maxon motor control	EPOS3 EtherCAT Positioning Controllers
Application Notes	Edition May 2016

EPOS3 EtherCAT

Positioning Controllers

Application Notes

epos.maxonmotor.com

Document ID: rel5440

maxon motor ag Brünigstrasse 220 P.O.Box 263 CH-6072 Sachseln Phone +41 41 666 15 00 Fax +41 41 666 16 50 www.maxonmotor.com

PLEASE READ THIS FIRST

The present document represents a compilation of (hopefully) helpful “Good-to-Knows” that might come in handy in your daily work with EPOS3 EtherCAT Positioning Controllers.

The individual chapters cover particular cases or scenarios and are intended to give you a hand for efficient setup and parameterization of your system.

We strongly stress the following facts:

- *The present document does not replace any other documentation covering the basic installation and/or parameterization described therein!*
- *Also, any aspect in regard to health and safety, as well as to secure and safe operation are not covered in the present document – it is intended and must be understood as complimenting addition to those documents!*

TABLE OF CONTENTS

1 About this Document

1.1 Intended Purpose

The purpose of the present document is to provide you specific information to cover particular cases or scenarios that might come in handy during commissioning of your drive system.

Use for other and/or additional purposes is not permitted. maxon motor, the manufacturer of the equipment described, does not assume any liability for loss or damage that may arise from any other and/or additional use than the intended purpose.

The present document is part of a documentation set. Please find below an overview on the documentation hierarchy and the interrelationship of its individual parts:

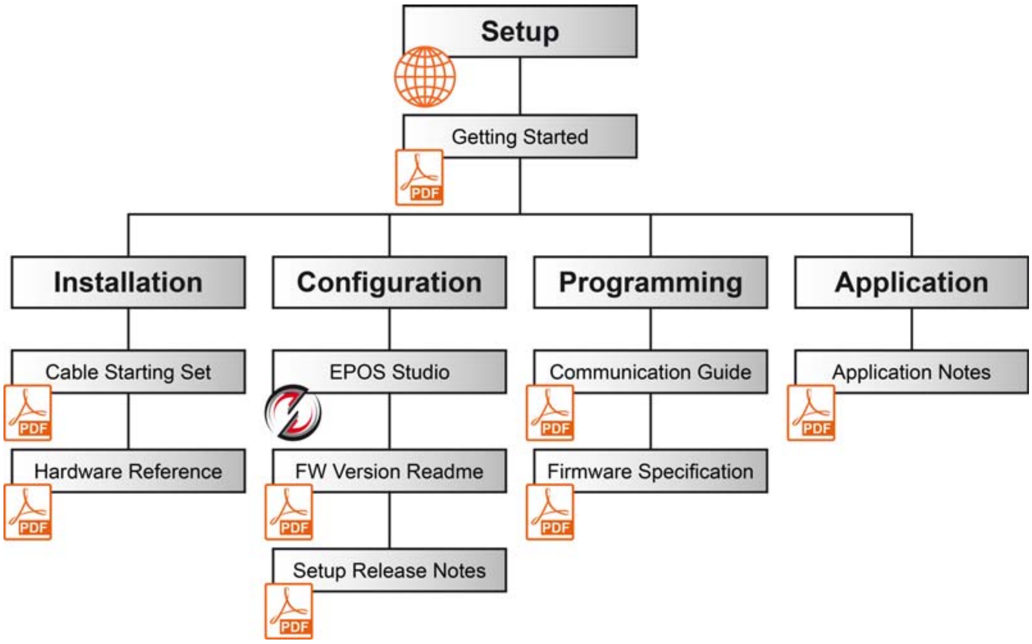


Figure 1-1 Documentation Structure

1.2 Target Audience

This document is meant for trained and skilled personnel working with the equipment described. It conveys information on how to understand and fulfill the respective work and duties.

This document is a reference book. It does require particular knowledge and expertise specific to the equipment described.

1.3 How to use

Take note of the following notations and codes which will be used throughout the document.

Notation	Explanation
«Abcd»	indicating a title or a name (such as of document, product, mode, etc.)
¤Abcd¤	indicating an action to be performed using a software control element (such as folder, menu, drop-down menu, button, check box, etc.) or a hardware element (such as switch, DIP switch, etc.)
(n)	referring to an item (such as order number, list item, etc.)
➔	denotes “see”, “see also”, “take note of” or “go to”

Table 1-1 Notations used in this Document

1.4 Symbols and Signs

1.4.1 Safety Alerts

Take note of when and why the alerts will be used and what the consequences are if you should fail to observe them!

Safety alerts are composed of…

- a signal word,
- a description of type and/or source of the danger,
- the consequence if the alert is being ignored, and
- explanations on how to avoid the hazard.

Following types will be used:

1) **DANGER**
Indicates an **imminently hazardous situation**. If not avoided, the situation **will** result in death or serious injury.
2) **WARNING**
Indicates a **potentially hazardous situation**. If not avoided, the situation **can** result in death or serious injury.
3) **CAUTION**
Indicates a **probable hazardous situation** and is also used to alert against unsafe practices. If not avoided, the situation **may** result in minor or moderate injury.

Example:

DANGER

High Voltage and/or Electrical Shock
Touching live wires causes death or serious injuries!

- *Make sure that neither end of cable is connected to live power!*
- *Make sure that power source cannot be engaged while work is in process!*
- *Obey lock-out/tag-out procedures!*
- *Make sure to securely lock any power engaging equipment against unintentional engagement and tag with your name!*

1.4.2 Prohibited Actions and Mandatory Actions

The signs define prohibitive actions. So, you **must not**!

Examples:

Do not touch!

Do not operate!

The signs point out actions to avoid a hazard. So, you **must**!

Examples:

Unplug!

Tag before work!

1.4.3 Informatory Signs

Requirement / Note / Remark
Indicates an action you must perform prior continuing or refers to information on a particular item.

Best Practice
Gives advice on the easiest and best way to proceed.

Material Damage
Points out information particular to potential damage of equipment.

Reference
Refers to particular information provided by other parties.

1.5 Trademarks and Brand Names

For easier legibility, registered brand names are listed below and will not be further tagged with their respective trademark. It must be understood that the brands (the below list is not necessarily conclud-ing) are protected by copyright and/or other intellectual property rights even if their legal trademarks are omitted in the later course of this document.

Brand Name	Trademark Owner
Adobe® Reader®	© Adobe Systems Incorporated, USA-San Jose, CA
EtherCAT®	© EtherCAT Technology Group, DE-Nuremberg
Excel	© Microsoft Corporation, USA-Redmond, WA
Micro-Fit™ Mini-Fit Jr.™	© Molex, USA-Lisle, IL
Pentium®	© Intel Corporation, USA-Santa Clara, CA
Windows®	© Microsoft Corporation, USA-Redmond, WA

Table 1-2 Brand Names and Trademark Owners

1.6 Sources for additional Information

Find the latest edition of additional documentation and software also on the Internet:
➔www.maxonmotor.com

For further details and additional information, please refer to below listed sources:

#	Reference
[1]	CiA 301 Communication Profile for Industrial Systems www.can-cia.org
[2]	CiA 402 Device Profile for Drives and Motion Control www.can-cia.org
[3]	CiA 305 Layer Setting Services (LSS) and Protocols www.can-cia.org
[4]	CiA 306 Electronic Data Sheet Specification www.can-cia.org
[5]	Konrad Etschberger: Controller Area Network ISBN 3-446-21776-2
[6]	maxon motor: EPOS3 EtherCAT Communication Guide EPOS Positioning Controller DVD or www.maxonmotor.com
[7]	Dr. Urs Kafader: The selection of high-precision microdrives ISBN 978-3-9520143-6-3 Also availably from «maxon academy» www.maxonmotor.com

Table 1-3 Sources for additional Information

1.7 System Units

Unit Dimension	Definition
Position units	steps (quadcounts = 4 x Encoder Counts / Revolution)
Velocity units	rpm (Revolutions per Minute)
Acceleration units	rpm/s (Velocity Unit / Second)

Table 1-4 Default Unit Dimensions

1.8 Copyright

maxon motor ag
Brünigstrasse 220
P.O.Box 263
CH-6072 Sachseln
Switzerland

Phone +41 41 666 15 00
Fax +41 41 666 16 50

www.maxonmotor.com

••page intentionally left blank••

2 EtherCAT Integration (Master Beckhoff TwinCAT)

2.1 In Brief

A wide variety of operating modes permit flexible configuration of drive and automation systems by using positioning, speed and current regulation. The built-in EtherCAT interface allows networking to multiple axes drives as well as online commanding by EtherCAT master units.

For fast communication with several EPOS3 70/10 EtherCAT devices, use the EtherCAT protocol. The individual devices of a network are commanded by a EtherCAT master.

2.1.1 Objective

The present Application Note explains how to integrate the EPOS3 EtherCAT positioning controller in the Master Beckhoff TwinCAT.

Contents

2.1.2 Scope

Hardware	Order #	Firmware Version	Reference
EPOS3 EtherCAT		2200h	Firmware Specification Communication Guide
EPOS3 70/10 EtherCAT	411146	2200h or higher	Cable Starting Set Hardware Reference

Table 2-5 Master Beckhoff TwinCAT – covered Hardware and required Documents

2.1.3 Tools

Tools	Description
Software	«EPOS Studio» Version 2.00 or higher

Table 2-6 Master Beckhoff TwinCAT – recommended Tools

2.2 Functionality

SDOs are used to access the object dictionary. The corresponding interface is CoE. The EPOS3 EtherCAT is described with an XML file bearing the so called ESI (EtherCAT Slave Information).

2.3 Integrating ESI Files

To integrate an EPOS3 EtherCAT axis into the Beckhoff Master System, copy the ESI (EtherCAT Slave Information) XML file to the following folder. Note that the actual folder designation (***) depends on the TwinCAT version you are using:

- For **TwinCAT XAE** use path “C:\TwinCAT***3.1\Config\Io\EtherCAT\”.
- For **TwinCAT2** use path “C:\TwinCAT\Io\EtherCAT\”.

2.4 Scanning the EtherCAT Slave Device

1) Connect the EPOS3 EtherCAT to the EtherCAT Master and turn on power.
2) Open the Beckhoff System Manager and create a new project using menu ¤File¤, then ¤New¤.
3) Open menu ¤Options¤, then select ¤Show Real Time Ethernet Compatible Devices¤.

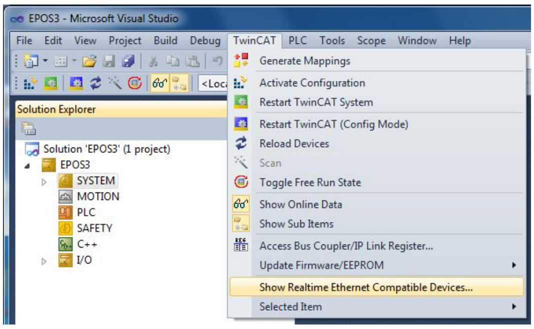


Figure 2-2 TwinCAT System Manager – Create new Project

4) If “Installed and ready to use devices” does not list a network card, you must install the EtherCAT driver for one of the present network cards.
   a) Click one of the listed network cards.
   b) Click button ¤Install¤.

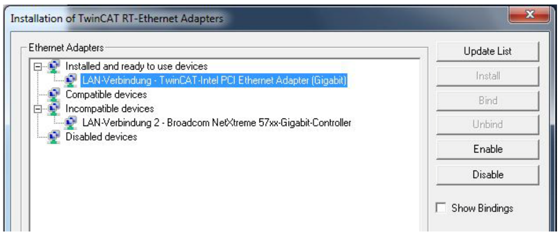


Figure 2-3 TwinCAT System Manager – Installation of TwinCAT RT Ethernet Adapters

5) In the TwinCAT System Manager navigation tree, click right on ¤I/O Devices¤, then select ¤Scan Devices¤.

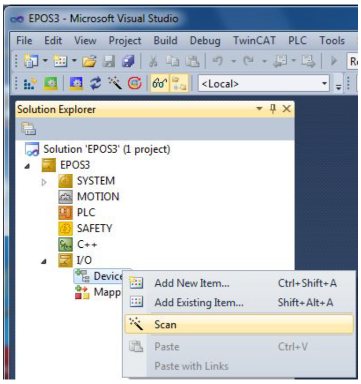


Figure 2-4 TwinCAT System Manager – Scan Devices

6) Click ¤OK¤ to confirm.

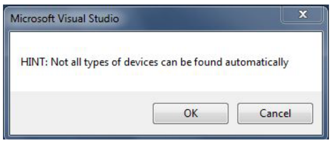


Figure 2-5 TwinCAT System Manager – Confirmation

7) All detected E/A devices (network cards) will be listed.
Tick to select the network card to which the EtherCAT devices were connected to. Untick all the others and click ¤OK¤.

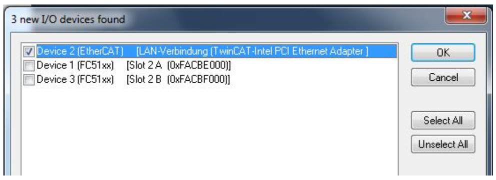


Figure 2-6 TwinCAT System Manager – New I/O Devices found

8) Click ¤Yes¤ to confirm.

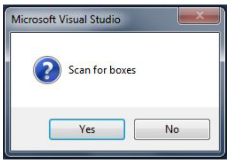


Figure 2-7 TwinCAT System Manager – Scan for Boxes Confirmation

9) The TwinCAT System Manager now searches for connected devices. If one or more controller were found, the following messages appears.

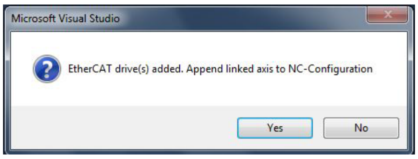


Figure 2-8 TwinCAT System Manager – Add Drives Message

10) Depending on the intended use:

- Click ¤Yes¤ if you plan to use the drive as a NC-Configuration.
- Click ¤No¤ if you do not plan to use the drive a NC-Configuration.

11) Click ¤Yes¤ to confirm.

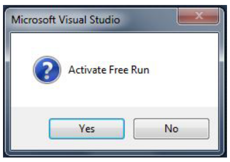


Figure 2-9 TwinCAT System Manager – Activate Free Run Message

12) Save the project.

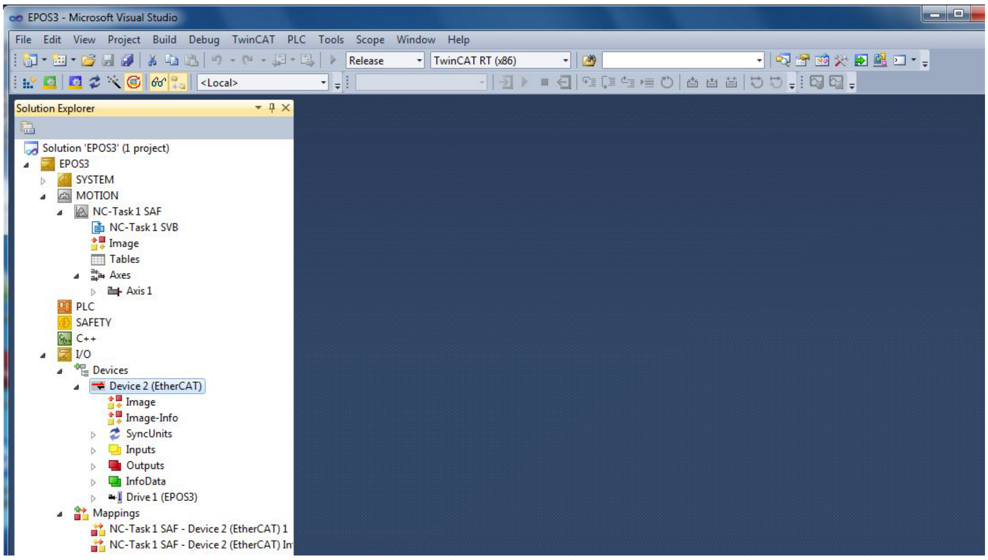


Figure 2-10 TwinCAT System Manager – Save Project

2.5 Changing Operating Modes

Via the EtherCAT interface, usually the following operating modes will be used:

- ➔“Cyclic Synchronous Position (CSP)” on page 6-66
- ➔“Cyclic Synchronous Velocity (CSV)” on page 6-67
- ➔“Cyclic Synchronous Torque (CST)” on page 6-68

If the controller will be operated in «Cycle Synchronous Mode», PDO Mapping must be configured accordingly by defining “Slots”.

Additionally, the following “normal” EPOS operating modes may be used:

- Profile Position Mode
- Profile Velocity Mode

1) Upon recognition of the involved axes, the following structure tree (example) will be displayed.

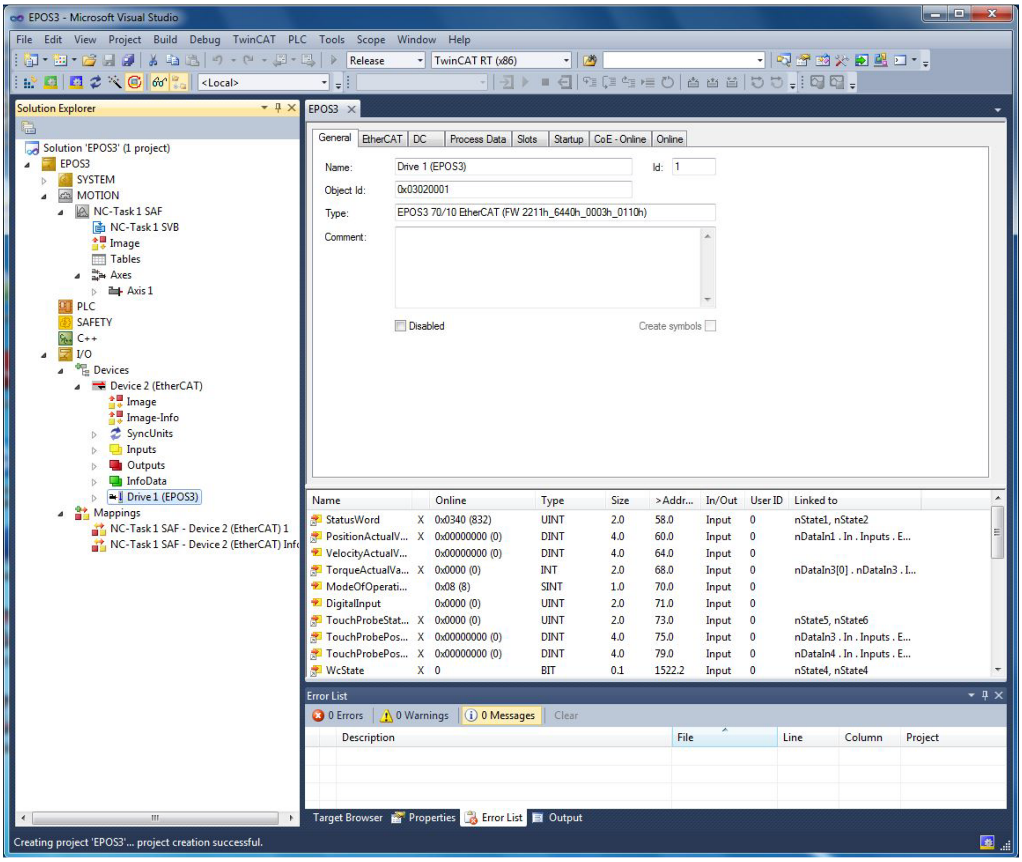


Figure 2-11 TwinCAT System Manager – Structure Tree

2) Use the tab ¤Slots¤ to allocate the operating mode is configured using.
   a) Select a Slot from the left pane ¤Slot¤.
   b) Select desired operating mode from right pane ¤Module¤.

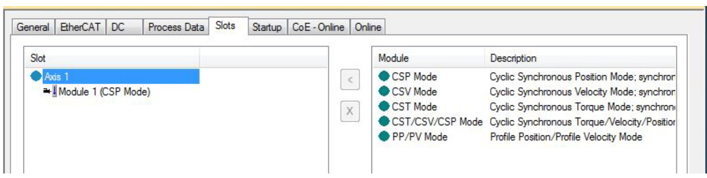


Figure 2-12 TwinCAT System Manager – Configuration of Slots

2.6 Verify CSP Settings

1) Enable the Distributed Clock from the EPOS3 Drive.

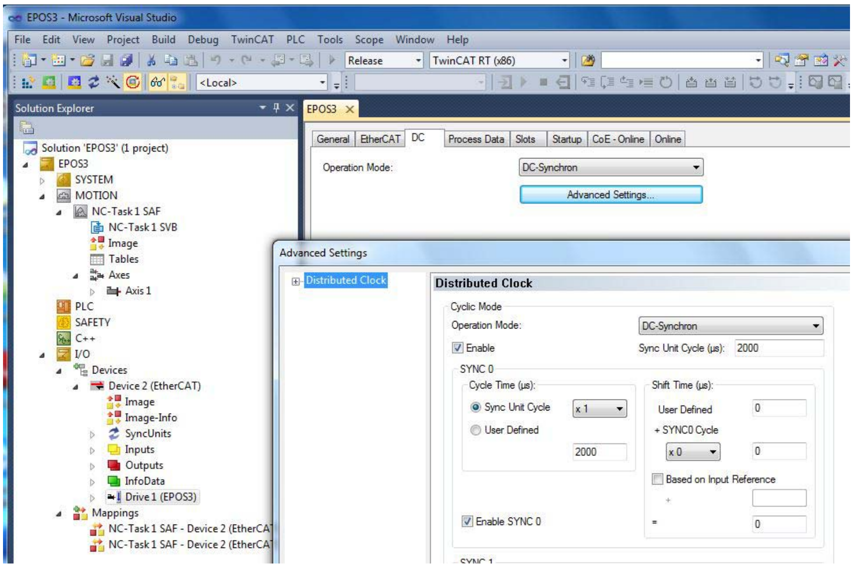


Figure 2-13 TwinCAT System Manager – Distributed Clock

2) Set cycle time of NC-Task 1 SAF to 2 ms.

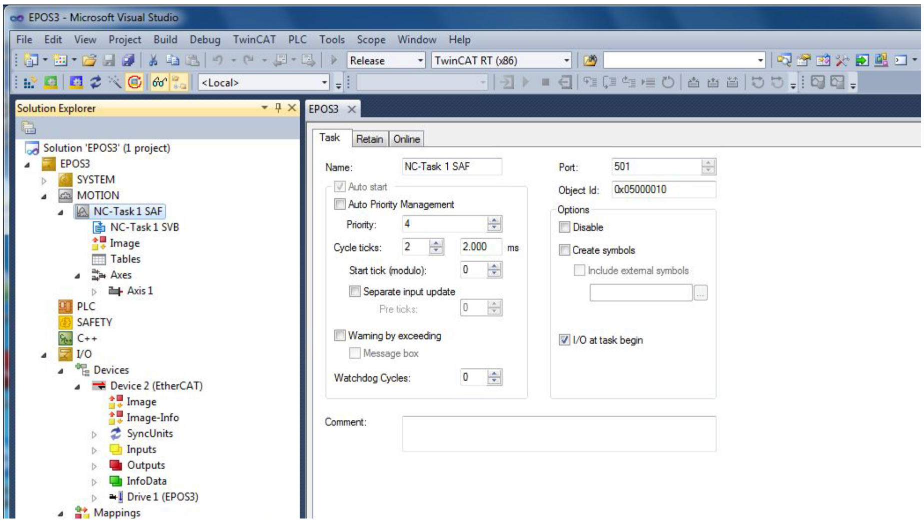


Figure 2-14 TwinCAT System Manager – Cycle Ticks

2.7 Configuration of the Axis

1) In tab ¤Settings¤, verify that ¤Link To I/O…¤ is assigned to the MAXPOS axis (naming by your choice).

