6086	
	x = -2^{15}... 2^{15} - 1					x = 0 .. 2^{16} - 1			
Manufacturer specific	x = -2^{15}... -1				not used				
Linear ramp (trapezoidal profile)	x = 0	▸	Bit 3 and Bit 4		Trapezium	Bit 3 and Bit 4	▸	y = 0	Velocity profile in the controller: bit 4 bit 3: 0 0: trapezoidal 1 0: Sin² 0 1: S-curve 1 1: reserved
Sin^2 ramp	x = 1	▸	Bit 3 and Bit 4		Sin^2	Bit 3 and Bit 4	▸	y = 1	
Jerk-free ramp	x = 2	▸	Bit 3 and Bit 4		S curve	Bit 3 and Bit 4	▸	y = 2	
Jerk-limited ramp	x = 3				not used				
For future profile type	x = 4 .. 2^{15} - 1				not used				
position_encoder_ resolution	0x608F							0x608F	Only used in case of changes in option module G/H configuration 1 bit 14 = 1.
encoder_increments	SIX 0x01	▸	**P2172**		PosResEncIncDS402	**P2172**	▸	SIX 0x01	
	x = 0 .. 2^{32} - 1	▸	y = x			x = 0 .. 2^{32} - 1	▸	y = x	
motor_revolutions	SIX 0x02	▸	**P2173**		PosResMotRevDS402	**P2173**	▸	SIX 0x02	
	x = 0 .. 2^{32} - 1	▸	y = x			x = 0 .. 2^{32} - 1	▸	y = x	

CANopen object	Index Value range	▸	P. no. Scaling		Controller parameters	P. no. Value range	▸	Index re-scaling	Comment
velocity_encoder_ resolution	0x6090							0x6090	Only used in case of changes in option module G/H configuration 1 bit 14 = 1.
encoder_increments_per_ second	SIX 0x01	▸	**P2175**		VelActValueDS402	**P2175**	▸	SIX 0x01	
	x = 0 .. 2^{32} - 1	▸	y = x			x = 0 .. 2^{32} - 1	▸	y = x	
motor_revolutions_per_ second	SIX 0x02	▸	**P2176**		VelResEncIncDS402	**P2176**	▸	SIX 0x02	
	x = 0 .. 2^{32} - 1	▸	y = x			x = 0 .. 2^{32} - 1	▸	y = x	
feed_constant	0x6092							0x6092	
feed	SIX 0x01	▸	**P3050**		PosScalingUserUnit	**P3050**	▸	SIX 0x01	
	x = 0 .. 2^{32} - 1	▸	y = x			x = 1 .. 2^{24} - 1	▸	y = x	
shaft_revolutions	SIX 0x02	▸	**P3051**		PosScalingRevolution	**P3051**	▸	SIX 0x02	
	x = 0 .. 2^{32} - 1	▸	y = x			x = 1 .. 2^{8} - 1	▸	y = x	

CANopen object	Index Value range	▸	P. no. Scaling		Controller parameters	P. no. Value range	▸	Index re-scaling	Comment
homing_method	0x6098	▸	**P1205**		PPosHomingMode (ref. homing mode)	**P1205**	▸	0x6098	
Manufacturer specific	x = -128 .. -12				not used			y = -128 ..-12	
Manufacturer specific	x = -10	▸	y = x		Reaching of mechanical stop with zero pulse, counter-clockwise	x = -10	▸	y = x	
Manufacturer specific	x = -9	▸	y = x		Reaching of mechanical stop with zero pulse, clockwise rotation	x = -9	▸	y = x	
Manufacturer specific	x = -8	▸	y = x		Reaching of mechanical stop, counter-clockwise	x = -8	▸	y = x	
Manufacturer specific	x = -7	▸	y = x		Reaching of mechanical stop, clockwise rotation	x = -7	▸	y = x	
Manufacturer specific	x = -6	▸	y = x		Reaching of the next encoder zero angle	x = -6	▸	y = x	
Manufacturer specific	x = -5	▸	y = x		Moving to pos. limit switch	x = -5	▸	y = x	
Manufacturer specific	x = -4	▸	y = x		Moving to neg. limit switch	x = -4	▸	y = x	
Manufacturer specific	x = -3	▸	y = x		Setting of home position	x = -3	▸	y = x	
Manufacturer specific	x = -2	▸	y = x		Reaching the encoder zero angle or zero pulse with counter-clockwise rotation	x = -2	▸	y = x	
Manufacturer specific	x = -1	▸	y = x		Reaching the encoder zero angle or zero pulse with clockwise rotation	x = -1	▸	y = x	
No homing operation	x = 0				not used			y = 0	
Homing on the neg. limit switch	x = 1	▸	y = x		Neg. limit switch with encoder zero angle or zero pulse reference	x = 1	▸	y = x	
Homing on the pos. limit switch	x = 2	▸	y = x		Pos. limit switch with encoder zero angle or zero pulse reference	x = 2	▸	y = x	
Homing on the positive home switch & index pulse	x = 3	▸	y = x		Pos. zero point switch with encoder zero angle or zero pulse reference, counter-clockwise rotation	x = 3	▸	y = x	