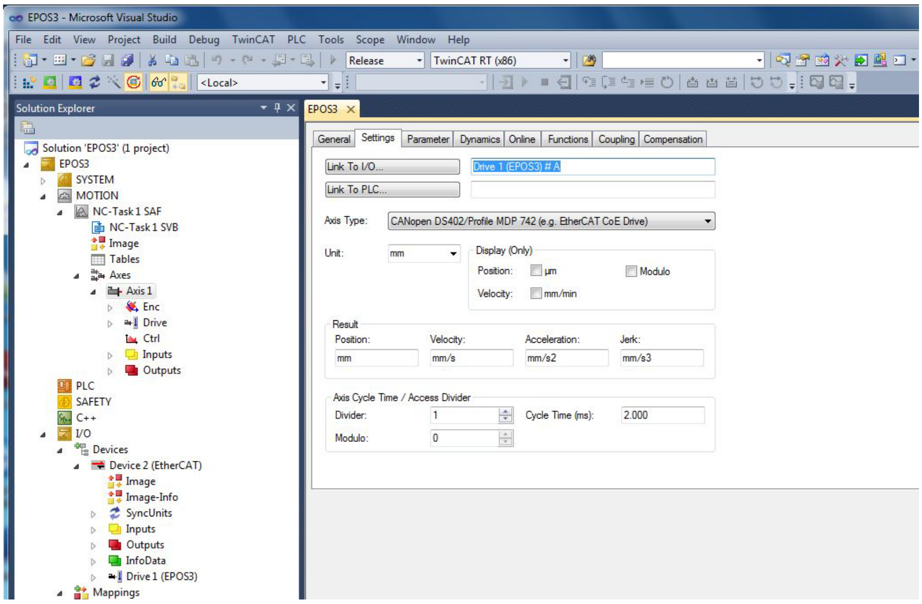


Figure 2-15 TwinCAT System Manager – Axis Link

2) In tab ¤Parameter¤, adjust the motor speed settings as to the motor's capability and to the supply voltage.

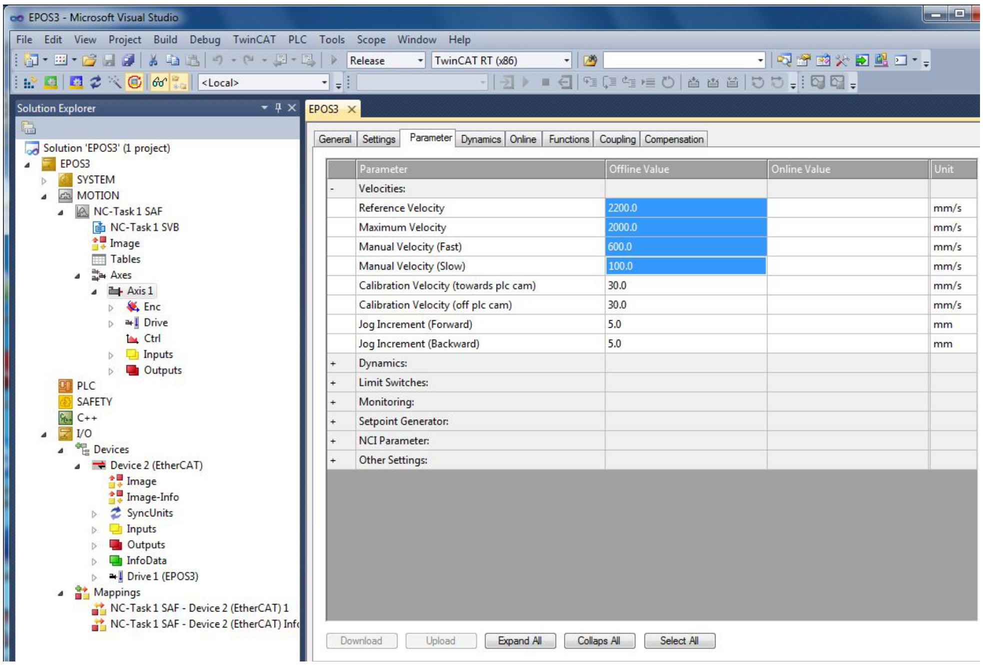


Figure 2-16 TwinCAT System Manager – Speed Settings

3) Set Dead Time Compensation to approximately three to four times the set NC-Task SAF Cycle ticks (➔“Verify CSP Settings” on page 2-18; step 2)

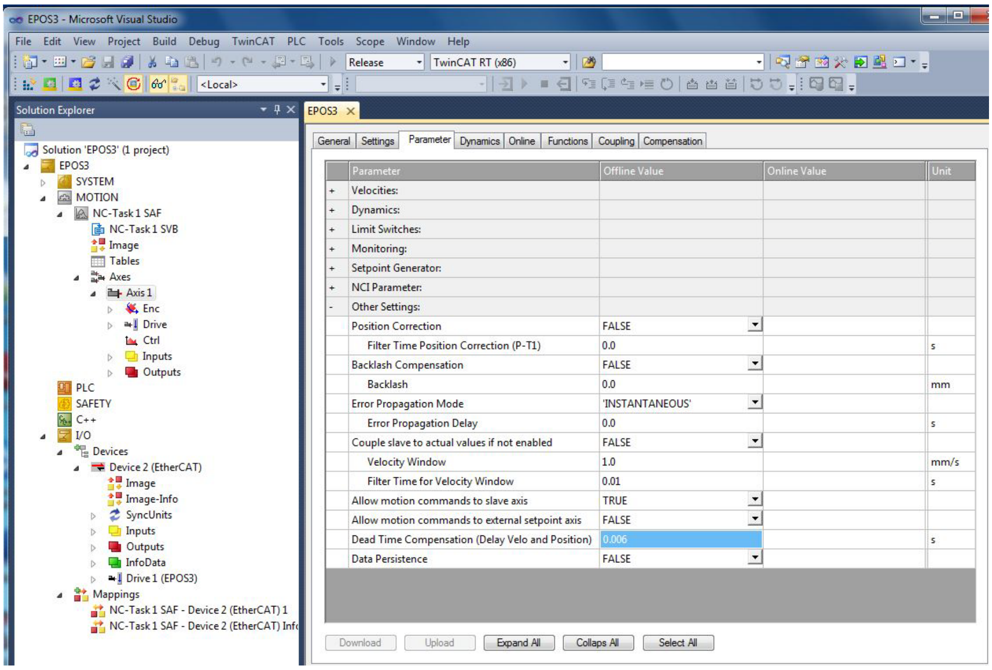


Figure 2-17 TwinCAT System Manager – Dead Time Compensation

4) Make sure to set the correct encoder resolution.

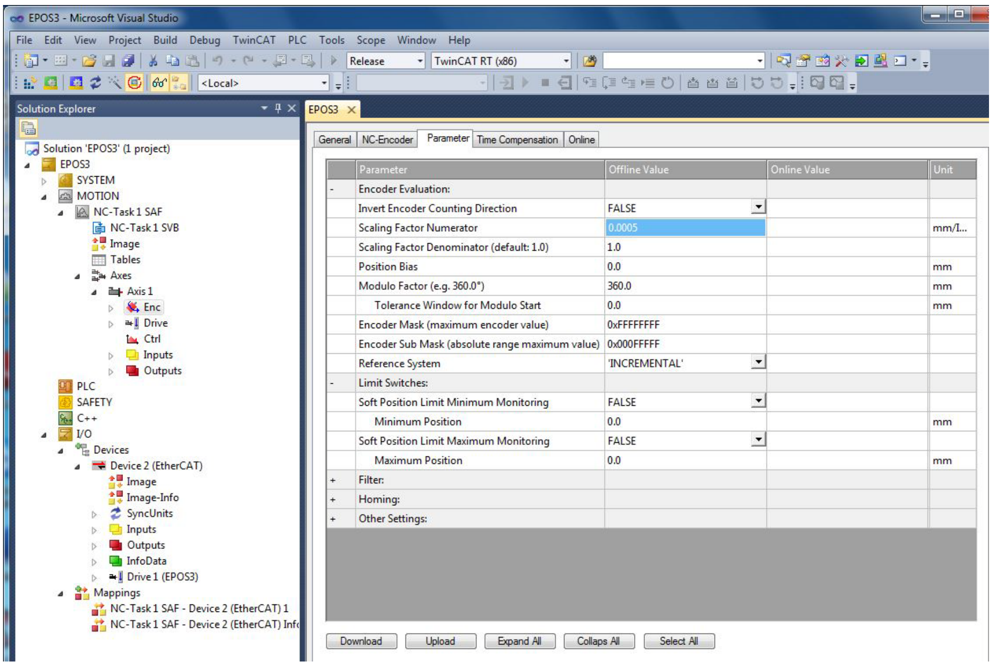


Figure 2-18 TwinCAT System Manager – Encoder Settings

3 Digital Inputs and Outputs

3.1 In Brief

Drive systems typically require inputs and outputs – "Home Switch", Positive/Negative Limit Switches" and "Brake Output" with sufficient current, just to mention a few.

The inputs and outputs can easily be configured using the «Configuration Wizard» and may be changed online via EtherCAT.

3.1.1 Objective

The present Application Note explains the functionality of digital inputs and outputs and features "in practice examples" suitable for daily use.

Contents

3.1.2 Scope

Hardware	Order #	Firmware Version	Reference
EPOS3 EtherCAT		2200h	Firmware Specification
EPOS3 70/10 EtherCAT	411146	2200h or higher	Cable Starting Set Hardware Reference

Table 3-7 Digital Inputs and Outputs – covered Hardware and required Documents

3.1.3 Tools

Tools	Description
Crimper	Molex hand crimper (63819-0000)
	Molex hand crimper (63819-0900)
Software	«EPOS Studio» Version 2.00 or higher

Table 3-8 Digital Inputs and Outputs – recommended Tools

3.2 Functionality

3.2.1 Digital Inputs

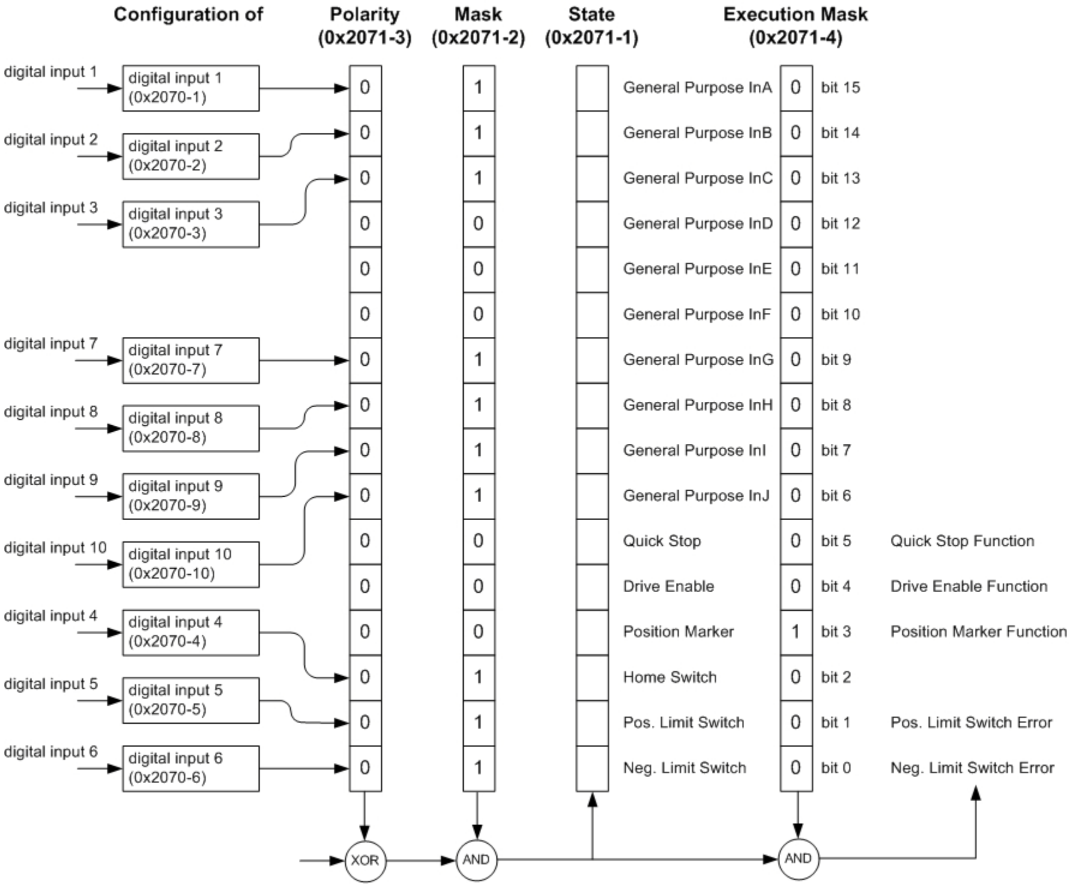


Figure 3-19 Digital Input Functionality – EPOS3 70/10 EtherCAT Overview (default Configuration)

Configuration Parameter

Name	Index	Sub-index	Description
Configuration of Digital Input 1 (➔Table 3-11)	0x2070	0x01	Defines functionality assigned to DigIN1.
Configuration of Digital Input 2 (➔Table 3-11)	0x2070	0x02	Defines functionality assigned to DigIN2.
Configuration of Digital Input 3 (➔Table 3-11)	0x2070	0x03	Defines functionality assigned to DigIN3.
Configuration of Digital Input 4 (➔Table 3-11)	0x2070	0x04	Defines functionality assigned to DigIN4.
Configuration of Digital Input 5 (➔Table 3-11)	0x2070	0x05	Defines functionality assigned to DigIN5.
Configuration of Digital Input 6 (➔Table 3-11)	0x2070	0x06	Defines functionality assigned to DigIN6.
Configuration of Digital Input 7 (➔Table 3-11)	0x2070	0x07	Defines functionality assigned to DigIN7.
Configuration of Digital Input 8 (➔Table 3-11)	0x2070	0x08	Defines functionality assigned to DigIN8.
Configuration of Digital Input 9 (➔Table 3-11)	0x2070	0x09	Defines functionality assigned to DigIN9.
Configuration of Digital Input 10 (➔Table 3-11)	0x2070	0x0A	Defines functionality assigned to DigIN10.
Digital Input Functionalities Mask (➔Table 3-12)	0x2071	0x02	Displayed state of Digital Input Functionalities may be filtered.
Digital Input Functionalities Polarity (➔Table 3-13)	0x2071	0x03	Polarity of Digital Input Functionalities.
Digital Input Functionalities Execution Mask (➔Table 3-12)	0x2071	0x04	Execution of Digital Input Functionalities can be inhibited.

Table 3-9 Digital Input – Configuration Parameter

Input Parameter

Name	Index	Sub-index	Description
Digital Input Functionalities State (➔Table 3-12)	0x2071	0x01	Display state of Digital Input Functionalities.

Table 3-10 Digital Input – Input Parameter

Input Configuration Values

Parameter “Configuration of Digital Input” defines bit position in “Digital Input Functionalities State”.

Value	Functionality	Description
15	General Purpose A	State can be read.
14	General Purpose B	State can be read.
13	General Purpose C	State can be read.
12	General Purpose D	State can be read.
11	General Purpose E	State can be read.
10	General Purpose F	State can be read.
9	General Purpose G	State can be read.
8	General Purpose H	State can be read.
7	General Purpose I	State can be read.
6	General Purpose J	State can be read.
5	Quick Stop	Set Quick Stop profile.
4	Device Enable	Enables/disables device.
3	Position Marker	Samples current position.
2	Home Switch	Used in some homing modes.
1	Positive Limit Switch	Generates limit error / used in some homing modes.
0	Negative Limit Switch	Generates limit error / used in some homing modes.

Table 3-11 Digital Input – Input Configuration Values

Parameter Description

Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
General Purpose A	General Purpose B	General Purpose C	General Purpose D	General Purpose E	General Purpose F	General Purpose G	General Purpose H

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
General Purpose I	General Purpose J	Quick Stop	Device Enable	Position Marker	Home Switch	Pos. Limit Switch	Neg. Limit Switch

Table 3-12 Digital Input – Execution Mask Parameter

Polarity Values

Bit	0	1
associated pin	high active	low active

Table 3-13 Digital Input – Polarity Values

Note

- *“Digital Input Functionalities State” will only be displayed, if “Digital Input Functionalities Mask” is set to “Enable”.*
- *“Digital Input Functionalities State” enables/disables the specific function.*

3.2.2 Digital Outputs

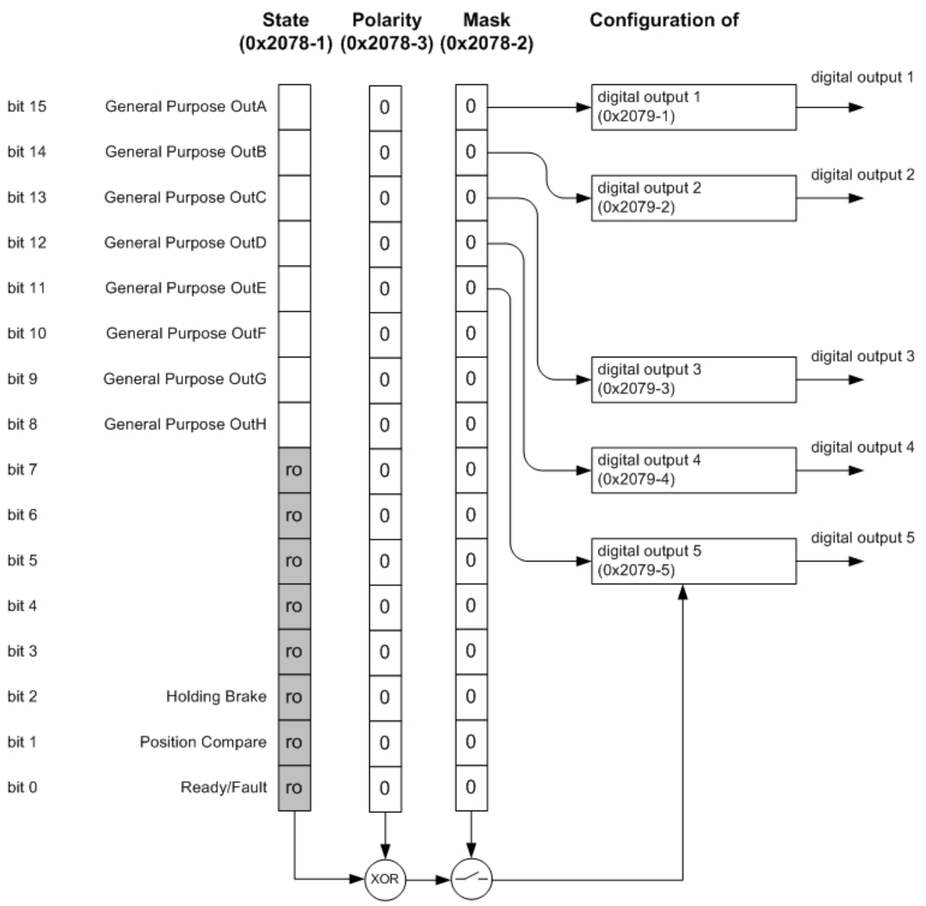


Figure 3-20 Digital Output Functionality – EPOS3 70/10 EtherCAT Overview (default Configuration

Configuration Parameter

Name	Index	Sub-index	Description
Configuration of Digital Output 1 (➔Table 3-15)	0x2079	0x01	Defines functionality assigned to DigOUT1.
Configuration of Digital Output 2 (➔Table 3-15)	0x2079	0x02	Defines functionality assigned to DigOUT2.
Configuration of Digital Output 3 (➔Table 3-15)	0x2079	0x03	Defines functionality assigned to DigOUT3.
Configuration of Digital Output 4 (➔Table 3-15)	0x2079	0x04	Defines functionality assigned to DigOUT4.
Configuration of Digital Output 5 (➔Table 3-15)	0x2079	0x05	Defines functionality assigned to DigOUT5.
Digital Output Functionalities State (➔Table 3-16)	0x2078	0x01	State of digital outputs may be set.
Digital Output Functionalities Mask (➔Table 3-16)	0x2078	0x02	Digital outputs may be filtered.
Digital Input Functionalities Polarity (➔Table 3-17)	0x2078	0x03	Change of polarity of digital output.

Table 3-14 Digital Output – Configuration Parameter

Output Configuration Values

Parameter “Configuration of Digital Output” defines bit position in “Digital Output Functionalities State”.

Value	Functionality	Description
15	General Purpose A	State can be read.
14	General Purpose B	State can be read.
13	General Purpose C	State can be read.
12	General Purpose D	State can be read.
11	General Purpose E	State can be read.
10…8	not used	–
7…3	reserved	–
2	Holding Brake	Active output = activated brake Inactive output = deactivated brake
1	Position compare	Trigger output of Position Compare.
0	Ready / Fault	Active on Device Ready / Inactive on Fault

Table 3-15 Digital Output – Output Configuration Values

Parameter Description

Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10…3	Bit 2	Bit 2	Bit 0
General Purpose A	General Purpose B	General Purpose C	General Purpose D	General Purpose E	not used / reserved	Holding Brake	Position Compare	Ready / Fault

Table 3-16 Digital Output – Execution Mask Parameter

Polarity Values

Bit	0	1
associated pin	not inverted 1 → high 0 → low	inverted 0 → high 1 → low

Table 3-17 Digital Output – Polarity Values

Note

A change in “Digital Output Functionalities State” is only of effect, if “Digital Output Functionalities Mask” is set to “Enable”.

3.3 Connection

3.3.1 EPOS3 70/10 EtherCAT

Signal Cable 16core (275932) – Connector J6

Figure 3-21 Signal Cable 16core

Technical Data	
Cable cross-section	16 x 0.14 mm^2
Length	3 m
Head A	Molex Micro-Fit 3.0 16 poles (430-25-1600) Molex Micro-Fit 3.0 female crimp terminals (43030-xxxx)
Head B	Cable end sleeves 0.14 mm2

Table 3-18 Signal Cable 16core – Technical Data

Wire	Head A Pin	Head B Pin	Twisted Pair	Signal	Description
white	1		–	IN_COM2	Common signal 2 for DigIN4…6
brown	2		–	IN_COM1	Common signal 1 for DigIN1…3
green	3		–	DigIN6	Digital Input 6 “Negative Limit Switch”
yellow	4		–	DigIN5	Digital Input 5 “Positive Limit Switch”
grey	5		–	DigIN4	Digital Input 4 “Home Switch”
pink	6		–	DigIN3	Digital Input 3 “General Purpose”
blue	7		–	DigIN2	Digital Input 2 “General Purpose”
red	8		–	DigIN1	Digital Input 1 “General Purpose”
black	9		–	+V Opto IN	External supply input voltage for Digital Outputs (+12…24 VDC)
violet	10		–	DigOUT4	Digital Output 4 “Brake / General Purpose”
grey/ pink	11		–	DigOUT3	Digital Output 3 “Brake / General Purpose”
red/blue	12		–	DigOUT2	Digital Output 2 “General Purpose”
white/ green	13		–	DigOUT1	Digital Output 1 “General Purpose”
brown/ green	14		–	DigOUT_Gnd	Digital OUT ground reference to “+V Opto IN”
white/ yellow	15		–	DigIN11	Digital Input 11 “Power Stage Enable”
yellow/ brown	16		–	IN_COM3	Common signal 3 for DigIN11

Table 3-19 Signal Cable 16core – Pin Assignment EPOS3 70/10 EtherCAT

Signal Cable 6x2core (300586) – Connector J5

Figure 3-22 Signal Cable 6x2core

Technical Data	
Cable cross-section	6 x 2 x 0.14 mm²
Length	3.00 m
Head A	Molex Micro-Fit 3.0 12 poles (430-25-1200) Molex Micro-Fit 3.0 female crimp terminals (43030-xxxx)
Head B	Cable end sleeves 0.14 mm2

Table 3-20 Signal Cable 6x2core – Technical Data

Wire	Head A Pin	Head B Pin	Twisted Pair	Signal	Description
white	1		1	DigIN10/	Digital Input 10 “High Speed Command” complement
brown	2			DigIN10	Digital Input 10 “High Speed Command”
green	3		2	DigIN9/	Digital Input 9 “High Speed Command” complement
yellow	4			DigIN9	Digital Input 9 “High Speed Command”
grey	5		3	DigIN7/	Digital Input 7 “High Speed Command” complement
pink	6			DigIN7	Digital Input 7 “High Speed Command”
blue	7		4	DigIN8/	Digital Input 8 “High Speed Command” complement
red	8			DigIN8	Digital Input 8 “High Speed Command”
black	9		5	$+V_{AUX}$	Auxiliary output voltage +5 VDC / 150 mA
violet	10			D_GND	Digital signal ground
grey/ pink	11		6	DigOUT5/	Digital Output 5 “High Speed Command” complement
red/blue	12			DigOUT5	Digital Output 5 “High Speed Command”

Table 3-21 Signal Cable 6x2core – Pin Assignment EPOS3 70/10 EtherCAT

3.4 Configuration

Configuration is handled by a dynamic wizard assisting you in selecting desired functions and assigning them to inputs and outputs of you choice.

Note
The following explanations show you how to initiate the Configuration Wizard. Its further coarse will then depend on the functions and options you will actually chose. The stated figures are thereby meant as examples.

3.4.1 Step A: Open I/O Configuration Wizard

1) Complete standard system configuration (Startup Wizard) in «EPOS Studio».
2) Doubleclick ¤I/O Configuration Wizard¤ to commence configuration.

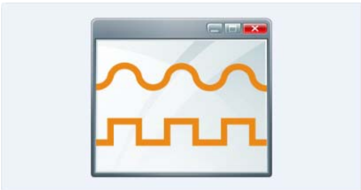


Figure 3-23 Open I/O Configuration Wizard

3) A screen will appear showing the number of I/Os available for configuration.
4) Click ¤Next¤ to continue.

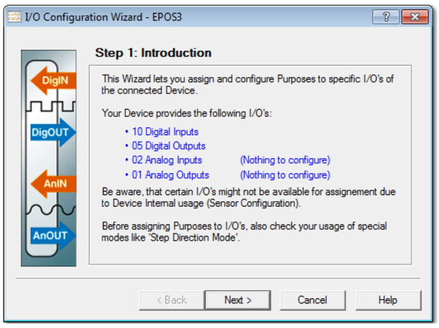


Figure 3-24 Configuration Wizard – Introduction

3.4.2 Step B: Configure Digital Inputs

1) Select predefined functions you wish to use by ticking respective check boxes. An available digital input will automatically be assigned to your selection.
2) If you wish to assign a particular digital input to a given function, select desired input from the ¤Dropdown menu¤ in column “Input”.
3) Click ¤Next¤ to continue.

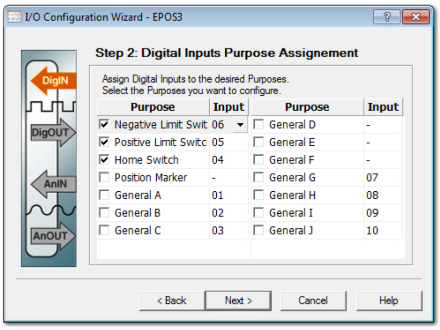


Figure 3-25 Configuration Wizard – Configure Digital Inputs

4) Define mask, type of switch (NPN or PNP) and switch output state.
5) Set limit switch error.
6) Click ¤Next¤ to continue.
7) Repeat for every earlier selected digital input.

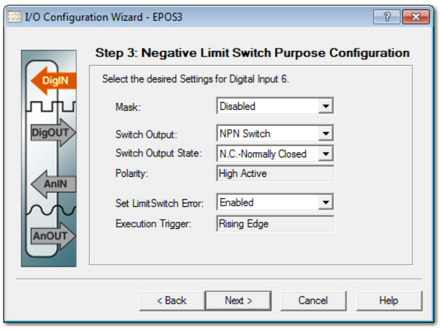


Figure 3-26 Configuration Wizard – Configure Digital Input Functionality

3.4.3 Step C: Configure Digital Outputs

1) Select predefined functions you wish to use by ticking respective check boxes. An available digital output will automatically be assigned to your selection.
2) If you wish to assign a particular digital output to a given function, select desired input from the ¤Dropdown menu¤ in column “Output”.
3) Click ¤Next¤ to continue.

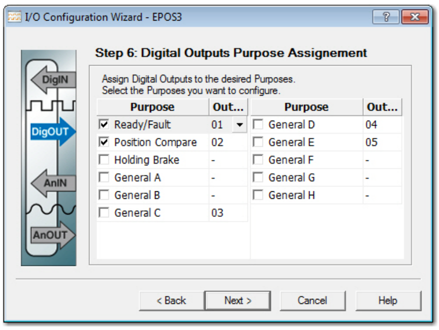


Figure 3-27 Configuration Wizard – Configure Digital Outputs

3.4.4 Step D: Save Configuration

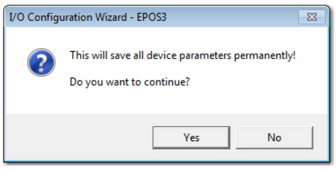


Figure 3-28 Safe Configuration

Note
You may check the status and alter the configuration at any time using the «I/O Monitor».

3.5 Wiring Examples

3.5.1 EPOS3 70/10 EtherCAT

3.5.1.1 Proximity Switches

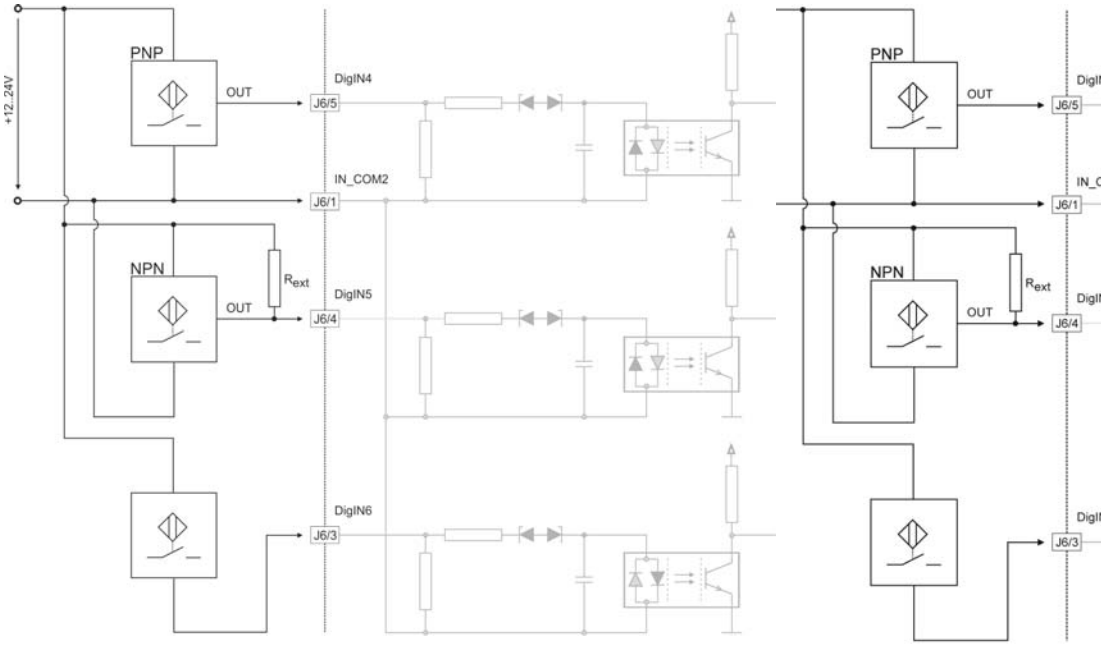


Figure 3-29 EPOS3 70/10 EtherCAT – DigIN4…6 / PNP/NPN Proximity Switches

Best Practice

- *We recommend the use of 3-wire PNP proximity switches.*
- *The use of 3-wire NPN proximity switches requires an additional external pull-up resistor:*
  – R_{ext} (12 V) = 560 Ω (300 mW)
  – R_{ext} (24 V) = 3 kΩ (200 mW)
- *The use of 2-wire proximity switches is possible.*

3.5.1.2 Permanent Magnet Brake

EPOS3 70/10 EtherCAT output 4 permits direct activation of loads with very high current demand (such as motor brakes and warning lights, etc.).

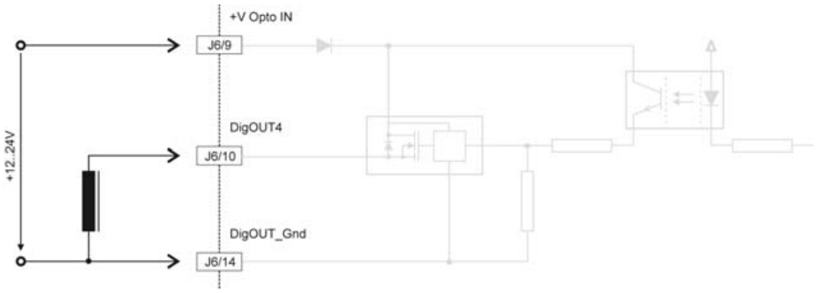


Figure 3-30 EPOS3 70/10 EtherCAT – DigOUT4 / permanent Magnet Brake

••page intentionally left blank••

4 Interpolated Position Mode

4.1 In Brief

A wide variety of operating modes permit flexible configuration of drive and automation systems by using positioning, speed and current regulation. The built-in EtherCAT interface allows networking to multiple axes drives as well as online commanding by EtherCAT master units.

For fast communication with several EPOS3 70/10 EtherCAT devices, use the EtherCAT protocol. The individual devices of a network are commanded by a EtherCAT master.

4.1.1 Objective

«Interpolated Position Mode» is used to control multiply coordinated axes or a single axis with the need for time interpolation of setpoint data. The trajectory is calculated by the EtherCAT master and passed on to the controller's interpolated position buffer as a set of points. The controller then reads the points from the buffer and performs linear or cubic interpolation between them.

The present Application Note explains structure, functionality and use of the operation mode «Interpolated Position Mode» and features "in practice examples" suitable for daily use.

Contents

4.1.2 Scope

Hardware	Order #	Firmware Version	Reference (➔page 1-10)
EPOS3 EtherCAT		2200h	Firmware Specification Communication Guide (➔[6]
EPOS3 70/10 EtherCAT	411146	2200h or higher	

Table 4-22 Interpolated Position Mode – covered Hardware and required Documents

4.1.3 Tools

Tools	Description
Software	«EPOS Studio» Version 2.00 or higher

Table 4-23 Interpolated Position Mode – recommended Tools

4.2 In Detail

4.2.1 Introductory Analogy

Let us assume: In a company, a department manager must convert the department goals into clear tasks for his coworkers. It must be considered that the individual tasks oftentimes stand to each other in close interdependency. Thus, the department manager will gladly count on capable coworkers, being able to solve their tasks even on basis on just substantial data. For the solution's quality, it is in particular important that it…

a) is factually correct; i.e. it will not require further checks,

b) will be finished in time and

c) was reached efficiently.

The functionality «Interpolated Position Mode» values up the positioning controller EPOS3 EtherCAT to such a "capable coworker" in a superordinate drive system. Following, the thesis' description:

In a drive system, normally several axes must be moved according to the guidelines of a central controller. This can take place in the way that each local axis controller receives the next target position in real time – in time and at the same time to each sampling instance. This strategy has the advantage that the local controllers need only little intelligence. However, the central controller must compute target positions for every sampling interval and must communicate the data to every local controller in real time.

As to above analogy…

- it would be favorable if only few, but substantial points of the driving profiles would be considered,
- it would be desirable if the corresponding data could be transmitted to the local controller not necessarily at the same time, but rather in time.

Both goals can be reached by interpolation and data buffering.

First, the central controller decides which points of the local trajectories are substantial for a synchronized total movement. Then, each relevant point of the local trajectories is supplemented with the corresponding velocity and time – i.e. triplicates of the kind (position, velocity, time = PVT) are formed. These triplicates are then transferred to the associated axis controllers, in time. Each local controller possesses a buffer to receive these data. EPOS3 EtherCAT's buffer covers 64 locations for triplicates. The data transfer to the EPOS3 EtherCAT is in time as long as the buffer contains 1 to 64 new triplicates.

In EPOS3 EtherCAT, local position regulation is sampled with a rate of 1 kHz. Thus, requiring 1000 target positions per second in real time. These target positions are computed in EPOS3 EtherCAT by means of interpolation. Each triplicate forms a base point with the abscissa time and the two ordinates position and velocity. Therefore, two triplicates deliver two abscissas and four corresponding ordinates, permitting an interpolation polynomial of third order unambiguously computed between the two base points. The computation, as well as the evaluation of the polynomial in the local sampling clock, take place on basis of simple arithmetic and are efficiently executed by the EPOS3 EtherCAT.

The endpoint of the polynomial [n] forms the starting point of the polynomial [n+1]. Therefore, it is sufficient to indicate only the relative time in a data triplicate (i.e. the length of the time interval). In fact, with the EPOS3 EtherCAT, the time distance of two base points can be selected between 1 ms and 255 ms. This interval length can be adapted by the central controller to realize the desired total movement.

Finally, Interpolated Position Mode can be qualified as follows: The resulting smooth driving profiles, as well as the close temporal synchronization allow to superpose several high-dynamic single movements to a precise total movement in a drive system.

4.2.2 General Description

The Interpolated Position Mode described in the CANopen specification CiA 402 V3.0 is a general case. The objects are well-specified or a linear interpolation (PT). The interpolation type can also be extended by manufacturer-specific algorithms (selectable by «Interpolation Submode Selection», Object 0x60C0).

4.2.3 Spline Interpolation

For the Interpolated Position Mode, the interpolation type is a cubic spline interpolation. The higher-level trajectory planner sends a set of interpolation points by PVT reference point. Each PVT reference point contains information on position, velocity and time of a profile segment end point. The trajectory generator of the drive performs a third order interpolation between the actual and the next reference point.

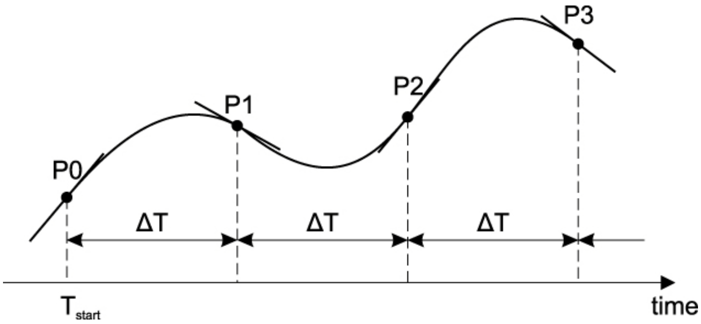


Figure 4-31 Interpolated Position Mode – PVT Principle

From two successive PVT reference points, the interpolation parameters a, b, c and d can be calculated:

$d = P[t_0] = P[n]$

$c = V[t_0] = V[n]$

$b = T^{-2}[n] * (3 * (P[n] - P[n-1]) + T[n] * (V[n] + 2 * V[n-1]))$

$a = T^{-3}[n] * (2 * (P[n] - P[n-1]) + T[n] * (V[n] + V[n-1]))$

The interpolated values for position, velocity and (possibly also) acceleration will be calculated as follows:

$P(t) = a * (t - t_0)^3 + b * (t - t_0)^2 + c * (t - t_0) + d$

$V(t) = 3a * (t - t_0)^2 + 2b * (t - t_0) + c$

$A(t) = 6a * (t - t_0) + 2b$

t_0: time of interpolation segment end (→ in this calculation t_0 is greater then t!)

It is not mandatory that the time intervals are identical.

4.3 IPM Implementation by maxon

The Interpolated Position Mode is implemented in the EPOS3 EtherCAT as an additional operational mode (operating mode 7 as specified in CiA 402 V3.0).

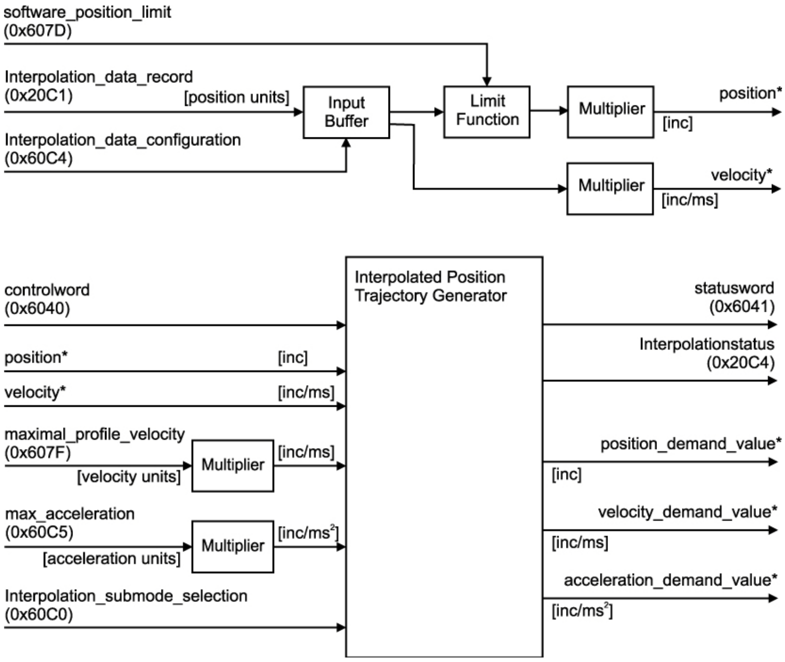


Figure 4-32 Interpolated Position Mode – Interpolation Controller

4.3.1 Interpolated Position Data Buffer

PVT reference points will be sent in a manufacturer-specific 64 bit data record of a complex data structure to a FIFO object.

4.3.1.1 Definition of complex Data Structure 0x0040

MSB		LSB
Time (unsigned8)	Velocity (signed24)	Position (signed32)

Table 4-24 Interpolated Position Mode – IPM Data Buffer Structure

4.3.1.2 Structure of the FIFO

The FIFO will be implemented by a circular buffer with the length of 64 entries

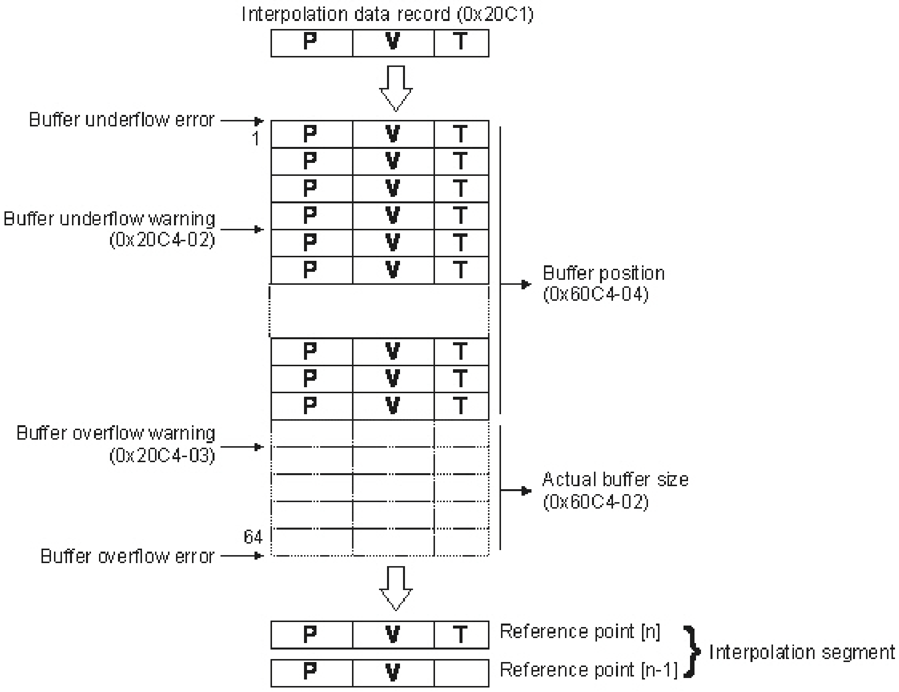


Figure 4-33 Interpolated Position Mode – FIFO Organization

4.3.2 Interpolated Position Mode FSA

The interpolated position finite state automaton is a sub FSA of the Operation enable state.

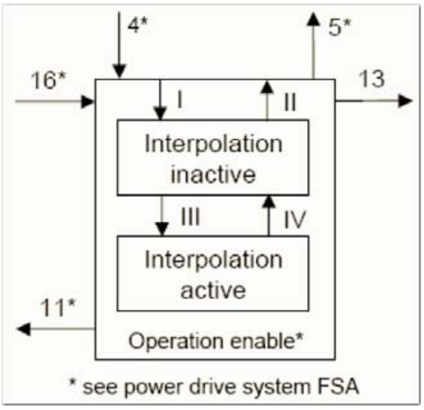


Figure 4-34 Interpolated Position Mode – FSA

FSA State	Function
Interpolation inactive	The drive device accepts input data and buffers it for interpolation calculations, but does not move the axis.
Interpolation active	The drive device accepts input data and moves the axis.

Table 4-25 Interpolated Position Mode – FSA States and supported Functions

Transition	Event		Action
I	ip mode selected (➔object 6060h, page 4-48)		clear data buffer
II	ip mode not selected (➔object 6060h, page 4-48)		none
III	enable ip mode:	set Controlword bit 4 to 1	none
IV	disable ip mode:	set Controlword bit 4 to 0 or ip data record with time = 0	none

Table 4-26 Interpolated Position Mode – Transition Events and Actions

4.3.3 Configuration Parameters

Parameter	Index	Description
Interpolation Sub Mode Selection	0x60C0	Indicates the actually chosen interpolation mode.
Interpolation Time Period	0x60C2	Indicates the configured interpolation cycle time.
Interpolation Data Configuration	0x60C4	Provides information on configuration and state of the buffer. It can also be used to clear the buffer.
Software Position Limit	0x607D	Contains the sub-parameters «Minimal Position Limit» and «Maximal Position Limit» that define the absolute position limits or the position demand value. A new target position will be checked against these limits
Position Window	0x6067	Permits definition of a position range around a target position to be regarded as valid. If the drive is within this area for a specified time, the related Statusword control bit 10 «Target reached» is set.
Position Window Time	0x6068	Defines the time or the position window.
Profile Velocity	0x6081	If calculated velocity of the interpolation exceeds this value, a warning bit in Interpolation Buffer Status Word will be set.
Profile Acceleration	0x6083	If calculated acceleration of the interpolation exceeds this value, a warning bit in Interpolation Buffer Status Word will be set.
Maximal Profile Velocity	0x607F	If calculated velocity of the interpolation exceeds this value, an error bit in Interpolation Buffer Status Word will be set and the device will switch to Fault reaction state.
Maximal Acceleration	0x60C5	If calculated acceleration of the interpolation exceeds this value, an error bit in Interpolation Buffer Status Word will be set and the device will switch to Fault reaction state.
Interpolation Status	0x20C4	The Interpolation buffer underflow/overflow warning level is configured in subindex 2 and 3.

Table 4-27 Interpolated Position Mode – Configuration Parameters

4.3.4 Commanding Parameters

Parameter	Index	Description
Controlword	0x6040	The mode will be controlled by a write access to the Controlword's mode-dependent bits.
Interpolation Data Record	0x20C1	Contains a FIFO to feed PVT reference points to the data buffer.

Table 4-28 Interpolated Position Mode – Commanding Parameters

Controlword (Interpolated Position Mode-specific Bits)

Bit 15…9	Bit 8	Bit 7	Bit 6, 5	Bit 4	Bit 3…0
➔FwSpec	Halt	➔FwSpec	reserved (0)	Enable ip mode	➔FwSpec

Table 4-29 Interpolated Position Mode – Controlword

Name	Value	Description
Enable ip mode	0	Interpolated Position Mode inactive
	1	Interpolated Position Mode active
Halt	0	Execute instruction of bit 4
	1	Stop axis with profile deceleration

Table 4-30 Interpolated Position Mode – Controlword Bits

4.3.5 Output Parameters

Parameter	Index	Description
Interpolation status	0x20C4	The mode's statusword is placed in subindex 1 of this object.
Statusword	0x6041	Mode state can be observed by Statusword bits.
Position Demand Value	0x6062	The output of the trajectory generator – it is used as input for the position control function.

Table 4-31 Interpolated Position Mode – Output Parameters

Statusword (Interpolated Position Mode-specific Bits)

Bit 15, 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9…0
➔FwSpec	reserved	ip mode active	➔FwSpec	Target reached	➔FwSpec

Table 4-32 Interpolated Position Mode – Statusword

Name	Value	Description
Target reached	0	Halt = 0: Target Position not (yet) reached Halt = 1: Axle decelerates
	1	Halt = 0: Target Position reached Halt = 1: Velocity of axle is 0
ip mode active	0	ip mode inactive
	1	ip mode active

Table 4-33 Interpolated Position Mode – Statusword Bits

4.3.6 Object Description in Detail

4.3.6.1 Interpolation Data Record

Description

Sets PVT reference points in the Interpolated Position Mode in the cubic spline interpolation sub-mode. The position is given absolute in [Position units], typically [qc]), the velocity is given in [Velocity units], typically [rpm]), and the time is given in [ms]. The object structure is defined in ➔“Interpolated Position Data Buffer” on page 4-38.

Remarks

Normally used to feed PVT reference points to the drive while a PVT motion is executing. Therefore the object may be mapped to a RxPDO with transmission type of 255 (asynchronous).