CANopen object	Index Value range	►	P. no. Scaling		Controller parameters	P. no. Value range	►	Index re-scaling	Comment
Homing on the positive Home Switch & Index Pulse	x = 4	►	y = x		Pos. zero point switch with encoder zero angle or zero pulse reference, clockwise rotation	x = 4	►	y = x	
Homing on the negative Home Switch & Index Pulse	x = 5	►	y = x		Neg. zero point switch with zero encoder angle or zero pulse reference, clockwise rotation	x = 5	►	y = x	
Homing on the negative Home Switch & Index Pulse	x = 6	►	y = x		Neg. zero point switch with encoder zero angle or zero pulse reference, counter-clockwise rotation	x = 6	►	y = x	
Zero reference cam switch, left to pos. edge with Zero pulse; CW move	x = 7	►	y = x		Zero point switch, to the left of pos. edge with zero pulse; clockwise direction	x = 7	►	y = x	
Zero reference cam switch, right of pos. edge with Zero pulse; CW move	x = 8	►	y = x		Zero point switch, to the right of pos. edge with zero pulse, clockwise rotation	x = 8	►	y = x	
Zero reference cam switch, left to neg. edge with Zero pulse; CW move	x = 9	►	y = x		Zero point switch, to the left of neg. edge with zero pulse, clockwise rotation	x = 9	►	y = x	
Zero reference cam switch, right to neg. edge with Zero pulse; CW move	x = 10	►	y = x		Zero point switch, to the right of neg. edge with zero pulse; clockwise rotation	x = 10	►	y = x	
Zero reference cam switch, right to neg. edge with Zero pulse; CCW move	x = 11	►	y = x		Zero point switch, on the right of neg. edge with zero pulse; counter-clockwise rotation	x = 11	►	y = x	
Zero reference cam switch, right of pos. edge with Zero pulse; CCW move	x = 12	►	y = x		Zero point switch, on the right of pos. edge with zero pulse; counter-clockwise rotation	x = 12	►	y = x	
Zero reference cam switch, left to neg. edge with Zero pulse; CCW move	x = 13	►	y = x		Zero point switch, on the right of pos. edge with zero pulse; counter-clockwise rotation	x = 13	►	y = x	
Zero reference cam switch, right to neg. edge with Zero pulse; CCW move	x = 14	►	y = x		Zero point switch, on the right of neg. edge with zero pulse; counter-clockwise rotation	x = 14	►	y = x	
reserved	x = 15, 16				not used				
Negative limit switch	x = 17	►	y = x		Negative limit switch	x = 17	►	y = x	
Positive limit switch	x = 18	►	y = x		Positive limit switch	x = 18	►	y = x	