In the Interpolation active state at least two data records have to be in the FIFO. Otherwise a Queue underflow Emergency will be launched and the drive changes to Fault reaction state.

A data record with time = 0 changes the state to Interpolation inactive without any error.

Name	Interpolation Data Record	
Index	0x20C1	
Subindex	0x00	
Type	complex data structure 0x0040	
Access	WO	
Default Value	–	
Value Range	–	–
PDO Mapping	yes	

4.3.6.2 Interpolation Status

Description

Provides access to status information on the IP input data buffer.

Name	Interpolation Status
Index	0x20C4
Number of entries	0x03

Name	Interpolation Buffer Status	
Index	0x20C4	
Subindex	0x01	
Type	UNSIGNED16	
Access	RO	
Default Value	–	
Value Range	–	–
PDO Mapping	yes	

Bit 15	Bit 14	Bit 13…12	Bit 11…8	Bit 7…4	Bit 3…0
IP Mode active	Buffer enabled	reserved (0)	IPM buffer errors	reserved (0)	IPM buffer warnings

Table 4-34 Interpolation Buffer Status Word

Name	Bit	Value	Description
Underflow Warning	0	0	No buffer underflow warning
		1	Buffer underflow warning level (0x20C4-2) is reached
Overflow Warning	1	0	No buffer overflow warning
		1	Buffer overflow warning level (0x20C4-3) is reached
Velocity Warning	2	0	No profile velocity violation detected
		1	IPM velocity greater than profile velocity (0x6081) detected
Acceleration Warning	3	0	No profile acceleration violation detected
		1	IPM acceleration greater than profile acceleration (0x6083) detected
Underflow Error	8	0	No buffer underflow error
		1	Buffer underflow error (trajectory abort)
Overflow Error	9	0	No buffer overflow error
		1	Buffer overflow error (trajectory abort)
Velocity Error	10	0	No maximal profile velocity error
		1	IPM velocity greater than maximal profile velocity (0x607F) detected
Acceleration Error	11	0	No maximal profile acceleration error
		1	IPM acceleration greater than maximal profile acceleration (0x60C5) detected
Buffer enabled	14	0	Disabled access to the input buffer
		1	Access to the input buffer enabled
IP Mode active	15	0	IP mode inactive (same as bit 12 in statusword)
		1	IP mode active

Table 4-35 Interpolation Buffer Status Bits

Description

Gives the lower signalization level of the data input FIFO. If the filling level is below this border the warning flag (bit 0) in the Interpolation buffer status will be set.

Name	Interpolation Buffer Underflow Warning	
Index	0x20C4	
Subindex	0x02	
Type	UNSIGNED16	
Access	RW	
Default Value	4	
Value Range	0	63
PDO Mapping	no	

Description

Gives the higher signalization level of the data input FIFO. If the filling level is above this border the warning flag (bit 1) in the Interpolation buffer status will be set.

Name	Interpolation Buffer Overflow Warning	
Index	0x20C4	
Subindex	0x03	
Type	UNSIGNED16	
Access	RW	
Default Value	60	
Value Range	1	64
PDO Mapping	no	

4.3.6.3 Interpolation Sub Mode Selection

Description

Indicates the actually chosen interpolation mode.

Name	Interpolation Sub Mode Selection	
Index	0x60C0	
Subindex	0x00	
Type	INTEGER16	
Access	RW	
Default Value	−1	
Value Range	−1	−1
PDO Mapping	no	

Value	Description
−32 768…−2	Manufacturer-specific (reserved)
−1	cubic spline interpolation (PVT)
0	Linear interpolation (not yet implemented)
1…32 767	reserved

Table 4-36 Interpolation Sub Mode Selection – Definition

4.3.6.4 Interpolation Time Period

Description

Indicates the configured interpolation cycle time. The interpolation time period (subindex 0x01) value is given in $10^{\text{interpolation time index}}$ per second. The interpolation time index (subindex 0x02) is dimensionless.

Name	Interpolation Time Period
Index	0x60C2
Number of entries	0x02

Name	Interpolation Time Period Value	
Index	0x60C2	
Subindex	0x01	
Type	UNSIGNED8	
Access	RW	
Default Value	1	
Value Range	1	1
PDO Mapping	no	

Name	Interpolation Time Index	
Index	0x60C2	
Subindex	0x01	
Type	INTEGER8	
Access	RW	
Default Value	−3	
Value Range	−3	−3
PDO Mapping	no	

4.3.6.5 Interpolation Data Configuration

Description

Provides the maximal buffer size and is given in interpolation data records.

Name	Interpolation Data Configuration
Index	0x60C4
Number of entries	0x06

Name	Maximum Buffer Size	
Index	0x60C4	
Subindex	0x01	
Type	UNSIGNED32	
Access	RO	
Default Value	–	
Value Range	64	64
PDO Mapping	no	

Description

Provides the actual free buffer size and is given in interpolation data records.

Name	Actual Buffer Size	
Index	0x60C4	
Subindex	0x02	
Type	UNSIGNED32	
Access	RO	
Default Value	–	
Value Range	0	64
PDO Mapping	yes	

Description

The value 0 indicates a FIFO buffer organization.

Name	Buffer Organization	
Index	0x60C4	
Subindex	0x03	
Type	UNSIGNED8	
Access	RW	
Default Value	–	
Value Range	–	–
PDO Mapping	no	

Value	Description
0	FIFO buffer
1	Ring buffer (not supported)
2…255	reserved

Table 4-37 Buffer Organization – Definition

Description

Provides used buffer space and is given in interpolation data records. Writing to this object has no effect.

Name	Buffer Position	
Index	0x60C4	
Subindex	0x04	
Type	UNSIGNED16	
Access	RW	
Default Value	0	
Value Range	0	64
PDO Mapping	no	

Description

Interpolation data record size is 8 bytes.

Name	Size of Data Record	
Index	0x60C4	
Subindex	0x05	
Type	UNSIGNED8	
Access	WO	
Default Value	–	
Value Range	8	8
PDO Mapping	no	

Description

If 0 is written, the data buffer is cleared and the access to it is denied. If 1 is written, the access to the data buffer is enabled.

Related Objects

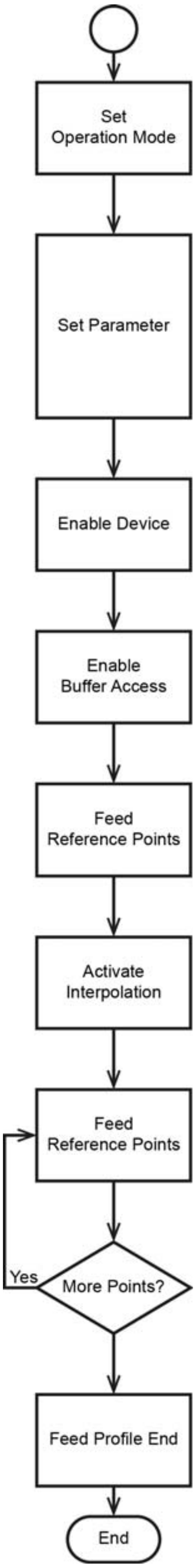
➔“Interpolation Status” on page 4-42

Name	Buffer Clear	
Index	0x60C4	
Subindex	0x06	
Type	UNSIGNED8	
Access	WO	
Default Value	0	
Value Range	0	1
PDO Mapping	no	

Value	Description
0	Clear input buffer (and all data records) access disabled
1	Enable access to the input buffer for the drive functions
2…255	reserved

Table 4-38 Buffer Clear – Definition

4.3.7 Typical IPM Commanding Sequence



Object Name	Object	User Value [Default Value]
Modes of Operation	0x6060-00	0x07 (Interpolated Position Mode)
Max. Following Error Min. Position Limit Max. Position Limit Max. Profile Velocity Max. Acceleration Profile Velocity Profile Acceleration Quick Stop Deceleration	0x6065-00 0x607D-01 0x607D-02 0x607F-00 0x60C5-00 0x6081-00 0x6083-00 0x6084-00	Application specific [2000 qc] Application specific [−2147483648 qc] Application specific [2147483647 qc] Motor specific [25000 rpm] Application specific [4294967295 rpm/s] Application specific [1000 rpm] Application specific [10000 rpm/s] Application specific [10000 rpm/s
Controlword (Shutdown) Controlword (SwitchOn)	0x6040-00 0x6040-00	0x0006 0x000F
Buffer Clear	0x60C4-06	0x01
Interpolation Data Record	0x20C1-00	Reference points (PVT), minimum 2 points!
Controlword (enable ip mode)	0x6040-00	0x001F
if (Interpolation Buffer Status) **do** Interpolation Data Record **until** (Interpolation Buffer Status)	0x20C4-01 0x20C1-00 0x20C4-01	Bit 0 == 1 (Underflow Warning) Reference point (PVT) Bit 1 == 1 (Overflow Warning)
Interpolation Data Record	0x20C1-00	Reference point (PVT) with time = 0

Table 4-39 Interpolated Position Mode – typical Command Sequence

As long as the interpolation is active, feeding of new reference points is the main task. To minimize the communication overhead, it might make sense to map the “Interpolation Data Record” in a (asynchronous) receive PDO. If the “Interpolation Buffer Status” is mapped to an event trigger transmit PDO (possibly along with the Statusword), processing of reference point feeding can easier be implemented.

4.4 Configuration

4.4.1 Interruption in Case of Error

If a currently running interpolation (index 0x20C4, subindex 0x03 “Interpolation Status” bit 15 “ip mode active” set) will be interrupted by an occurring error, the EPOS3 EtherCAT will react accordingly (i.e. disabling the controller and changing to state switch on disabled).

The interpolation can only be restarted by re-synchronization due to the fact that state “Operation enable” must be entered again, whereby the bit “ip mode active” will be cleared.

••page intentionally left blank••

5 Regulation Tuning

5.1 In Brief

A wide variety of operating modes permit flexible configuration of drive and automation systems by using positioning, speed and current regulation. The built-in EtherCAT interface allows networking to multiple axes drives as well as online commanding by EtherCAT master units.

«Regulation Tuning» is an important attribute of EPOS3 EtherCAT. It is a procedure for automatic start-up of all relevant regulation modes, such as current, velocity and/or positioning control. This intelligent tool is easy to handle and substantially increases the use of the positioning control unit.

5.1.1 Objective

The present Application Note explains use of «Regulation Tuning» and features “in practice examples” suitable for daily use.

Contents

5.1.2 Scope

Hardware	Order #	Firmware Version	Reference
EPOS3 EtherCAT		2200h	Firmware Specification
EPOS3 70/10 EtherCAT	411146	2200h or higher	Cable Starting Set Hardware Reference

Table 5-40 Regulation Tuning – covered Hardware and required Documents

5.1.3 Tools

Tools	Description
Software	«EPOS Studio» Version 2.00 or higher

Table 5-41 Regulation Tuning – recommended Tools

5.2 Regulation Structures

EPOS3 EtherCAT can be interconnected within three essential regulation structures.

5.2.1 Current Control

To provide accurate motion control, given forces and/or torques within the drive system need to be compensated. Hence, EPOS3 EtherCAT offers a current control loop. The current controller is implemented as a PI controller.

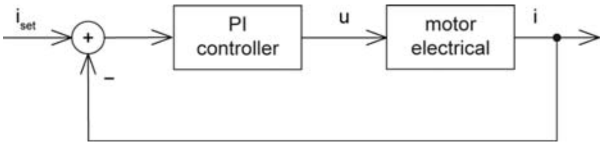


Figure 5-35 Regulation Tuning – Current Control

Current control can be operated either directly as the main regulator, or it serves as subordinated regulator in one of the two following cascade regulation structures.

5.2.2 Velocity Control (with Velocity and Feedforward Acceleration)

Based on the subordinated current control, a velocity control loop can be established. The velocity controller is implemented as a PI controller.

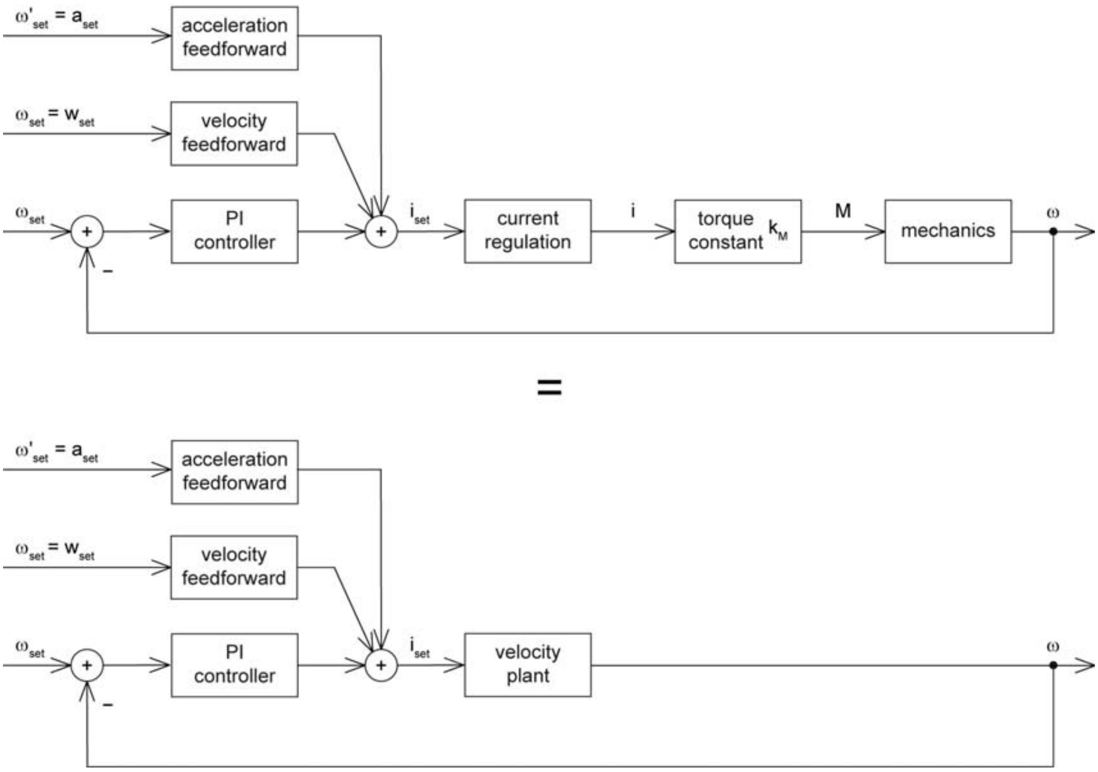


Figure 5-36 Regulation Tuning – Velocity Control

5.2.3 Position Control (with Velocity and Feedforward Acceleration)

Based on the subordinated current control, a position control loop can be established. The position controller is implemented as a PID controller.

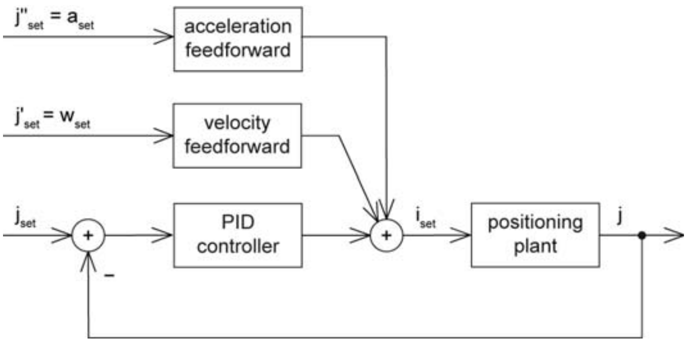


Figure 5-37 Regulation Tuning – Position Control

To improve the reference action of the motion system, position control is supplemented by feedforward control. Velocity feedforward compensates for speed-proportional friction, whereas known inertia can be taken into account by acceleration feedforward.

5.3 Working Principle

«Regulation Tuning» is based on three features:

1) **Identification and modeling** of the plant.
2) **Mapping** model parameters of the plant to derivate controller parameters (PI, PID, feedforward).
3) **Verification** of the resulting regulation structure.

5.3.1 Identification and Modeling

For identification, the plant is activated by a two-point element – positive and negative current of varying amplitudes, which are based on motor parameters – until a stable oscillation of a fixed amplitude is achieved. This experiment is repeated at a different frequency. The characteristics of the oscillations represent substantial properties of the plant.

Hence, the modeling parameters of a simple mathematical model of the plant can be calculated.

5.3.2 Mapping

Now, the model parameters serve for calculation of controller parameters (PI or PID, respectively) and of feedforward velocity and acceleration parameters.

The validity range of the regulation parameters is characterized, among other aspects, by the regulation bandwidth which is determined as well.

5.3.3 Verification

To achieve proper operation with the gained motion control parameters, the system reaction is verified with a motion profile corresponding to the calculated bandwidth.

5.4 Regulation Tuning Wizard

«Regulation Tuning» is a procedure for automated parameterization of the three above mentioned motion controller types (current, velocity and positioning regulation) including position control's feedforward parameters.

For successful Regulation Tuning, correct setup of system parameters in Startup Wizard is essential. Particularly important are…

- Motor data,
- Encoder data, and
- Communication with the PC.

Initiating the "Regulation Tuning Wizard"

1) Complete standard system configuration (Startup Wizard) in «EPOS Studio».
2) Select ¤Wizards¤ and select ¤Regulation Tuning¤.

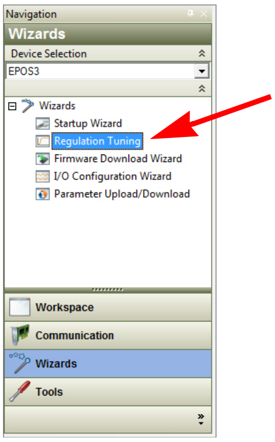


Figure 5-38 Regulation Tuning Wizard

3) Select one of the two modes (for details ➔"Tuning Modes" on page 5-55):
   – ¤Auto Tuning¤
   – ¤Expert Tuning¤.

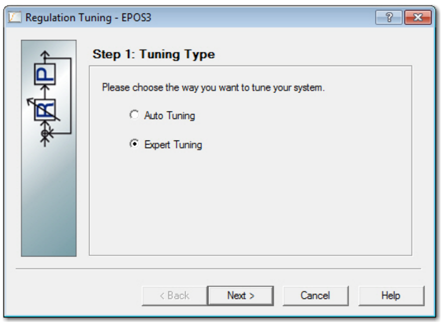


Figure 5-39 Regulation Tuning Mode Selection

5.5 Tuning Modes

5.5.1 Auto Tuning

Auto Tuning is the Regulation Tuning's "very-easy-to-use option". The only thing needed to accomplish automated tuning is to push the start button. A message will inform you that the system will move during the subsequent procedure. Upon confirming the message, Auto Tuning will commence. All required settings are already implemented, so Auto Tuning can parameterize the motion system for most common load cases without further help.

Under certain conditions (strong motor cogging torque, unbalanced friction, low position sensor resolution, etc.) however, or to cover particular requirements (wear, noise or energy optimized operation), Expert Tuning may be used.

5.5.2 Expert Tuning

Expert Tuning offers additional self-describing options for optimum regulation behavior. The following example illustrates tuning using Position Control. Handling of Current Control or Velocity Control however are similar.

Expert Tuning's user interface is divided in four sections:

a) Cascade
b) Identification
c) Parameterization
d) Verification:

Cascade

Provides information on the selected cascade structure.

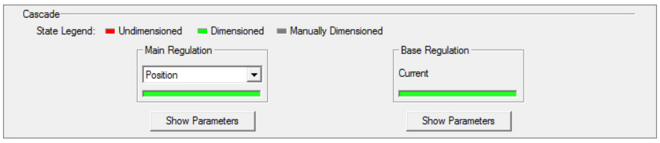


Figure 5-40 Expert Tuning – Cascade

The view is split into two panes; "Main Regulation" and "Base Regulation" (or subordinated regulation). Their respective status is displayed in colored bars:

- Red: Undimensioned – the controller is not yet parameterized.
- Green: Dimensioned – the controller is already parameterized.
- Grey: Manually Dimensioned – the control parameters are being set manually (➔"Manual Tuning" on page 5-57).

Click ¤Show Parameters¤ to view/alter the currently set values.

Velocity control can be viewed and adjusted (in "Main Regulation" window), even if the position was originally defined to be the main controlled variable. However, in order to avoid inconsistencies with the position main regulations, current control cannot be changed. If velocity control's current regulation needs to be optimized, velocity must be defined as Main Regulation variable.

Now, Regulation Tuning is being executed in three steps:

Identification

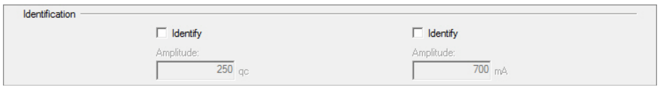


Figure 5-41 Expert Tuning – Identification

Tick ¤Identify¤ if identification of a new plant is necessary (e.g. if the plant properties have changed). In this case, the status of the corresponding controller, as well as all controllers of higher regulation hierarchy, will change to "Undimensioned" (red).

By adjusting the identification amplitude, nonlinear properties (e.g. Coulomb Friction) can be simulated appropriately and can be considered in the plant model by means of harmonic linearization. However, presetting already offers a good basis for plant identification for most applications.

Parameterization

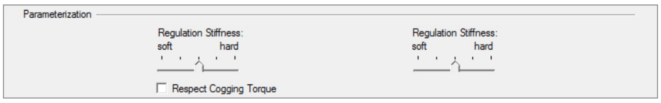


Figure 5-42 Expert Tuning – Parameterization

The calculated controller parameters can be modified to match given requirements by means of sliders:

- "Soft" means: slow regulation behavior, but well dampened.
- "Hard" means: quick regulation behavior, but less dampened.

Tick ¤Respect Cogging Torque¤ to achieve a hard, nevertheless well dampened motion regulation, which brings particular advantages for motors with high cogging torque. In case of unbalanced friction, the regulation behavior can be improved with this adjustment as well.

Verification

The verification of the resulting control system – including feedforward – permits examination of the overall performance. The verification can either take place with a movement profile (which takes bandwidth of the position regulation into account), or a step response. As interesting feature; in addition to the position, the corresponding current is recorded, too.

To zoom the recorded diagrams, crop the "area of interest" and click right.

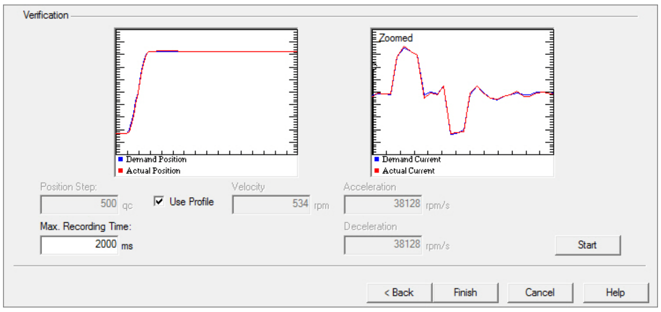


Figure 5-43 Expert Tuning – Verification

The parameters "Position Step", "Velocity", "Acceleration" and "Deceleration" are computed automatically. They can be adjusted only if the positioning controller is in state "Manually Dimensioned" (grey).

The parameter "Max. Recording Time" limits the time interval for data acquisition. This can be useful, if details concerning the beginning of the movement profile are of interest.

¤Start¤ launches Expert Tuning. ¤Finish¤ will save the obtained feedback and feedforward parameters in the EPOS3 EtherCAT and make them valid for all operation modes. ¤Cancel¤ will reject the results and returns to the starting situation.

5.5.3 Manual Tuning

In certain conditions, you might wish to change control parameters manually to see how the system reacts without performing automated system identification and modeling.

Also, the manual mode can be used…

- for fine tuning and optimization in very demanding applications, or
- if the outcome of Auto Tuning/Expert Tuning is not satisfactory.

Initiate Manual Tuning by selecting ¤Manually Dimensioned¤ in ¤Show Parameter¤ dialog (➔“Cascade” on page 5-55). As a result, the status will switch to “Manually Dimensioned” (grey), thus neither automated identification nor parameterization will be carried out. In addition, you can define the motion profile (➔“Verification” on page 5-56).

After ticking ¤Identify¤, or if you make any changes (➔“Parameterization” on page 5-56), Manual Tuning is terminated showing status “Undimensioned” (red).

••page intentionally left blank••

6 Device Programming

6.1 In Brief

A wide variety of operating modes permit flexible configuration of drive and automation systems by using positioning, speed and current regulation. The built-in EtherCAT interface allows networking to multiple axes drives as well as online commanding by EtherCAT master units.

6.1.1 Objective

The present Application Note explains typical commanding sequences for different operating modes. The explanations are based on writing/reading commands to access the Object Dictionary. For detailed information on the objects itself ➔separate document «EPOS3 EtherCAT Firmware Specification» (subsequently referred to as “FwSpec”). For detailed information on the command structure ➔«EPOS Studio» (command analyzer). For motor-specific data ➔maxon Catalog.

Contents

6.1.2 Scope

Hardware	Order #	Firmware Version	Reference
EPOS3 EtherCAT		2200h	Firmware Specification
EPOS3 70/10 EtherCAT	411146	2200h or higher	

Table 6-42 Device Programming – covered Hardware and required Documents

6.1.3 Tools

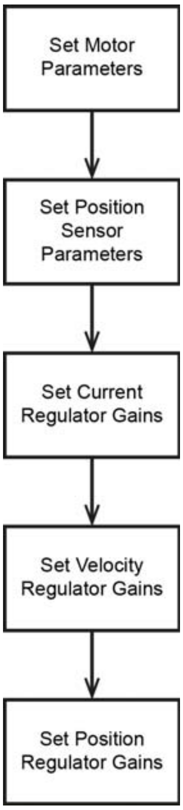
Tools	Description
Software	«EPOS Studio» Version 2.00 or higher

Table 6-43 Device Programming – recommended Tools

6.2 First Step

Before the motor will be activated, motor parameters, position sensor parameters and regulation gains must be set. For detailed description ➔FwSpec.

Note
For detailed information on the command structure ➔«EPOS Studio» (command analyzer).



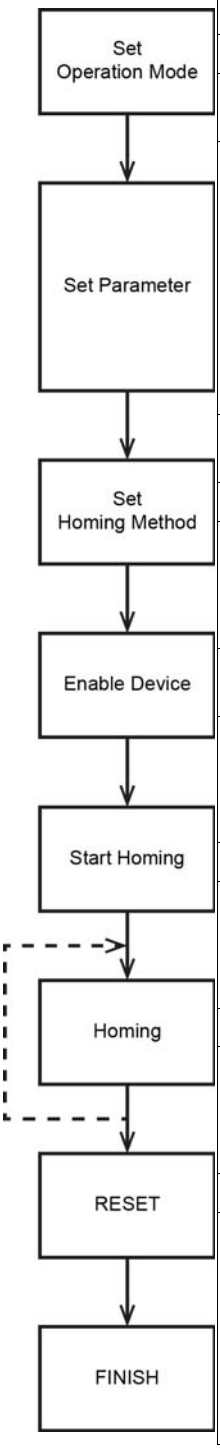
Object Name	Object	User Value [Default Value]
Motor Type Continuous Current Limit Pole Pair Number Thermal Time Constant Winding	0x6402-00 0x6410-01 0x6410-03 0x6410-05	Motor-specific [10] Motor-specific [5000] Motor-specific [1] Motor-specific [40]
Encoder Pulse Number Position Sensor Type	0x2210-01 0x2210-02	Sensor-specific [500] Sensor-specific [1]
Current Regulator P-Gain Current Regulator I-Gain	0x60F6-01 0x60F6-02	Motor-specific. Determine optimal parameter using “Regulation Tuning” in «EPOS Studio».
Speed Regulator P-Gain Speed Regulator I-Gain	0x60F9-01 0x60F9-02	Motor-specific. Determine optimal parameter using “Regulation Tuning” in «EPOS Studio».
Position Regulator P-Gain Position Regulator I-Gain Position Regulator D-Gain	0x60FB-01 0x60FB-02 0x60FB-03	Motor-specific. Determine optimal parameter using “Regulation Tuning” in «EPOS Studio».

Table 6-44 Device Programming – First Step

6.3 Homing Mode

6.3.1 Start Homing

The axis references to an absolute position using the selected homing method.



Object Name	Object	User Value [Default Value]
Modes of Operation	0x6060-00	0x06 (Homing Mode)
Max. Following Error Home Offset Max. Profile Velocity Quick Stop Deceleration Speed for Switch Search Speed for Zero Search Homing Acceleration Current Threshold Homing Mode Home Position	0x6065-00 0x607C-00 0x607F-00 0x6085-00 0x6099-01 0x6099-02 0x609A-00 0x2080-00 0x2081-00	User-specific [2000 qc] User-specific [0 qc] Motor-specific [25000 rpm] User-specific [10000 rpm/s] User-specific [100 rpm] User-specific [10 rpm] User-specific [1000 rpm/s] User-specific [500 mA] User-specific [0 qc]
Homing Method	0x6098-00	Select Homing Method (➔FwSpec)
Controlword (Shutdown) Controlword (Switch-on)	0x6040-00 0x6040-00	0x0006 0x000F
Controlword (Start homing mode)	0x6040-00	0x001F
Statusword (Homing attained)	0x6041-00	wait until bit 12 is set to “1”
Controlword (Switch-on)	0x6040-00	0x000F

Table 6-45 Device Programming – Homing Mode (Start)

6.3.2 Read Status

Read Statusword

Object Name	Object	User Value [Default Value]
Statusword (Homing attained)	0x6041-00	Home position is reached if bit 12 is set to “1”.

Table 6-46 Device Programming – Homing Mode (Read)

6.3.3 Stop Positioning

Stop Homing

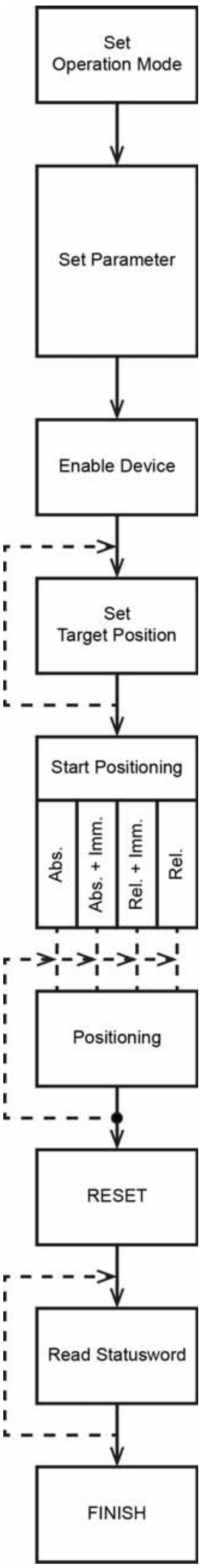
Object Name	Object	User Value [Default Value]
Controlword (Switch-on)	0x6040-00	0x000F
or		
Controlword (Halt homing)	0x6040-00	0x011F
or		
Controlword (Quick stop)	0x6040-00	0x000B

Table 6-47 Device Programming – Homing Mode (Stop)

6.4 Profile Position Mode

6.4.1 Set Position

The axis moves to an absolute or relative position using a motion profile (➔next page).



Object Name	Object	User Value [Default Value]
Modes of Operation	0x6060-00	0x01 (Profile Position Mode)
Max. Following Error Min. Position Limit Max. Position Limit Max. Profile Velocity Profile Velocity Profile Acceleration Profile Deceleration Quick Stop Deceleration Motion Profile Type	0x6065-00 0x607D-01 0x607D-02 0x607F-00 0x6081-00 0x6083-00 0x6084-00 0x6085-00 0x6086-00	User-specific [2000 qc] User-specific [−2147483648 qc] User-specific [2147483647 qc] Motor-specific [25000 rpm] Desired Velocity [1000 rpm] User-specific [10000 rpm/s] User-specific [10000 rpm/s] User-specific [10000 rpm/s] User-specific [0]
Controlword (Shutdown) Controlword (Switch-on)	0x6040-00 0x6040-00	0x0006 0x000F
Target Position	0x607A-00	Desired Position [qc]
Controlword (absolute pos.) or Controlword (absolute pos., start immediately) or Controlword (relative pos., start immediately) or Controlword (relative positioning)	0x6040-00 0x6040-00 0x6040-00 0x6040-00	0x001F 0x003F 0x007F 0x005F
Statusword (set point acknowledge)	0x6041-00	wait until bit 12 is set to “1”
Controlword (Bit 12 set point acknowledge will be reset)	0x6040-00	0x000F
Statusword (Target reached)	0x6041-00	wait until bit 10 is set to “1”

Table 6-48 Device Programming – Profile Position Mode (Set)

6.4.2 Read Status

Read Statusword

Object Name	Object	User Value [Default Value]
Statusword (Target reached)	0x6041-00	The axis is at target position if bit 10 is set.

Table 6-49 Device Programming – Profile Position Mode (Read)

6.4.3 Stop Positioning

Stop Positioning

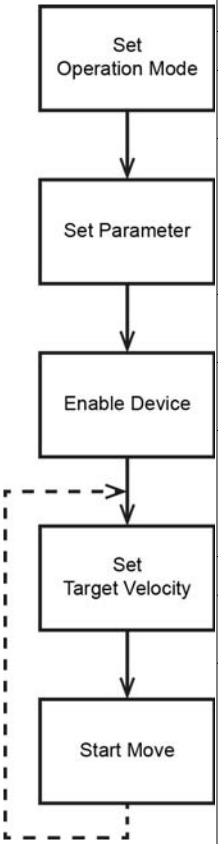
Object Name	Object	User Value [Default Value]
Controlword (Stop positioning) or Controlword (Quick stop)	0x6040-00 0x6040-00	0x010F 0x000B

Table 6-50 Device Programming – Profile Position Mode (Stop)

6.5 Profile Velocity Mode

6.5.1 Start Velocity

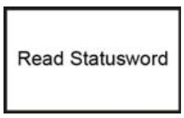
Motor shaft rotates with a certain speed with velocity profile.



Object Name	Object	User Value [Default Value]
Modes of Operation	0x6060-00	0x03 (Profile Velocity Mode)
Max. Profile Velocity Profile Acceleration Profile Deceleration Quick Stop Deceleration Motion Profile Type	0x607F-00 0x6083-00 0x6084-00 0x6085-00 0x6086-00	Motor-specific [25000 rpm] User-specific [10000 rpm/s] User-specific [10000 rpm/s] User-specific [10000 rpm/s] User-specific [0]
Controlword (Shutdown) Controlword (Switch-on)	0x6040-00 0x6040-00	0x0006 0x000F
Target Velocity	0x60FF-00	Velocity for movement [rpm]
Controlword	0x6040-00	0x000F

Table 6-51 Device Programming – Profile Velocity Mode (Start)

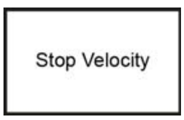
6.5.2 Read Status



Object Name	Object	User Value [Default Value]
Statusword (Target velocity reached)	0x6041-00	Target velocity is reached if bit 10 is set.

Table 6-52 Device Programming – Profile Velocity Mode (Read)

6.5.3 Stop Velocity



Object Name	Object	User Value [Default Value]
Controlword (Halt Profile Velocity Mode) or Controlword (Quick stop)	0x6040-00 0x6040-00	0x010F 0x000B

Table 6-53 Device Programming – Profile Velocity Mode (Stop)

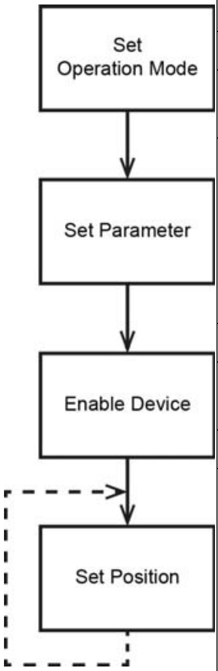
6.6 Interpolated Position Mode (PVT)

For detailed information ➔chapter “4 Interpolated Position Mode” on page 4-35.

6.7 Cyclic Synchronous Position (CSP)

6.7.1 Set Position

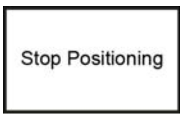
The axis moves to the new absolute position. If the difference between actual and new position is greater than “Max Following Error”, an emergency procedure will be launched.



Object Name	Object	User Value [Default Value]
Modes of Operation	0x6060-00	0x08 (Cyclic Synchronous Position)
Max. Following Error Min. Position Limit Max. Position Limit Max. Motor Speed Nominal Current Motor Torque Constant	0x6065-00 0x607D-01 0x607D-02 0x6410-04 0x6410-01 0x6410-06	User-specific [2000 qc] User-specific [−2147483648 qc] User-specific [2147483647 qc] Motor-specific Motor-specific (optional) Motor-specific (optional)
Controlword (Shutdown) Controlword (Switch-on)	0x6040-00 0x6040-00	0x0006 0x000F
Target Position Position Offset Velocity Offset Torque Offset	**0x607A-00** 0x60B0-00 0x60B1-00 0x60B2-00	**New Position [Position units]** [Position units] (optional) [Velocity units] (optional) [‰/rated torque] (optional)

Table 6-54 Device Programming – Cyclic Synchronous Position (Set)

6.7.2 Stop Positioning



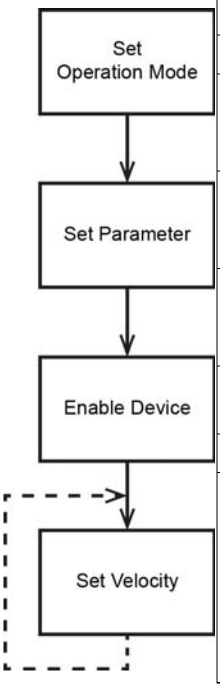
Object Name	Object	User Value [Default Value]
Controlword (Quick stop)	0x6040-00	0x000B

Table 6-55 Device Programming – Cyclic Synchronous Position (Stop)

6.8 Cyclic Synchronous Velocity (CSV)

6.8.1 Set Velocity

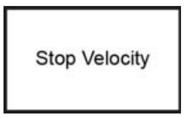
Motor shaft runs with a certain speed.



Object Name	Object	User Value [Default Value]
Modes of Operation	0x6060-00	0x09 (Cyclic Synchronous Velocity)
Max. Motor Speed Nominal Current Motor Torque Constant	**0x6410-04** 0x6410-01 0x6410-06	**Motor-specific** Motor-specific (optional) Motor-specific (optional)
Controlword (Shutdown) Controlword (Switch-on)	0x6040-00 0x6040-00	0x0006 0x000F
Target Velocity Velocity Offset Torque Offset	**0x60FF-00** 0x60B1-00 0x60B2-00	**Velocity per movement [Velocity units]** [Velocity units] (optional) [‰/rated torque] (optional)

Table 6-56 Device Programming – Cyclic Synchronous Velocity (Set)

6.8.2 Stop Velocity



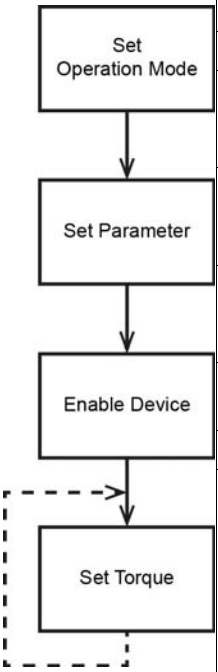
Object Name	Object	User Value [Default Value]
Controlword (Quick stop)	0x6040-00	0x000B

Table 6-57 Device Programming – Cyclic Synchronous Velocity (Stop)

6.9 Cyclic Synchronous Torque (CST)

6.9.1 Set Torque

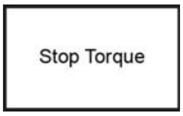
This command applies a certain torque at the motor.



Object Name	Object	User Value [Default Value]
Modes of Operation	0x6060-00	0x0A (Cyclic Synchronous Torque)
Nominal Current Nominal Motor Speed Motor Torque Constant	0x6410-01 0x6410-04 0x6410-06	Motor-specific Motor-specific Motor-specific
Controlword (Shutdown) Controlword (Switch-on)	0x6040-00 0x6040-00	0x0006 0x000F
Target Torque Torque Offset	**0x6071-00** 0X60B2-00	**[‰/rated torque]** [‰/rated torque] (optional)

Table 6-58 Device Programming – Cyclic Synchronous Torque (Set)

6.9.2 Stop Motion



Object Name	Object	User Value [Default Value]
Controlword (Quick stop)	0x6040-00	0x000B

Table 6-59 Device Programming – Cyclic Synchronous Torque (Stop)

6.10 State Machine

6.10.1 Clear Fault

Resetting “Fault” condition sends the Controlword with value 0x0080.

Clear Fault

Object Name	Object	User Value [Default Value]
Controlword (Fault Reset)	0x6040-00	0x0080

Table 6-60 Device Programming – State Machine (Clear Fault)

6.11 Motion Info

6.11.1 Get Movement State

Read Statusword

Object Name	Object	User Value [Default Value]
Read Statusword	0x6041-00	Bit 10 tells states that target is reached. For details ➔FwSpec.

Table 6-61 Device Programming – Motion Info (Get Movement State)

6.11.2 Read Position

Read Position

Object Name	Object	User Value [Default Value]
Position actual value	0x6064-00	[Position units]

Table 6-62 Device Programming – Motion Info (Read Position)

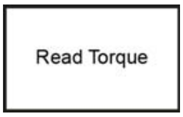
6.11.3 Read Velocity

Read Velocity

Object Name	Object	User Value [Default Value]
Velocity actual value	0x 606C-00	[Velocity units]

Table 6-63 Device Programming – Motion Info (Read Velocity)

6.11.4 Read Current



Object Name	Object	User Value [Default Value]
Torque actual value	0x6077-00	[‰/rated torque]

Table 6-64 Device Programming – Motion Info (Read Current)

6.12 Utilities

6.12.1 Store all Parameters

Saves all parameters.

Object Name	Object	User Value [Default Value]
Save All Parameters	0x10101-01	0x65766173 “save”

Table 6-65 Device Programming – Utilities (Store all Parameters)

6.12.2 Restore all default Parameters

Restores all parameters to factory settings.

Object Name	Object	User Value [Default Value]
Restore All Default Parameters	0x1011-01	0x64616F6C “load”

Table 6-66 Device Programming – Utilities (Restore all default Parameters)

7 Controller Architecture

7.1 In Brief

A wide variety of operating modes permit flexible configuration of drive and automation systems by using positioning, speed and current regulation. The built-in EtherCAT interface allows networking to multiple axes drives as well as online commanding by EtherCAT master units.

In addition to the standard EPOS3 EtherCAT PID position control, also feedforward compensation is available. The feedforward compensation provides faster setpoint following in applications with higher load inertia and accelerations and/or in applications with considerable speed-dependent load (as with friction-afflicted drives). With EPOS3 EtherCAT Positioning Controllers, dual loop regulation is available.

7.1.1 Objective

The present Application Note explains the EPOS3 EtherCAT controller architecture. Furthermore explained will be mapping of internal controller parameters to controller parameters in SI units, and vice versa.

In addition to PID position regulation, the functionalities of built-in acceleration and velocity feedforward are described. Their advantages, compared to simple PID control are shown using two “in practice examples”.

Contents

7.1.2 Scope

Hardware	Order #	Firmware Version	Reference
EPOS3 EtherCAT		2200h	Firmware Specification
EPOS3 70/10 EtherCAT	411146	2200h or higher	

Table 7-67 Controller Architecture – covered Hardware and required Documents

7.1.3 Tools

Tools	Description
Software	«EPOS Studio» Version 2.00 or higher

Table 7-68 Controller Architecture – recommended Tools

7.2 Overview

The EPOS3 EtherCAT controller architecture contains three built-in control loops.

- Current regulation is used in all modes.
- Position and velocity controllers are only used in position-based, respectively velocity-based modes.
- Current control loop receives as input the position, respectively velocity controller's output.

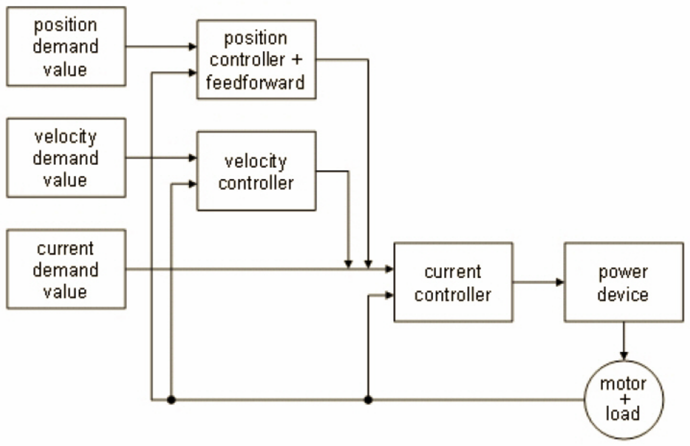


Figure 7-44 Controller Architecture

7.3 Regulation Methods

7.3.1 Current Regulation

During a movement within a drive system, forces and/or torques must be controlled. Therefore, as a principal regulation structure, EPOS3 EtherCAT offers current-based control.

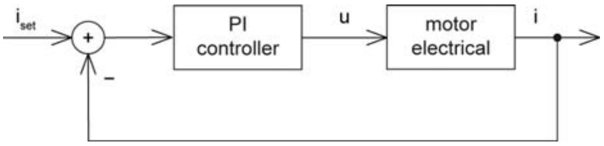


Figure 7-45 Controller Architecture – Current Regulator

Constants

Sampling period: T_s = 100 µs

Object Dictionary Entries

Symbol	Name	Index	Subindex
K_{P_EPOS3}	Current Regulator P-Gain	0x60F6	0x01
K_{I_EPOS3}	Current Regulator I-Gain	0x60F6	0x02

Table 7-69 Current Regulation – Object Dictionary

Conversion of PI Controller Parameters (EPOS3 to SI Units)

$$K_{P\ldots SI} = \frac{1\Omega}{2^8} \cdot K_{P\ldots EPOS3} = 3.91m\Omega \cdot K_{P\ldots EPOS3}$$

$$K_{I\ldots SI} = \frac{1\Omega}{2^8 T_s} \cdot K_{I\ldots EPOS3} = 39.1\frac{\Omega}{s} \cdot K_{I\ldots EPOS3}$$

Current controller parameters in SI units can be used in analytical calculations, respectively numerical simulations via transfer function:

$$C_{current}(s) = K_{P\ldots SI} + \frac{K_{I\ldots SI}}{s}$$

7.3.2 Velocity Regulation (with Feedforward)

Based on the subordinated current control, EPOS3 EtherCAT also offers velocity regulation.

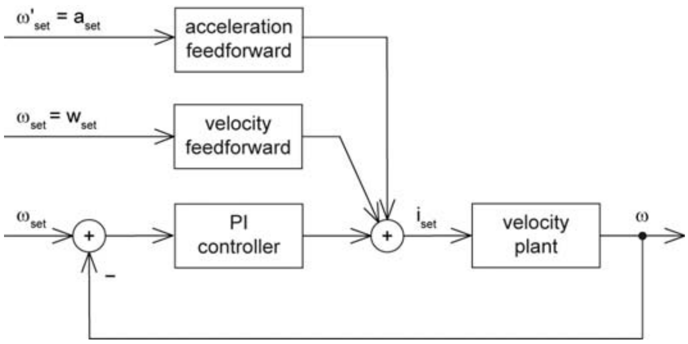


Figure 7-46 Controller Architecture – Velocity Regulator

Constants

Sampling period: T_s = 1 ms

Object Dictionary Entries

Symbol	Name	Index	Subindex
K_{P_EPOS3}	Speed Regulator P-Gain	0x60F9	0x01
K_{I_EPOS3}	Speed Regulator I-Gain	0x60F9	0x02
K_{ω_EPOS3}	Velocity Feedforward Factor in Speed Regulator	0x60F9	0x04
K_{α_EPOS3}	Acceleration Feedforward Factor in Speed Regulator	0x60F9	0x05

Table 7-70 Velocity Regulation – Object Dictionary

Conversion of PI Controller Parameters (EPOS3 to SI Units)

$$K_{P...SI} = 20\frac{\mu A}{(rad)/s} \cdot K_{P...EPOS3}$$

$$K_{I...SI} = 5\frac{(mA)/s}{(rad)/s} \cdot K_{I...EPOS3}$$

Velocity controller parameters in SI units can be used in analytical calculations, respectively numerical simulations via transfer function:

$$C_{velocity}(s) = K_{P...SI} + \frac{K_{I...SI}}{s}$$

Conversion of Feedforward Parameters (EPOS3 to SI Units)

Velocity feedforward: $K_{\omega...SI} = 1\frac{\mu A}{(rad)/s} \cdot K_{\omega...EPOS3}$

Acceleration feedforward: $K_{\alpha...SI} = 1\frac{\mu A}{(rad)/s^2} \cdot K_{\alpha...EPOS3}$

7.3.3 Position Regulation (with Feedforward)

Based on the subordinated current control, EPOS3 EtherCAT is able to close a positioning control loop.