CANopen object	Index Value range	▸	P. no. Scaling		Controller parameters	P. no. Value range	▸	Index re-scaling	Comment
Positive Zero reference switch, CCW move	x = 19	▸	y = x		Positive zero point switch; counter-clockwise rotation	x = 19	▸	y = x	
Positive Zero reference switch, CW move	x = 20	▸	y = x		Positive zero point switch; clockwise rotation	x = 20	▸	y = x	
Negative Zero reference switch, CW move	x = 21	▸	y = x		Negative zero point switch; clockwise rotation	x = 21	▸	y = x	
Negative Zero reference switch, CCW move	x = 22	▸	y = x		Negative zero point switch; counter-clockwise rotation	x = 22	▸	y = x	
Zero reference cam switch, left to pos. edge; CW move	x = 23	▸	y = x		Zero point switch, to the left of pos. edge; clockwise rotation	x = 23	▸	y = x	
Zero reference cam switch, right to pos. edge; CW move	x = 24	▸	y = x		Zero point switch, to the right of pos. edge; clockwise rotation	x = 24	▸	y = x	
Zero reference cam switch, left to neg. edge; CW move	x = 25	▸	y = x		Zero point switch, to the left of neg. edge; clockwise rotation	x = 25	▸	y = x	
Zero reference cam switch, right to neg. edge; CW move	x = 26	▸	y = x		Zero point switch, to the right of neg. edge; clockwise rotation	x = 26	▸	y = x	
Zero reference cam switch, right to neg. edge; CCW move	x = 27	▸	y = x		Zero point switch, on the right of neg. edge; counter-clockwise rotation	x = 27	▸	y = x	
Zero reference cam switch, left to neg. edge; CCW move	x = 28	▸	y = x		Zero point switch; on the left of neg. edge; counter-clockwise rotation	x = 28	▸	y = x	
Zero reference cam switch, right to pos. edge; CCW move	x = 29	▸	y = x		Zero point switch, on the right of pos. edge; counter-clockwise rotation	x = 29	▸	y = x	
Zero reference cam switch, left to pos. edge; CCW move	x = 30	▸	y = x		Zero point switch, on the left of pos. edge; counter-clockwise rotation	x = 30	▸	y = x	
reserved	31 .. 32				not used	31..32			
Nearest zero pulse; CCW move	x = 33	▸	y = x		Next zero pulse; counter-clockwise rotation	x = 33	▸	y = x	
Nearest Zero pulse; CW move	x = 34	▸	y = x		Next zero pulse with clockwise rotation	x = 34	▸	y = x	
Homing on the Current Postion	x = 35	▸	y = x		Setting of home position	x = 35	▸	y = x	
reserved	x = 36 .. 127				not used				

CANopen object	Index Value range	▸	P. no. Scaling		Controller parameters	P. no. Value range	▸	Index re-scaling	Comment
homing_speeds	0x6099				(Ref. speed.)			0x6099	Only in case of changes in option module G/H configuration 1 bit 2 = 1: Indication of the desired drive speed of the motion task in UU e. g. 1/100°/s for rotations. s ⇒ ms [1/1000]. In case limit values of **b maXX 4000** parameters has exceeded, the minimum and maximum values are set without transmitting an error message.
Speed_during_search_for_switch	SIX 0x01	▸	**P1201**		PPosHomingSpeed	**P1201**	▸	SIX 0x01	
	x = 0 .. 2^{32}-1	▸	y = x			x = 1 .. 13200	▸	y = x	
Speed_during_search_-for_zero	SIX 0x02	▸	**P1202**		PPosHomingFinalSpeed	**P1202**	▸	SIX 0x02	
	x = 0 .. 2^{32}-1	▸	y = x			x = 1 .. 50	▸	y = x	
homing_acceleration	0x609A	▸	**P1203**		PPosHomingAcceler (homing acceleration)	**P1203**	▸	0x609A	Only in case of changes in option module G/H configuration 1 Bit 2 = 1: The user-defined unit (acceleration units) of homing_acceleration in UU e. g. 10°/s^2 for rotations. s^2 ⇒ ms^2 [1/1000000]. In case limit values of **b maXX 4000** parameters has exceeded, the minimum and maximum values are set without transmitting an error message.
	x = 0 .. 2^{32}-1	▸	y = x			x = 25 .. 45000	▸	y = x	
max_slippage	0x60F8	▸	**P1054**		PosDevLimDyn	**P1054**	▸	0x60F8	
	x = 0 .. 2^{31} -1	▸	y = x			x = 0 .. 2^{31} -1	▸	y = x	
position_control_parameter_set	0x60FB								Manufacturer-specific CANopen object
Manufacturer specific	SIX 0x01 /ro				PosCtrlStatus	**P0360**	▸	SIX 0x01	Default = 0
						x = 0 .. 2^{16}-1	▸	y = x	
Manufacturer specific	SIX 0x02	▸	**P1050**		PosCtrlMode	**P1050**	▸	SIX 0x02	Default = 0
	x = 0 .. 2^{16}-1	▸	y = x			x = 0 .. 2^{16}-1	▸	y = x	
Manufacturer specific	SIX 0x03	▸	**P1051**		PosCtrl_Kv-Faktor	**P1051**	▸	SIX 0x03	Default = 10,0
	x = 0 .. 2^{15}-1	▸	y = x				▸	y = x	
Manufacturer specific	SIX 0x04	▸	**P0364**		PosSetRev	**P0364**	▸	SIX 0x04	Default = 0
	x = 0 .. 2^{16}-1	▸	y = x				▸	y = x	
Manufacturer specific	SIX 0x05	▸	**P0363**		PosSetAngle	**P0363**	▸	SIX 0x05	Default = 0
	x = 0 .. 2^{16}-1	▸	y = x				▸	y = x	