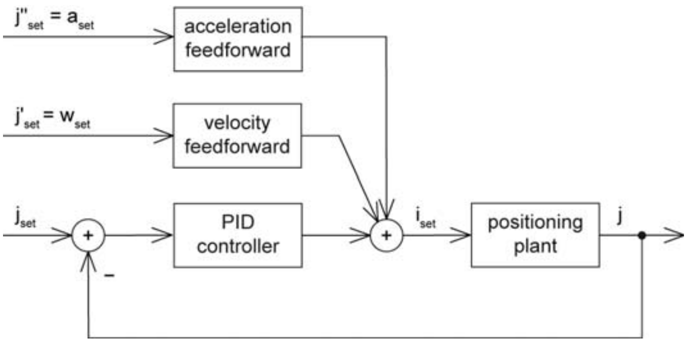


Figure 7-47 Controller Architecture – Position Regulator with Feedforward

Constants

Sampling period: T_s = 1 ms

Object Dictionary Entries

Symbol	Name	Index	Subindex
K_{P_EPOS3}	Position Regulator P-Gain	0x60FB	0x01
K_{I_EPOS3}	Position Regulator I-Gain	0x60FB	0x02
K_{D_EPOS3}	Position Regulator D-Gain	0x60FB	0x03
K_{ω_EPOS3}	Velocity Feedforward Factor in Position Regulator	0x60FB	0x04
K_{α_EPOS3}	Acceleration Feedforward Factor in Position Regulator	0x60FB	0x05

Table 7-71 Position Regulation with Feedforward – Object Dictionary

The position controller is implemented as PID controller. To improve the motion system's setpoint following, positioning regulation is supplemented by feedforward control. Thereby, velocity feedforward serves for compensation of speed-proportional friction, whereas acceleration feedforward considers known inertia.

Conversion of PI Controller Parameters (EPOS3 to SI Units)

$$K_{P...SI} = 10\frac{mA}{rad} \cdot K_{P...EPOS3}$$

$$K_{I...SI} = 78\frac{(mA)/s}{rad} \cdot K_{I...EPOS3}$$

$$K_{D...SI} = 80\frac{\mu As}{rad} \cdot K_{D...EPOS3}$$

Position controller parameters in SI units can be used in analytical calculations, respectively numerical simulations via transfer function:

$$C_{position}(s) = K_{P...SI} + \frac{K_{I...SI}}{s} + \frac{K_{D...SI}S}{1 + \frac{K_{D...SI}}{16K_{P...SI}}s}$$

Conversion of Feedforward Parameters (EPOS3 to SI Units)

Velocity feedforward: $K_{\omega...SI} = 1\frac{\mu A}{(rad)/s} \cdot K_{\omega...EPOS3}$

Acceleration feedforward: $K_{\alpha...SI} = 1\frac{\mu A}{(rad)/s^2} \cdot K_{\alpha...EPOS3}$

7.3.4 Operation Modes with Feedforward

Acceleration and velocity feedforward have an effect in «Profile Position Mode», «Profile Velocity Mode» and «Homing Mode». All other operating modes are not influenced.

7.3.4.1 Purpose of Velocity Feedforward

Velocity feedforward provides additional current in cases, where the load increases with speed, such as speed-dependent friction. The load is assumed to increase proportional with speed. The optimal velocity feedforward parameter in SI units is…

$$K_{\omega...SI} = \frac{r}{k_M}$$

Meaning: With given total friction proportional factor “r” relative to the motor shaft, and the motor’s torque constant “k_M”, you ought to adjust the velocity feedforward parameter to…

$$K_{\omega...EPOS3} = \frac{r}{k_M} \cdot \frac{(rad)/s}{1\mu A} = \frac{r}{k_M} \cdot \frac{10^6(rad)/s}{A}$$

7.3.4.2 Purpose of Acceleration Feedforward

Acceleration feedforward provides additional current in cases of high acceleration and/or high load inertias. The optimal acceleration feedforward parameter in SI units is…

$$K_{\alpha...SI} = \frac{J}{k_M}$$

Meaning: With given total inertia “J” relative to the motor shaft, and the motor’s torque constant “k_M”, you ought to adjust the acceleration feedforward parameter to…

$$K_{\alpha...EPOS3} = \frac{J}{k_M} \cdot \frac{(rad)/s^2}{1\mu A} = \frac{J}{k_M} \cdot \frac{10^6(rad)/s^2}{A}$$

7.4 Regulation Tuning

maxon motor’s «EPOS Studio» features «Regulation Tuning» as powerful wizard allowing to automatically tune all controller and feedforward parameters described above for most drive systems within a few minutes. For details ➔chapter “5 Regulation Tuning” on page 5-51.

7.5 Dual Loop Regulation

In many applications it is common to use gears to increase motor torque, or screw spindles to transform motor rotation into linear movement. The gear itself is made of a lot of different parts, such as, belts, pinions, pulleys, spindles, etc.

The associated elasticity and backlash of these parts create an effect of compliance and as well as a delay in the drive chain. Often, the mechanical transmission between motor and load has some backlash, too, resulting in a certain “delay” being introduced to the plant. This delay influences the regulation stability and may have such big impact that one may be forced to reduce the dynamic behavior or the precision of the drive.

To overcome these limitations and to combine a motor/gear system with a precise and high dynamic regulation, it will be necessary to control the motor movement as well as the load movement. This results in a new control structure called “dual loop”, featuring two individual encoders – one directly mounted to the motor, the another mounted at the gear or linear slide or directly on/near to the load.

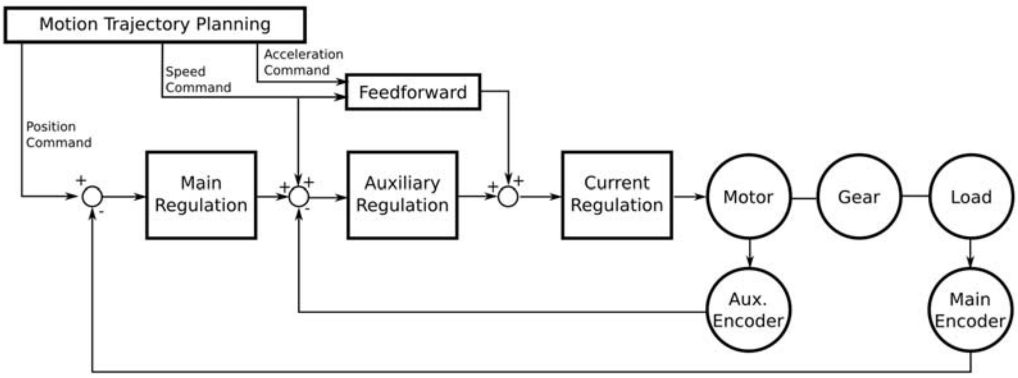


Figure 7-48 Dual Loop Architecture

The auxiliary regulation is designed to provide damping and dynamic system behavior while the main regulation generates the desired position precision.

General Rule
With Dual Loop Regulation, the following general restriction applies:

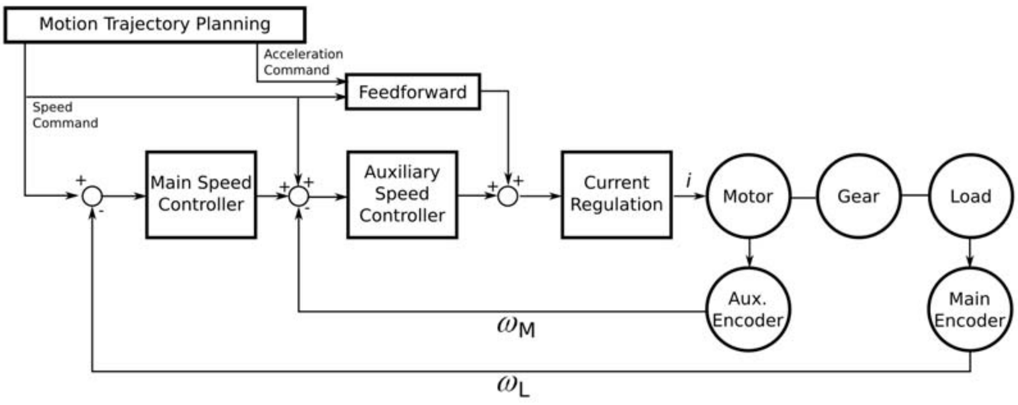
$$AuxEncoderResolution \cdot GearRatio \leq MainEncoderResolution$$

7.5.1 Current Regulation

The dual loop current controller is implemented similar to the current controller in a single loop system. For details ➔chapter “7.3.1 Current Regulation” on page 7-73.

7.5.2 Velocity Regulation (with Feedforward)

The design is based on current regulation.



ω_M motor speed
ω_L load speed

Figure 7-49 Dual Loop Velocity Regulation

In velocity regulation, the auxiliary controller appropriately stabilizes the loop; however, the main controller provides the correct speed feedback.

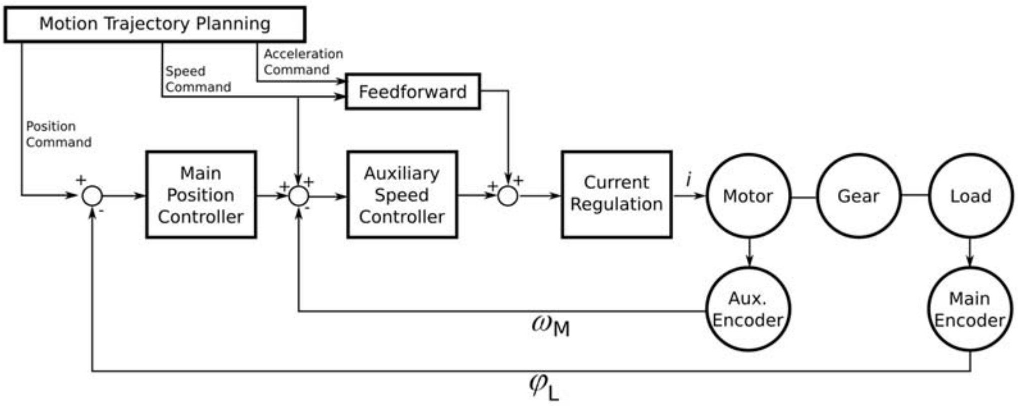
The dual loop velocity controller (that is main controller and auxiliary controller together) is implemented as PI controller.

Conversion parameters

Conversion of PI controller and feedforward parameters in dual loop (EPOS3 to SI units) are identical to that in single loop (➔chapter "7.3.2 Velocity Regulation (with Feedforward)" on page 7-74).

7.5.3 Position Regulation (with Feedforward)

The design is based on current regulation.



ω_M motor speed
φ_L load position

Figure 7-50 Dual Loop Position Regulation

In position regulation, the auxiliary controller is designed to stabilize the loop, whereas the main controller provides the correct position feedback.

The dual loop position controller (that is main controller and auxiliary controller together) is realized as PID controller and features the same sampling period as the dual loop velocity controller.

Conversion parameters3

Conversion of PI controller and feedforward parameters in dual loop (EPOS3 to SI units) are identical to that in single loop (➔chapter "7.3.3 Position Regulation (with Feedforward)" on page 7-75).

7.5.4 Conclusion

The dual loop topology is adequate if the ratio of motor inertia and load inertia is not too large. The drive elements (motor, gear, encoders, load) must be dimensioned correctly.

General Selection Practice

To achieve reliability of the system, follow the scheme below to determine the individual components:

- **Motor**
  Chose a motor capable to fulfill the load's requirements for maximum torque, continuous torque, and speed. For detailed information ➔chapter "1.6 Sources for additional Information" on page 1-10, item [7].
- **Gear**
  Chose a gear capable to fulfill the load's torque and speed range. Boundary conditions are maximum motor load, maximum gear load, and the associated speed limits.
  Another influence that might need consideration is the minimum motor heat dissipation capability. Use the following formula to determine the optimum gear ratio:

  $$I = \sqrt{\frac{Jl}{Jm}}$$

  Jl load inertia
  Jm motor inertia

- **Motor Encoder**
  Chose a motor encoder capable to provide sufficient stiffness in the inner loop. A few hundred increments per revolution as the motor encoder's minimum resolution are recommended.
- **Load Encoder**
  Chose a load encoder capable to at least deliver the required resolution and accuracy on the load side.

7.5.5 Auto Tuning

The dual loop start up is similar to the start up of the single loop regulation and can be described with the following major steps:

1) Identification and modeling of the plant.
2) Calculation of all controller parameters (current, auxiliary, main, feedforward).
3) Mapping; the calculated controller parameters (main, auxiliary) are mathematically transformed to PI controller parameters (for velocity regulation) or to PID controller parameters (for position regulation).
4) Verification; the system's dynamic response is measured and displayed using the scope function in «EPOS Studio». This allows verification, whether the system behavior is as expected.

7.6 Application Examples

Please find below two “in practice examples” suitable for daily use.

For comparability and validity reasons, the measured simulation results are converted to the units “mA”, “rpm” and “qc”!

7.6.1 Example 1: System with high Inertia and low Friction

System Components

Item	Description	Setting
Controller EPOS3 70/10 EtherCAT (411146)		
Motor maxon EC 40 (118896)	No load speed (line 2)	n_0 = 10’400 rpm
	No load current (line 3)	I_0 = 258 mA
	Nominal current (line 6)	I_n = 3.4 A
	Resistance phase to phase (line 10)	R = 1.25 Ω
	Inductance phase to phase (line 11)	L = 0.319 mH
	Torque constant (line 12)	k_M = 38.2 mNm/A
	Rotor inertia (line 16)	J_{motor} = 85 gcm²
Encoder HEDL 5540 (110516)	Encoder pulse number	500
Mechanical load Fly wheel	Inertia	J_{load} = 5000 gcm²

Table 7-72 Controller Architecture – Example 1: Components

Model of the Plant

The following parameters can be deduced:

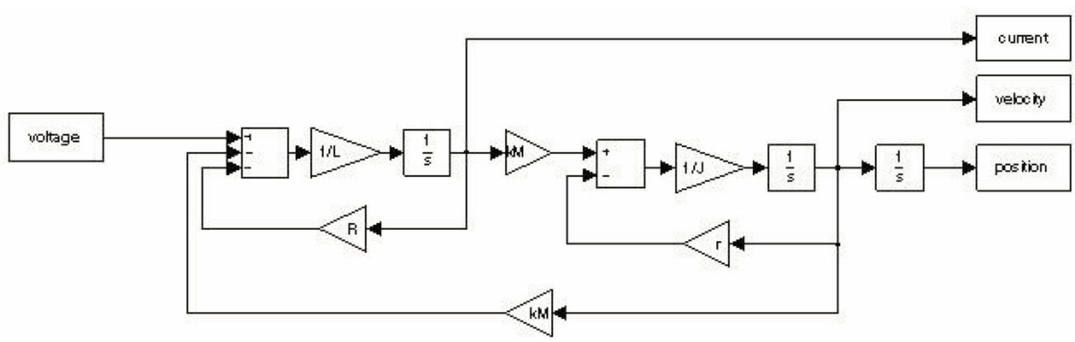


Figure 7-51 Example1 – Block Diagram

Electrical Part

R = 1.25 Ω

L = 0.319 mH

Interface between electrical and mechanical Parts

$$k_M = 38.2\frac{mNm}{A}$$

Mechanical Part

$$J = J_{motor} + J_{load} = 5085gcm^2$$

$$r = \frac{k_M I_0}{n_0 \frac{2\pi rad}{1} \cdot \frac{1min}{60s}} = \frac{9.86mNm}{1089rad^2} = 9.05\frac{\mu Nm}{(rad)/s}$$

- Input is the voltage at the motor winding.
- Outputs are current, velocity or position.

Regulation Tuning as to the described conditions results in the following controller and feedforward parameters:

Object Dictionary Access — EPOS3 is disabled

Active Object Filter: System Parameter

Index	SubIndex	Name	Type	Access	Value
0x2008	0x00	Miscellaneous Configuration	UInt16	RW	0
0x2210		Sensor Configuration			
0x2210	0x01	Pulse Number Incremental Encoder 1	UInt32	RW	500
0x2210	0x02	Position Sensor Type	UInt16	RW	1
0x2210	0x04	Position Sensor Polarity	UInt16	RW	0
0x6065	0x00	Max Following Error	UInt32	RW	200000
0x60F6		Current Control Parameter Set			
0x60F6	0x01	Current Regulator P-Gain	Int16	RW	434
0x60F6	0x02	Current Regulator I-Gain	Int16	RW	105
0x60F9		Velocity Control Parameter Set			
0x60F9	0x01	Speed Regulator P-Gain	Int16	RW	21983
0x60F9	0x02	Speed Regulator I-Gain	Int16	RW	747
0x60F9	0x04	Velocity Feedforward Factor in Speed Regulator	UInt16	RW	0
0x60F9	0x05	Acceleration Feedforward Factor in Speed Regulator	UInt16	RW	13061
0x60FB		Position Control Parameter Set			
0x60FB	0x01	Position Regulator P-Gain	Int16	RW	1120
0x60FB	0x02	Position Regulator I-Gain	Int16	RW	812
0x60FB	0x03	Position Regulator D-Gain	Int16	RW	8244
0x60FB	0x04	Velocity Feedforward Factor in Position Regulator	UInt16	RW	0
0x60FB	0x05	Acceleration Feedforward Factor in Position Regulator	UInt16	RW	13061
0x6402	0x00	Motor Type	UInt16	RW	1
0x6410		Motor Data			
0x6410	0x01	Nominal Current	UInt16	RW	1950
0x6410	0x02	Output Current Limit	UInt16	RW	3900
0x6410	0x03	Pole Pair Number	UInt8	RW	1
0x6410	0x04	Maximal Motor Speed	UInt32	RW	12000
0x6410	0x05	Thermal Time Constant Winding	UInt16	RW	300

Figure 7-52 Example1 – System Parameters, real

For numerical simulation, the conversion results from EPOS3 to SI units are as follows:

Current Controller

$$K_{P...EPOS3} = 434 \quad \rightrightarrows \quad K_{P...SI} = 1.70\Omega$$

$$K_{I...EPOS3} = 105 \quad \rightrightarrows \quad K_{I...SI} = 4.11\frac{k\Omega}{s}$$

Velocity Controller

$$K_{P...EPOS3} = 21983 \quad \rightrightarrows \quad K_{P...SI} = 0.440\frac{A}{(rad)/s}$$

$$K_{I...EPOS3} = 747 \quad \rightrightarrows \quad K_{I...SI} = 3.74\frac{A/s}{(rad)/s}$$

Position Controller

$$K_{P...EPOS3} = 1120 \quad \rightrightarrows \quad K_{P...SI} = 11.2\frac{A}{rad}$$

$$K_{I...EPOS3} = 812 \quad \rightrightarrows \quad K_{I...SI} = 63.2\frac{A/s}{rad}$$

$$K_{D...EPOS3} = 8244 \quad \rightrightarrows \quad K_{D...SI} = 0.660\frac{As}{rad}$$

Positioning and Velocity Feedforward

$$K_{\omega...EPOS3} = 0 \quad \rightrightarrows \quad K_{\varpi...SI} = 0\frac{A}{(rad)/s}$$

$$K_{\alpha...EPOS3} = 13061 \quad \rightrightarrows \quad K_{\alpha...SI} = 13.06\frac{mA}{(rad)/s^2}$$

Plausibility Check

$$K_{\omega...SI} = \frac{r}{k_M} = 237\frac{\mu A}{(rad)/s} \quad (\rightrightarrows) \quad K_{\varpi...SI} = 237\frac{\mu A}{(rad)/s} \sim 0\frac{A}{(rad)/s}$$

$$K_{\omega...SI} = \frac{J}{k_M} = \frac{5085 \cdot 10^{-7}\frac{Nm}{(rad)/s}}{38.2 \cdot 10^{-3}\frac{Nm}{A}} = 13.3\frac{mA}{(rad)/s^2}$$

Verification of Current Control

The plant is connected to the PI current controller. The controller is parameterized as described above.

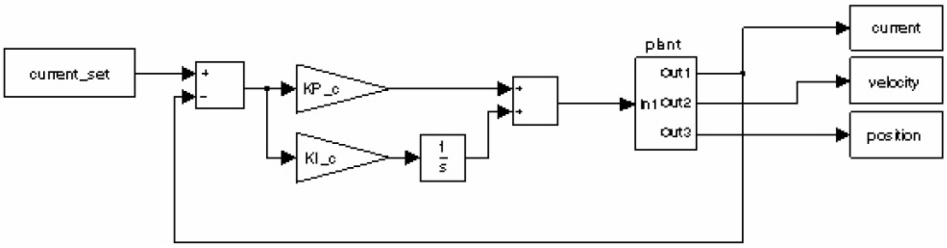


Figure 7-53 Example1 – Current Regulation, Block Model

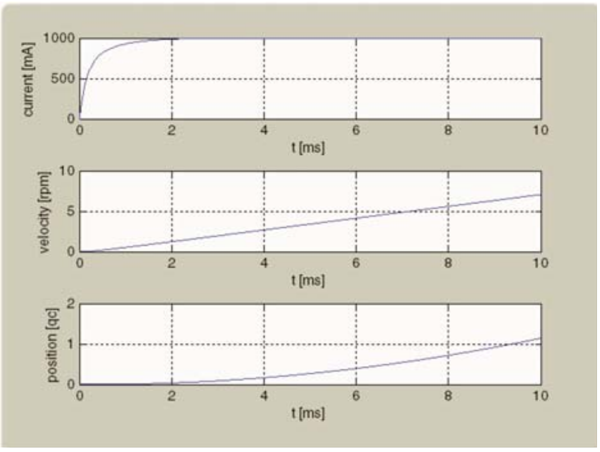


Figure 7-54 Example1 – Current Regulation, simulated

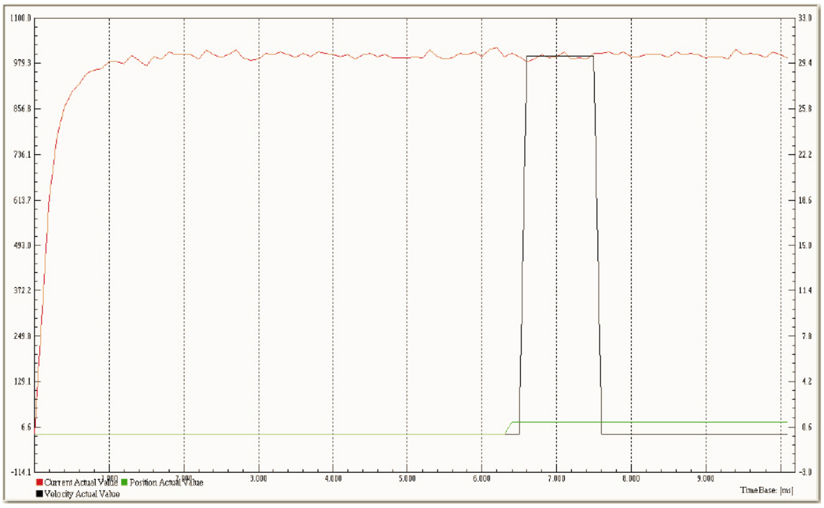


Figure 7-55 Example1 – Current Regulation, measured

Verification of Velocity Control

The PI velocity controller is connected to current regulation.

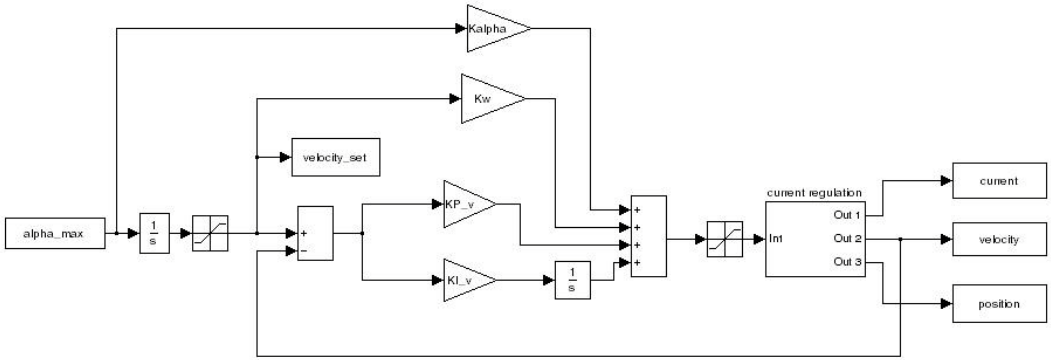


Figure 7-56 Example1 – Velocity Regulation, Block Model

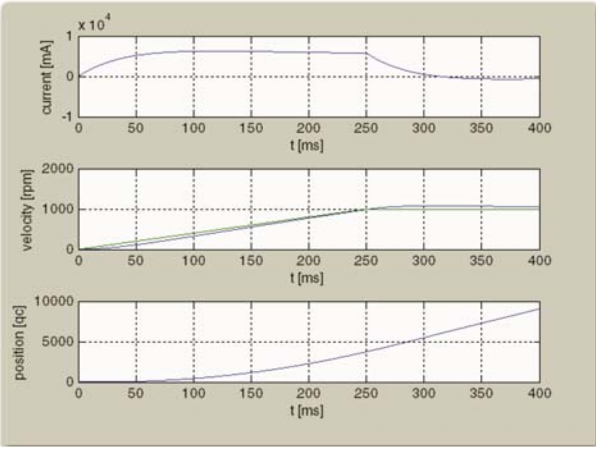


Figure 7-57 Example1 – Velocity Regulation, simulated

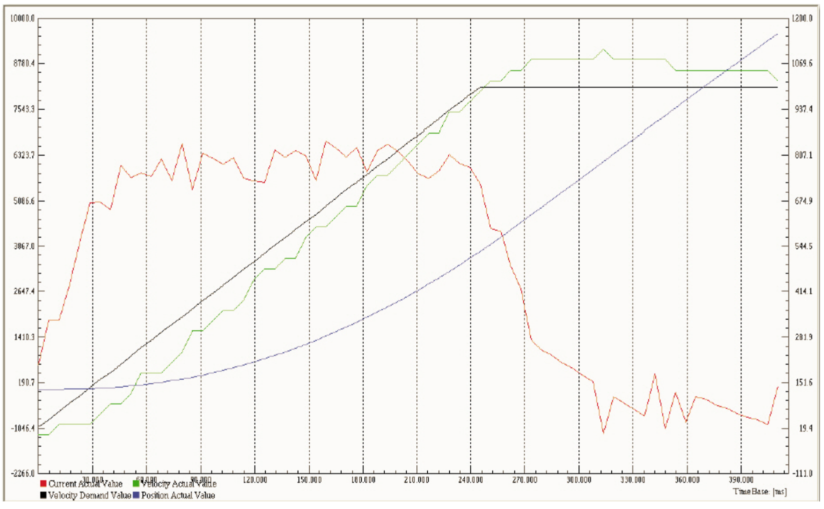


Figure 7-58 Example1 – Velocity Regulation, measured

Verification of Position Control with Feedforward

The PID position controller is connected to current regulation.

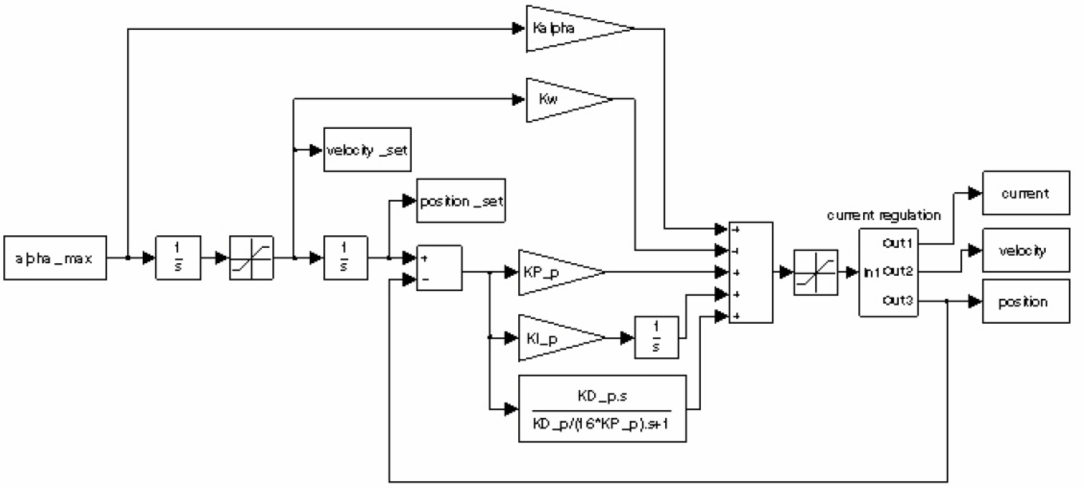


Figure 7-59 Example1 – Position Control with Feedforward, Block Model

With correct Feedforward

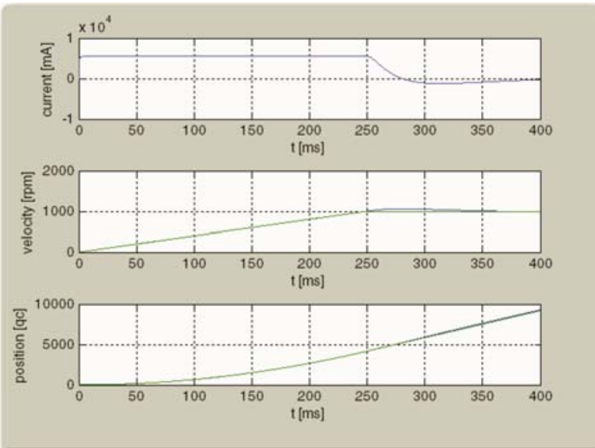


Figure 7-60 Example1 – Position Control with Feedforward, simulated

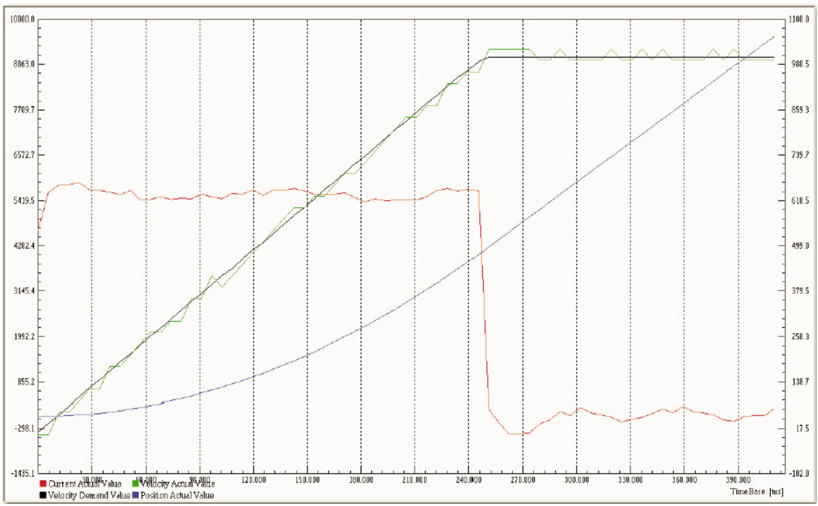


Figure 7-61 Example1 – Position Control with Feedforward, measured

Without Feedforward

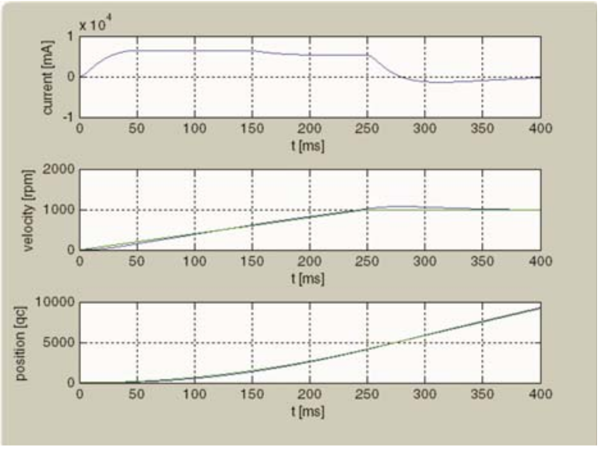


Figure 7-62 Example1 – Position Control without Feedforward, simulated

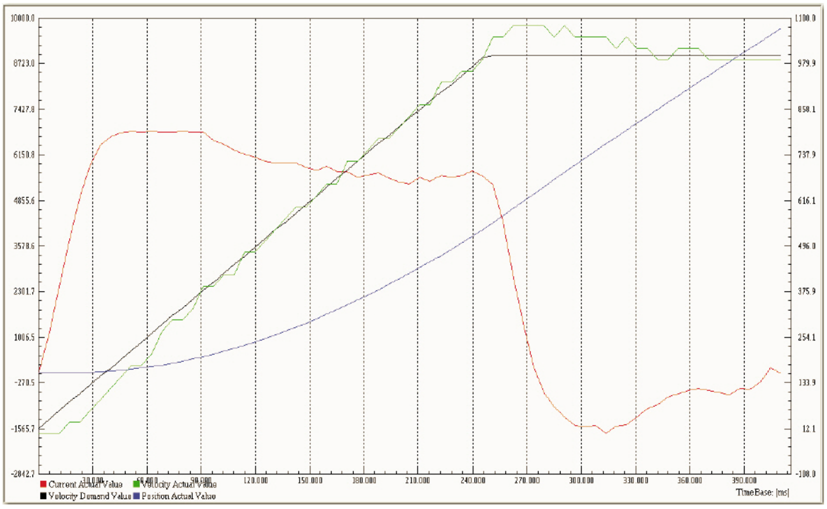


Figure 7-63 Example1 – Position Control without Feedforward, measured

With incorrect Feedforward (acceleration Feedforward parameter doubled)

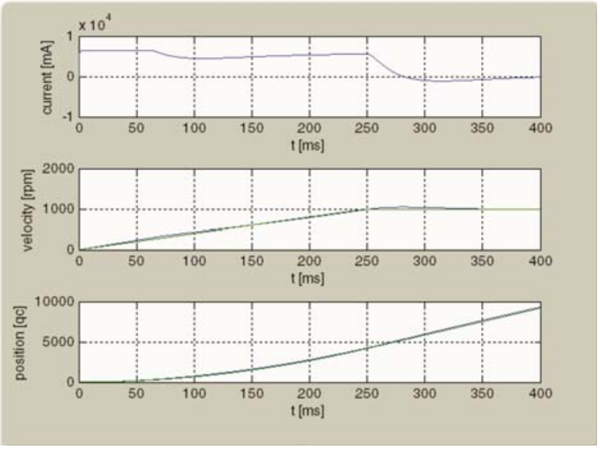


Figure 7-64 Example1 – Position Control with incorrect Feedforward, simulated

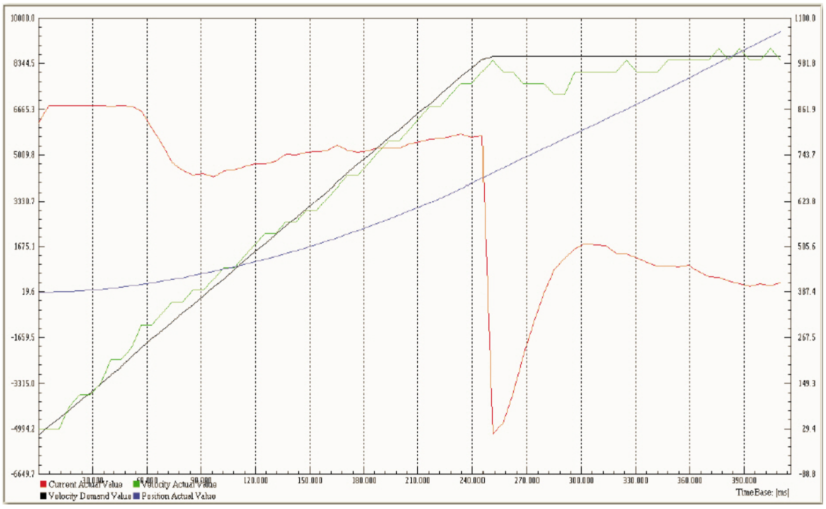


Figure 7-65 Example1 – Position Control with incorrect Feedforward, measured

7.6.2 Example 2: System with low Inertia, but high Friction

Figure 7-66 Controller Architecture – Example 2: System with low Inertia/high Friction

System Components

Item	Description	Setting
Controller EPOS3 70/10 EtherCAT (411146)		
Motor maxon RE 35 (273754)	No load speed (line 2)	n_0 = 7530 rpm
	No load current (line 3)	I_0 = 92.7 mA
	Nominal current (line 6)	I_n = 1.95 A
	Resistance phase to phase (line 10)	R = 2.07 Ω
	Inductance phase to phase (line 11)	L = 0.620 mH
	Torque constant (line 12)	k_M = 52.5 mNm/A
	Rotor inertia (line 16)	J_{motor} = 72 gcm²
Encoder HEDL 5540 (110514)	Encoder pulse number	500
Mechanical load Linear Drive	Inertia	J_{load} = 100 gcm²
	Friction, velocity-dependent	$M_r = 211\frac{\mu Nm}{(rad)/s}\omega + 8.65mNm \cdot sign(\omega)$

Table 7-73 Controller Architecture – Example 2: Components

Model of the Plant

The following parameters can be deduced:

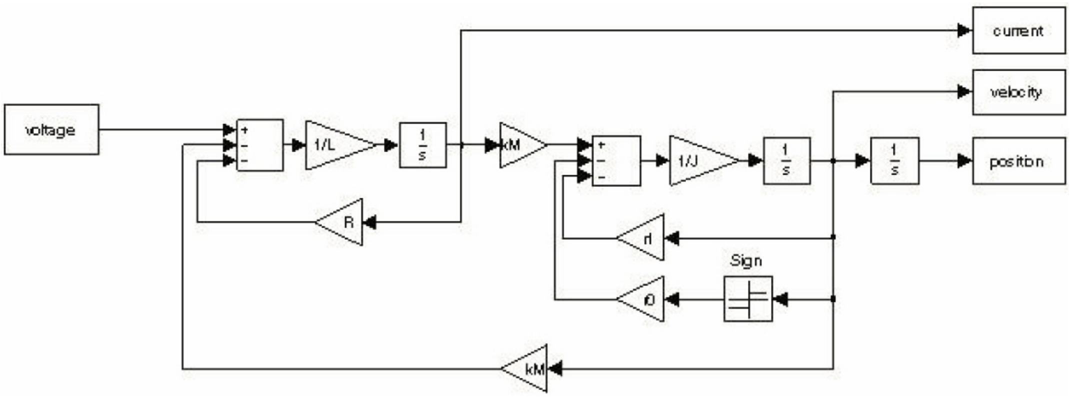


Figure 7-67 Example 2 – Block Diagram

Electrical Part

R = 2.07 Ω

L = 0.620 mH

Interface between electrical and mechanical Parts

$$k_M = 52.5\frac{mNm}{A}$$

Mechanical Part

$$J = J_{motor} + J_{load} = 172gcm^2$$

$$r_0 = 8.65mNm$$

$$r_1 = \underbrace{211\frac{\mu Nm}{(rad)/s}}_{load} + \underbrace{\frac{k_M I_0}{n_0\frac{2\pi rad}{1}\cdot\frac{1min}{60s}}}_{motor} = (211+6)\frac{\pi Nm}{(rad)/s} = 217\frac{\pi Nm}{(rad)/s}$$

- Input is the voltage at the motor winding.
- Outputs are current, velocity or position.

Regulation Tuning according to the described conditions results in the following controller and feedforward parameters:

Object Dictionary Access EPOS3 is disabled

Active Object Filter: System Parameter

Index	SubIndex	Name	Type	Access	Value
0x2008	0x00	Miscellaneous Configuration	UInt16	RW	0
0x2210		Sensor Configuration			
0x2210	0x01	Pulse Number Incremental Encoder 1	UInt32	RW	500
0x2210	0x02	Position Sensor Type	UInt16	RW	1
0x2210	0x04	Position Sensor Polarity	UInt16	RW	0
0x6065	0x00	Max Following Error	UInt32	RW	200000
0x60F6		Current Control Parameter Set			
0x60F6	0x01	Current Regulator P-Gain	Int16	RW	832
0x60F6	0x02	Current Regulator I-Gain	Int16	RW	209
0x60F9		Velocity Control Parameter Set			
0x60F9	0x01	Speed Regulator P-Gain	Int16	RW	1575
0x60F9	0x02	Speed Regulator I-Gain	Int16	RW	257
0x60F9	0x04	Velocity Feedforward Factor in Speed Regulator	UInt16	RW	4426
0x60F9	0x05	Acceleration Feedforward Factor in Speed Regulator	UInt16	RW	270
0x60FB		Position Control Parameter Set			
0x60FB	0x01	Position Regulator P-Gain	Int16	RW	386
0x60FB	0x02	Position Regulator I-Gain	Int16	RW	1193
0x60FB	0x03	Position Regulator D-Gain	Int16	RW	616
0x60FB	0x04	Velocity Feedforward Factor in Position Regulator	UInt16	RW	4426
0x60FB	0x05	Acceleration Feedforward Factor in Position Regulator	UInt16	RW	270
0x6402	0x00	Motor Type	UInt16	RW	1
0x6410		Motor Data			
0x6410	0x01	Nominal Current	UInt16	RW	1950
0x6410	0x02	Output Current Limit	UInt16	RW	3900
0x6410	0x03	Pole Pair Number	UInt8	RW	1
0x6410	0x04	Maximal Motor Speed	UInt32	RW	12000
0x6410	0x05	Thermal Time Constant Winding	UInt16	RW	300

Figure 7-68 Example 2 – System Parameters, real

For numerical simulation, the conversion results from EPOS3 to SI units are as follows:

Current Controller

$$K_{P...EPOS3} = 832 \quad \rightrightarrows \quad K_{P...SI} = 3.25\Omega$$

$$K_{I...EPOS3} = 209 \quad \rightrightarrows \quad K_{I...SI} = 8.17\frac{k\Omega}{s}$$

Velocity Controller

$$K_{P...EPOS3} = 1575 \quad \rightrightarrows \quad K_{P...SI} = 31.5\frac{mA}{(rad)/s}$$

$$K_{I...EPOS3} = 257 \quad \rightrightarrows \quad K_{I...SI} = 1.29\frac{A/s}{(rad)/s}$$

Position Controller

$$K_{P...EPOS3} = 386 \quad \rightrightarrows \quad K_{P...SI} = 3.86\frac{A}{rad}$$

$$K_{I...EPOS3} = 1193 \quad \rightrightarrows \quad K_{I...SI} = 93.1\frac{A/s}{rad}$$

$$K_{D...EPOS3} = 616 \quad \rightrightarrows \quad K_{D...SI} = 49.3\frac{mAs}{rad}$$

Positioning and Velocity Feedforward

$$K_{\omega...EPOS3} = 4426 \quad \rightrightarrows \quad K_{\varpi...SI} = 4.42\frac{mA}{(rad)/s}$$

$$K_{\alpha...EPOS3} = 270 \quad \rightrightarrows \quad K_{\alpha...SI} = 270\frac{\mu A}{(rad)/s^2}$$

Plausibility Check

$$K_{\omega...SI} = \frac{r_1}{k_M} = 4.13\frac{mA}{(rad)/s}$$

$$K_{\omega...SI} = \frac{J}{k_M} = \frac{172 \cdot 10^{-7}\frac{Nm}{(rad)/s}}{52.5 \cdot 10^{-3}\frac{Nm}{A}} = 327\frac{\mu A}{(rad)/s^2}$$

Verification of Current Control

The plant is connected to the PI current controller. The controller is parameterized as described above.

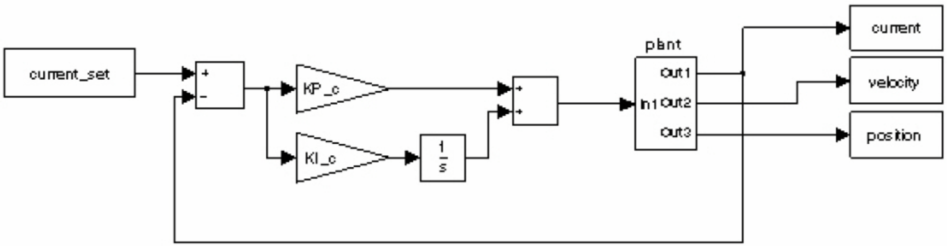


Figure 7-69   Example 2 – Current Regulation, Block Model

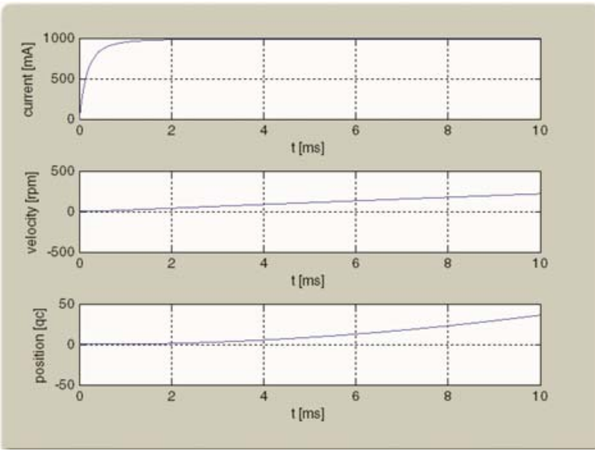


Figure 7-70   Example 2 – Current Regulation, simulated

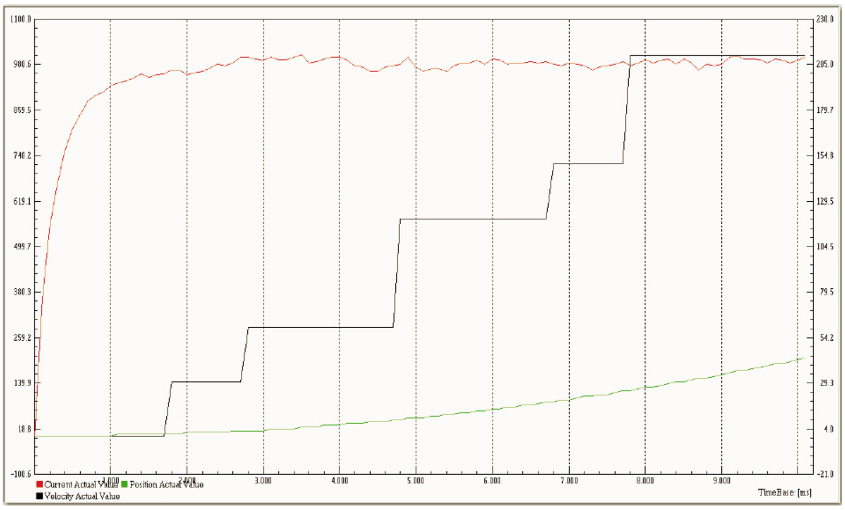


Figure 7-71   Example 2 – Current Regulation, measured

Verification of Velocity Control

The PI velocity controller is connected to current regulation.