CANopen object	Index Value range	▸	P. no. Scaling		Controller parameters	P. no. Value range	▸	Index re-scaling	Comment
Manufacturer specific	SIX 0x06	▸	**P1053**		SpeedFeedForFactor	**P1053**	▸	SIX 0x06	Default = 0x4000
	x = 0 .. 2^{16}-1	▸	y = 0 .. 0x5000			x = 0 .. 0x5000	▸	y = 0 .. 2^{16}-1	
Manufacturer specific	SIX 0x07 /ro				PosCtrlDev	**P0367**	▸	SIX 0x07	Default = 0
						x = -2^{31} .. 2^{31}-1	▸	y = x	
Manufacturer specific	SIX 0x08		**P0362**		PosActValue	**P0362**	▸	SIX 0x08	Default = 0
	x = 0 .. 2^{32}-1		y = x			x = 0 .. 2^{32}-1	▸	y = x	
Manufacturer specific	SIX 0x09 /ro				Enc1ActRev	**P0392**	▸	SIX 0x09	Default = 0
						x = 0 .. 2^{32} - 1	▸	y = x	
Manufacturer specific	SIX 0x0A /ro				Enc1ActAngle	**P0391**	▸	SIX 0x0A	Default = 0
						x = 0 .. 2^{32} - 1	▸	y = x	
Manufacturer specific	SIX 0x0B /ro				SpeedFeedFor	**P0365**	▸	SIX 0x0B	Default = 0
						x = -2^{31} .. 2^{31}-1	▸	y = x	
Manufacturer specific	SIX 0x0C /ro				PPosStatus	**P0460**	▸	SIX 0x0C	Default = 0
						x = 0 .. 2^{16} - 1	▸	y = x	
Manufacturer specific	SIX 0x0D	▸	**P1191**		PPosActRecordNumber	**P1191**	▸	SIX 0x0D	Default = 0
	x = 0 .. 2^{16}-1	▸	y = x			x = 0 .. 2^{16} - 1	▸	y = x	
Manufacturer specific	SIX 0x0E	▸	**P1190**		PPosMode	**P1190**	▸	SIX 0x0E	Default = 0
	x = 0 .. 2^{16}-1	▸	y = x			x = 0 .. 2^{16} - 1	▸	y = x	
Manufacturer specific	SIX 0x0F	▸	**P1200**		PPosHomePosition	**P1200**	▸	SIX 0x0F	Default = 0
	x = 0 .. 2^{32}-1	▸	y = x			x = 0 .. 2^{32} - 1	▸	y = x	
Manufacturer specific	SIX 0x10	▸	**P1208**		PPosSwitchMode	**P1208**	▸	SIX 0x10	Default = 0
	x = 0 .. 2^{16}-1	▸	y = x			x = 0 .. 2^{16} - 1	▸	y = x	
Manufacturer specific	SIX 0x11 /ro				PPosSpeedSetValue	**P0464**	▸	SIX 0x11	Default = 0
						x = -2^{15} .. 2^{15}-1	▸	y = x	
Manufacturer specific	SIX 0x12	▸	**P0605**		PPosBend0	**P0605**	▸	SIX 0x12	Default = 0
	x = 0 .. 2^{16}-1	▸	y = x			x = 0 .. 8191	▸	y = x	
Manufacturer specific	SIX 0x13	▸	**P1198**		PPosClipEnvironment1	**P1198**	▸	SIX 0x13	Default = 0
	x = 0 .. 2^{32}-1	▸	y = x			x = 0.. 2^{32} - 1	▸	y = x	
Manufacturer specific	SIX 0x14	▸	**P1199**		PPosClipEnvironment2	**P1199**	▸	SIX 0x14	Default = 0
	x = 0 .. 2^{32}-1	▸	y = x			x = 0 .. 2^{32} - 1	▸	y = x	
Manufacturer specific	SIX 0x15	▸	**P0601**		PPosTargetInput0	**P0601**	▸	SIX 0x15	Default = 0
	x = -2^{15} .. 2^{15}-1	▸	y = x			x = -2^{15} .. 2^{15}-1	▸	y = x	

CANopen object	Index Value range	▸	P. no. Scaling		Controller parameters	P. no. Value range	▸	Index re-scaling	Comment
Manufacturer specific	SIX 0x16	▸	**P0611**		PposTargetInput1	**P0611**	▸	SIX 0x16	Default = 0
	$x = -2^{15} .. 2^{15}-1$	▸	y = x			$-2^{15} .. 2^{15}-1$	▸	y = x	
Manufacturer specific	SIX 0x17	▸	**P0370**		PosIpSetAngle	**P0370**	▸	SIX 0x17	Default = 0
	$x = 0 .. 2^{32}-1$	▸	y = x			$x = 0 .. 2^{32}-1$	▸	y = x	
Manufacturer specific	SIX 0x18	▸	**P1209**		PPosEncoderOffset	**P1209**	▸	SIX 0x18	Default = 0
	$x = 0 .. 2^{16}-1$	▸	y = x			$x = 0 .. 2^{16}-1$	▸	y = x	
Manufacturer specific	SIX 0x19	▸	**P1204**		PPosHomingDeceler	**P1204**	▸	SIX 0x19	Default = 0
	$x = 0 .. 2^{16}-1$	▸	y = x			$x = 0 .. 2^{16}-1$	▸	y = x	
Manufacturer specific	SIX 0x1A / ro				SpeedActValue	**P0353**	▸	SIX 0x1A	The actual speed value (**P0353**) is being altered from a 32 bit value to 16384. 100 % of the maximum speed (in **P1031**) corresponds to 16384 units. The amount is edited.
							▸	y = x	
Manufacturer specific	SIX 0x1B / ro				AmpWarning/MotorWarning	**P0262**, **P0263**	▸	SIX 0x1B	Bit 0 **P0263** bit 1 motor temperature has exceeded threshold 2 Bit 1 **P0263** bit 1 motor temperature has exceeded threshold 2 Bit 2 **P0262** bit 1 power unit temperature > 80°C Bit 3 not assigned Bit 4 **P0263** bit 0 motor temperature has exceeded threshold 1 Bit 5 **P0263** bit 0 motor temperature has exceeded threshold 1
							▸	y = x	
Manufacturer specific	SIX 0x1C / ro				TorqueAct	**P0344**		SIX 0x1C	Default = 0
						$x = -2^{15} .. 2^{15}-1$		y = x	
Manufacturer specific	SIX 0x1D		**P1040**		SpeedSetValueAdd	**P1040**		SIX 0x1D	The user-defined unit (velocity units) is interpreted in the b maXX® controller as rpm. Only in case of changes in Options-modul G/H-Konfiguration 1 bit 2 = 1: Indication of the current speed in 1/10 rpm. e.g.: 200.0 revolutions ⇒ input 2000.
	$x = -2^{31} .. 2^{31}-1$	▸	y = x * 0x40000000 / MotorMaxSpeed			$x = -2^{31} .. 2^{31} -1$	▸	y = x * MotorMaxSpeed / 0x4000000	

CANopen object	Index Value range	▸	P. no. Scaling		Controller parameters	P. no. Value range	▸	Index re-scaling	Comment
digital_inputs	0x60FD /ro				DigInOutStatus	**P0461**	▸	0x60FD	
						x = 0 .. 2^{16} - 1	▸	y = 0 .. 2^{32}-1	
Negative limit switch					Status limit switch neg.	Bit 0	▸	Bit 0	
Positive limit switch					Status limit switch pos.	Bit 1	▸	Bit 1	
Home switch					Status zero point switch	Bit 2	▸	Bit 2	
Interlock					reservert	Bit 4			
reserved					reserved	Bit 3 .. 15			
Manufacturer specific					not used			Bit 16 .. 31	
target_velocity	0x60FF	▸	**P1179**		RFG1Input32	**P1179**	▸	0x60FF	The user-defined unit (velocity units) is interpreted in the b maXX® controller as rpm. Only in case of changes in Options-modul G/H-Konfiguration 1 bit 2 = 1: Indication of the current speed in 1/10 rpm. e.g.: 200.0 revolutions ⇒ input 2000.
	x = -2^{31} .. 2^{31}-1	▸	y = x * 0x40000000 / MotorMaxSpeed			x = -2^{31} .. 2^{31} -1	▸	y = x * Motor-MaxSpeed / 0x4000000	
drive_data	0x6510							0x6510	
Manufacturer specific	SIX 0x01/ ro				ControllerType	**P0001**	▸	SIX 0x01	
							▸	y = x	
Manufacturer specific	SIX 0x02 / ro				SoftwareType	**P0002**	▸	SIX 0x02	
							▸	y = x	
Manufacturer specific	SIX 0x03 / ro				SoftwareID	**P0003**	▸	SIX 0x03	
							▸	y = x	
Manufacturer specific	SIX 0x04 / ro				SoftwareVersion	**P0004**	▸	SIX 0x04	
							▸	y = x	
Manufacturer specific	SIX 0x05 / ro				ParamTableVersion	**P0005**	▸	SIX 0x05	
							▸	y = x	
Manufacturer specific	SIX 0x06 / ro				AmpSW_Version	**P0009**	▸	SIX 0x06	
							▸	y = x	
Manufacturer specific	SIX 0x07 / ro				FbgaVersion	**P0555**	▸	SIX 0x07	
							▸	y = x	
Manufacturer specific	SIX 0x08 / ro				BootloaderVersion	**P0556**		SIX 0x08	
								y = x	

D

APPENDIX D - TECHNICAL DATA: POWERLINK CONTROLLED NODE

In this appendix you will find a survey of the technical data of the **POWERLINK Controlled Node**.

D.1 Technical features

CPU	Nios® II
FPGA	Cyclone IV (Fa. Altera)
Memory	8 kByte DPRAM, 64 MByte DDR2 SDRAM, 8 MByte Flash-Eprom
Baud rate	100 Mbit
Operating voltage	+5 V internal
Plug-in connector	2 RJ45 sockets, 8-pin

D.2 Data channels to the b maXX 4000 controller

For the data transmission of **b maXX 4000** controller to the option module **POWERLINK Controlled Node** there are three channels:

- Two process data channels (1 PDO per communication direction)
- One service data channel (server SDO)

With PDOs objects can be transferred in cyclic data exchange. Not all objects are available for PDO transfer.

With the SDO transfer all **b maXX 4000** parameters can be accessed via the object index (exception string parameter).

Table of figures

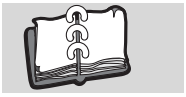


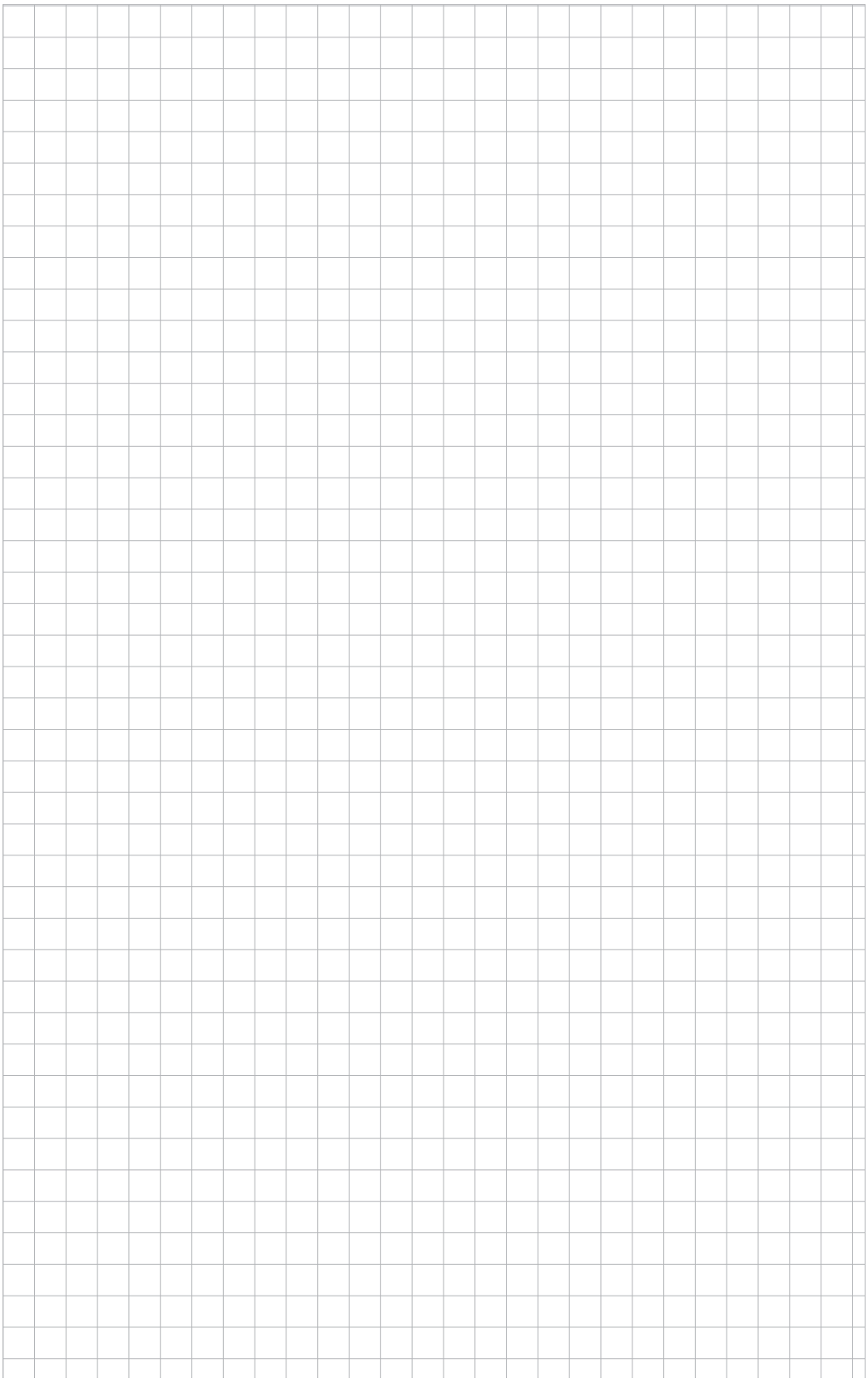
Table of figures

Index

Revision survey

Version	Status	Changes
5.13013.01	06.06.2013	First edition
5.13013.02	09.09.2013	Error correction
5.13013.03	13.08.2014	Addendum Factor Group
5.13013.04	13.01.2017	Addendum Gear factor Ch. 4.3.5

Notes:



be in motion

Baumüller Nürnberg GmbH Ostendstraße 80-90 90482 Nürnberg T: +49(0)911-5432-0 F: +49(0)911-5432-130 **www.baumueller.de**

All information given in this manual is customer information, subject to change without notice. We reserve the right to futher develop and actualize
our products continuously using our permanent revision service. Please notice, that specifications/data/information are current values according to the printing date.
These statements are not legally binding according to the measurement, computation and calculations. Before you make any information given in this manual to the basis
of your own calculations and/or applications, please make sure that you have the latest edition of the information in hand.
No liability can be accepted concerning the correctness of the information.