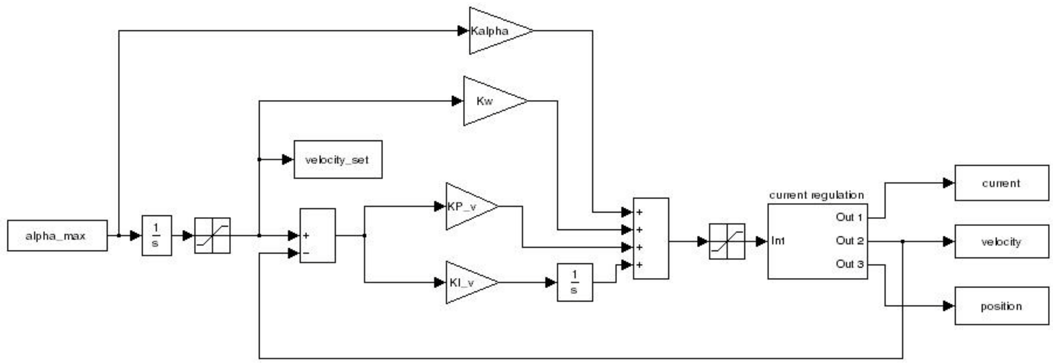


Figure 7-72 Example 2 – Velocity Regulation, Block Model

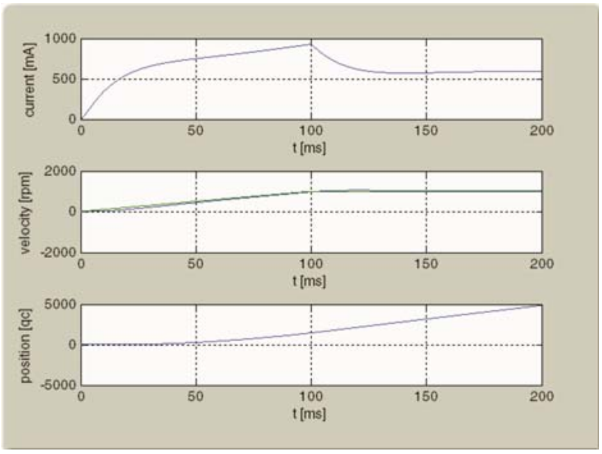


Figure 7-73 Example 2 – Velocity Regulation, simulated

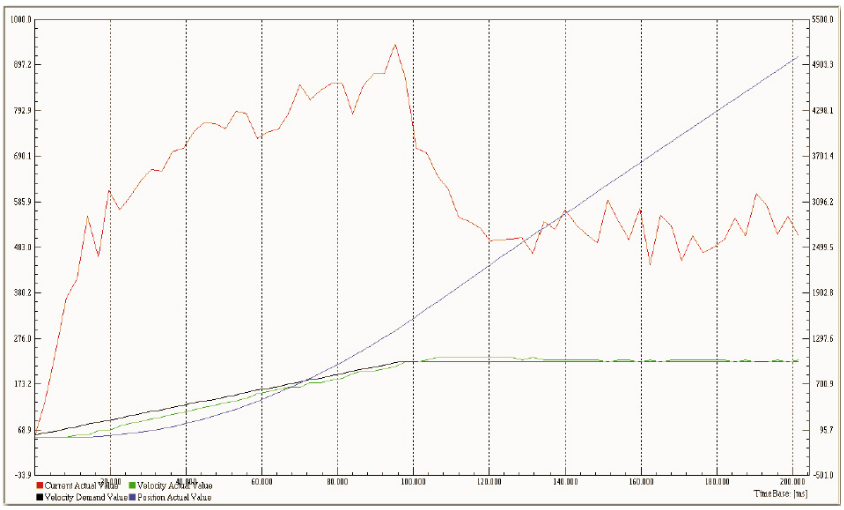


Figure 7-74 Example 2 – Velocity Regulation, measured

Verification of Position Control with Feedforward

The PID position controller is connected to current regulation.

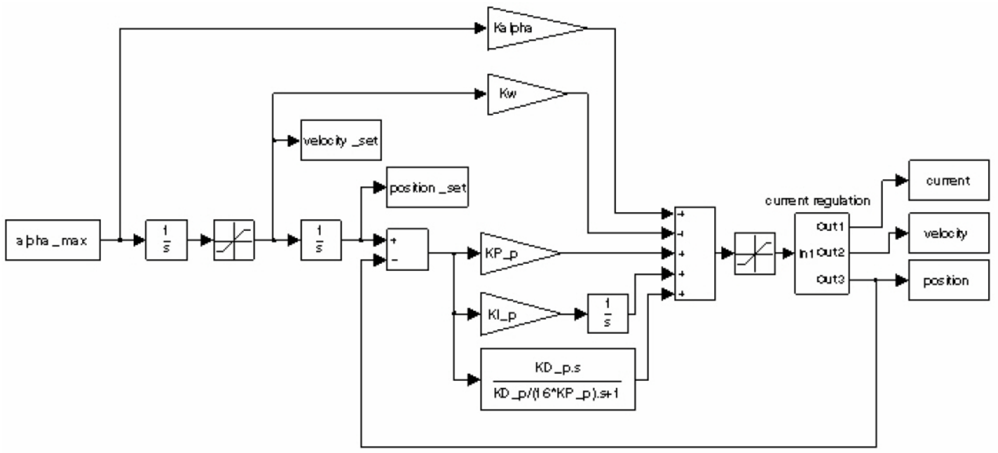


Figure 7-75 Example 2 – Position Control with Feedforward, Block Model

With correct Feedforward

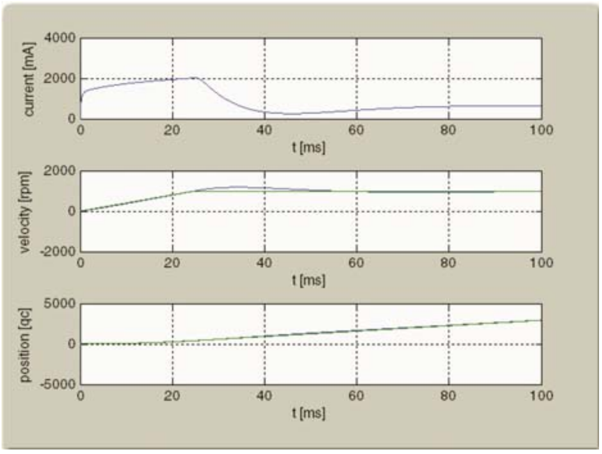


Figure 7-76 Example 2 – Position Control with Feedforward, simulated

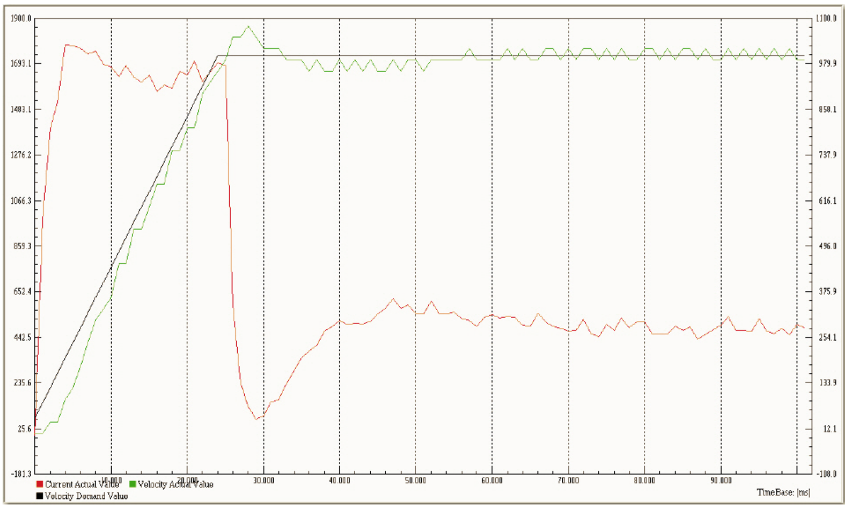


Figure 7-77 Example 2 – Position Control with Feedforward, measured

Without Feedforward

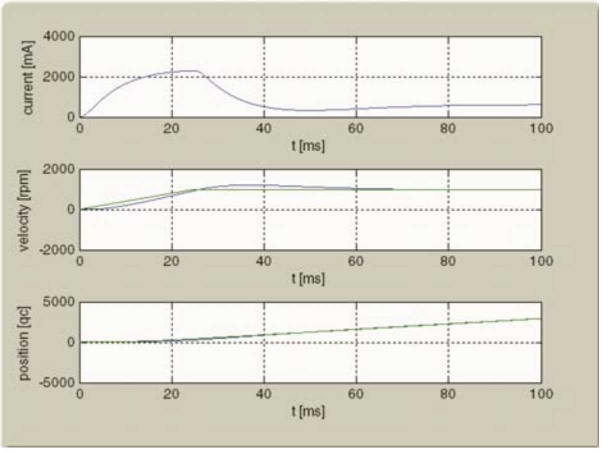


Figure 7-78 Example 2 – Position Control without Feedforward, simulated

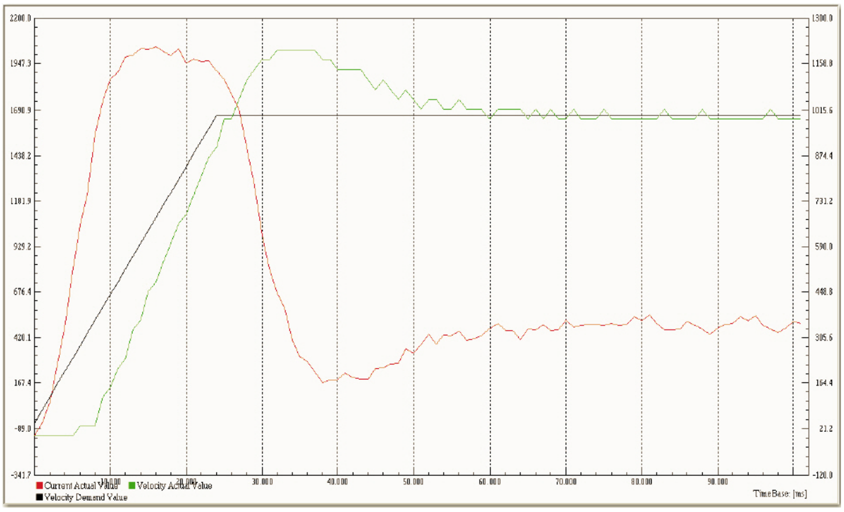


Figure 7-79 Example 2 – Position Control without Feedforward, measured

7.7 Conclusion

Scaling of the internal controller parameters is a specific EPOS3 EtherCAT feature. To understand these parameters and to use them in analytical calculations, respectively numerical simulations, understanding on how to map EPOS3 EtherCAT's internal controller parameters to SI units controller parameters, and vice versa, is essential.

In practice, direct drive systems are often used because of their lower overall costs and the requirement for a backlash-free behavior. As a result, the ratio between motor inertia and load inertia often are 1:10, or higher.

Therefore, EPOS3 EtherCAT's PID position control with feedforward compensation is of great advantage. Compared to simple PID control, the feedforward compensation provides significant faster and more accurate setpoint following.

8 Data Recording

8.1 In Brief

A wide variety of operating modes permit flexible configuration of drive and automation systems by using positioning, speed and current regulation. The built-in EtherCAT interface allows networking to multiple axes drives as well as online commanding by EtherCAT master units.

EPOS3 EtherCAT features a built-in data recorder to debug errors and to monitor motion control parameters and actual values.

8.1.1 Objective

The present Application Note explains the functionality of the built-in data recorder. Features and configuration options are explained.

Contents

8.1.2 Scope

Hardware	Order #	Firmware Version	Reference
EPOS3 EtherCAT		2200h	Firmware Specification
EPOS3 70/10 EtherCAT	411146	2200h or higher	

Table 8-74 Data Recording – covered Hardware and required Documents

8.1.3 Tools

Tools	Description
Software	«EPOS Studio» Version 2.00 or higher

Table 8-75 Data Recording – recommended Tools

8.2 Overview

8.2.1 Launching the Data Recorder

1) Start «EPOS Studio».
2) Start Data Recorder – either click right ¤Selected Node¤ or click ¤Tools¤ in Navigation Window.
3) Following screen will be displayed:

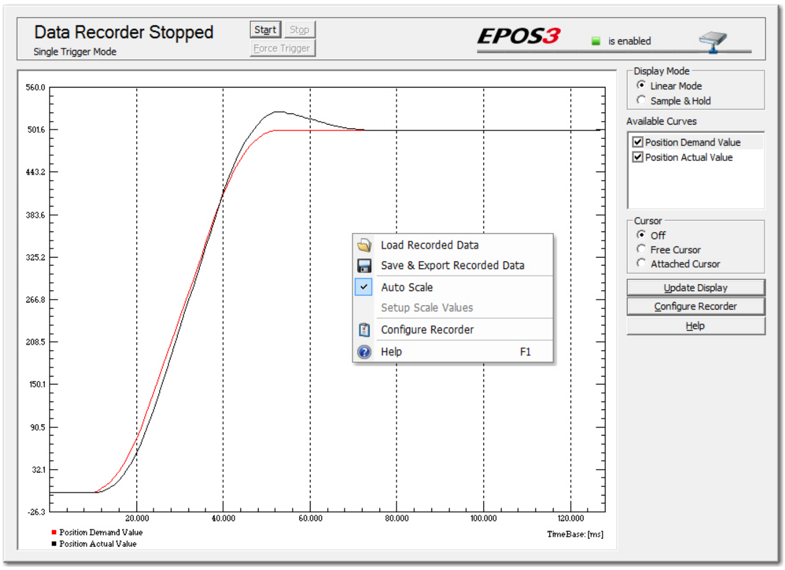


Figure 8-80 Data Recorder Overview

8.2.2 Control Elements and their Function

Title Bar

Control Element	Description / Function	
Status	Displays data recorder's status. The following states are possible:	
	Data Recorder Running Continuous Acquisition Mode	Data are continuously recorded and displayed.
	Data Recorder Waiting Single Trigger Mode	On standby, waiting to receive a trigger to start a single data record (for trigger options ➔ page 8-100).
	Data Recorder Triggered Single Trigger Mode	Sampling in process until data buffer is full.
	Data Recorder Stopped Single Trigger Mode or Continuous Acquisition Mode	Recording completed and stopped, results are being displayed.
Start	Commences sampling. In "Single Trigger Mode", the data recorder is waiting for a trigger. In "Continuous Acquisition Mode", the data recorder is continuously recording and displaying data.	
Stop	Stops sampling. Latest recorded data are being displayed.	
Force trigger	A trigger has been activated.	

Table 8-76 Data Recording – Title Bar

Options Bar

Control Element	Description / Function	
Display Mode	Linear Mode	To display data, linear interpolation will be used.
	Sample & Hold	Between samples, no interpolation will be used.
Available Curves	Available curves will be listed. Tick check box to show/untick to hide a curve in the display.	
Cursor	Off	No curser will be displayed.
	Free Cursor	Curser will appear, as soon as the mouse is moved.
	Attached Cursor	Moving the mouse will attach the cursor to the selected curve. Use “Available Curves” to select another curve.
Update Display	Last sampled data will be loaded and displayed.	
Configure Recorder	To select sampled data and to configure the data recorder (➔“Data Recorder Configuration” on page 8-98).	

Table 8-77 Data Recording – Option Bar

Display

Control Element	Description / Function
Zoom	**Zoom in:** Click left and draw a rectangle over desired area – status indication (upper left corner) will change to “Zoomed”. **Zoom out:** Click right – status indication will disappear.
Cursor	If activated, the cursor will appear as small circle. Cursor’s actual coordinates are displayed in the upper right corner.
Left / Right Scale	Each data set may be displayed in either left or right pane (➔Data Recorder Configuration).
Time Scale	At bottom border with corresponding time base at lower right corner.
Legend	Currently displayed curves’ legend appears in lower left corner.

Table 8-78 Data Recording – Display

Context Menu

Control Element	Description / Function	
Load Recorded Data	Load recorded data from file (*.rda).	
Save & Export Recorded Data	Save recorded data to file in following file formats:	
	*.rda	Binary Format (for use with «EPOS Studio»)
	*.txt	ASCII Text Format (for import in Microsoft Excel)
	*.csv	Comma Separated Values (for import in Microsoft Excel)
	*.bmp	Bitmap Format
Auto Scale	Select this option to automatically calculate optimal scale values.	
Setup Scale Values	If “Auto Scale” is deselected, left/right pane and time scale can be defined manually.	
Manual	Open connected device’s online help manual.	
Configure Recorder	To select sampled data and to configure data recorder (➔Data Recorder Configuration).	

Table 8-79 Data Recording – Context Menu

8.3 Data Recorder Configuration

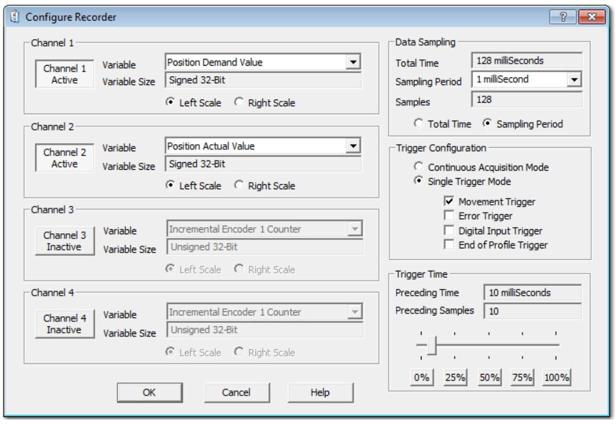


Figure 8-81 Data Recording – “Configure Recorder” Dialog

Channel 1…4

Control Element	Description / Function
Channel Active/ Inactive	Activate/deactivate up to four recorder channels.
Variable	Select desired variables to be recorded.
Variable Size	Displays size of selected variable.
Left / Right Scale	Select pane to display the recorded data.

Table 8-80 “Configure Recorder” – Channel

Data Sampling

Control Element	Description / Function
Total Time	Displays total duration.
Sampling Period	Select sampling period.
Samples	Displays number of samples per channel.
Total Time or Sampling Period	Select whether to determine the total time or the sampling period.

Table 8-81 “Configure Recorder” – Data Sampling

Trigger Configuration

<table>
<tr><th>Control Element</th><th colspan="2">Description / Function</th></tr>
<tr><td>Continuous Acquisition Mode</td><td colspan="2">Data will continuously be recorded.</td></tr>
<tr><td rowspan="4">Single Trigger Mode</td><td>Movement Trigger</td><td>A trigger is activated upon every start of a movement.</td></tr>
<tr><td>Error Trigger</td><td>A trigger is activated upon an occurring error.</td></tr>
<tr><td>Digital Input Trigger</td><td>A trigger is activated at an edge of a digital input.
Note: In “Homing Mode”, also the current threshold will be interpreted as a trigger.</td></tr>
<tr><td>End of Profile Trigger</td><td>A trigger is activated at the end of a movement profile.</td></tr>
</table>

Table 8-82 “Configure Recorder” – Trigger Configuration

Trigger Time

Control Element	Description / Function
Preceding Time	The lead time to be displayed prior activation of a trigger. “100%” permits display of data prior the trigger. **Best Practice:** Use the trigger time in combination with the error trigger to debug errors.
Preceding Samples	Displays the number of samples before the trigger.

Table 8-83 “Configure Recorder” – Trigger Time

8.4 Example: Data Recording in “Profile Position Mode”

8.4.1 Step 1: Configure Data Recorder

1) Click ¤Configure Recorder¤ in the options bar or select ¤Configure Recorder¤ from the context menu.

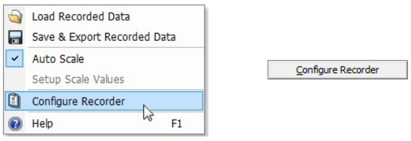


Figure 8-82 Configure Data Recorder

2) Select the following variables:
   - Position Demand Value
   - Position Actual Value
   - Velocity Actual Value
   - Current Actual Value
3) Select a sampling period of 1 ms.
4) Select ¤Single Trigger Mode¤ and tick ¤Movement Trigger¤.
5) Select a preceding time of 0 microseconds.

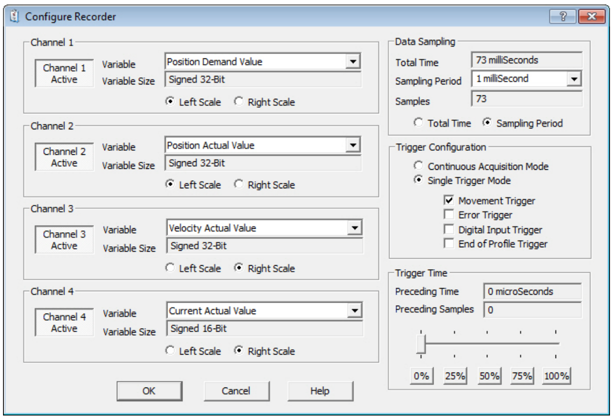


Figure 8-83 Select Configuration Options

6) Click ¤OK¤ to save settings.

8.4.2 Step 2: Execute Movement

1) Change the active view to “Profile Position Mode”.
2) Activate “Profile Position Mode”.
3) Enable the EPOS3 70/10 EtherCAT and start a relative movement.

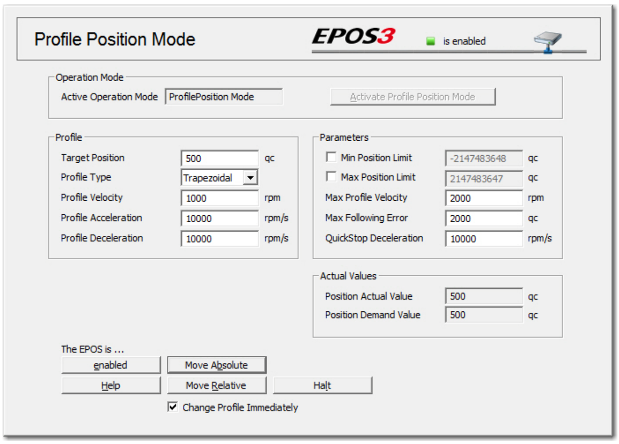


Figure 8-84 Execute Movement

8.4.3 Step 3: Update Display

Change back to the view “Data Recording”. If the display does not automatically refresh, press ¤Update Display¤ button.

8.4.4 Step 4: Save recorded Data

1) Click right ¤Save & Export Recorded Data¤ to open context menu.

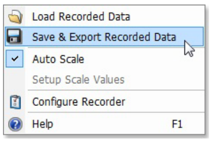


Figure 8-85 Save recorded Data

2) Select desired path.
3) Enter a file name.
4) Press ¤Save¤.

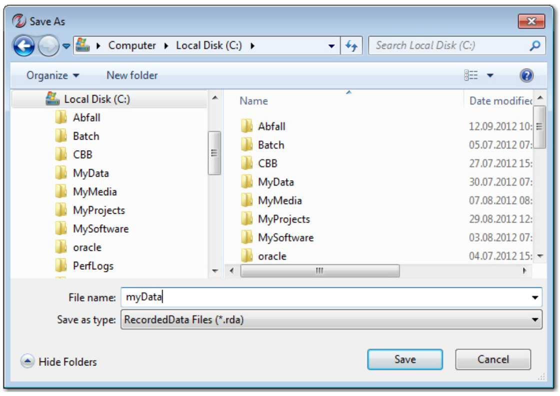


Figure 8-86 Save recorded Data

Best Practice
Save recorded data as ASCII text file or as bitmap!

8.4.5 Step 5: Analyze recorded Data

1) Select cursor mode ¤Attached Cursor¤.
2) Tick ¤Position Actual Value¤ in “Available Curves”.
3) Move cursor over the display and read the attached curve's values.

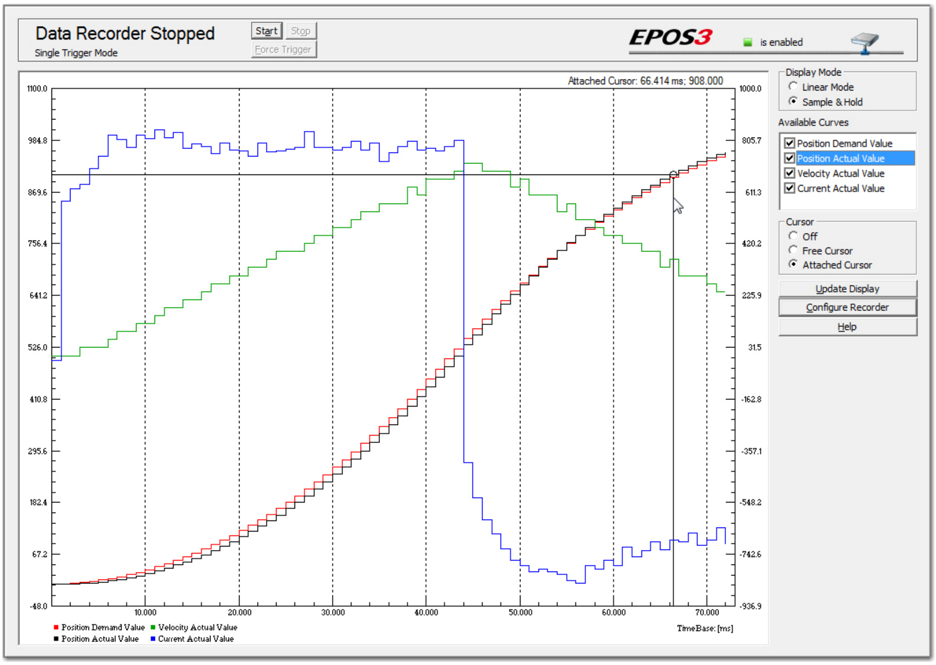


Figure 8-87 Analyze recorded Data

8.4.6 Step 6: Restart Data Recorder

Click ¤Start¤ to restart and prepare the data recorder for the next recording.

Figure 8-88 Restart Data Recorder

8.5 Data Recorder Specifications

8.5.1 Functionalities

Recorder

- Executed in current regulator (max 10 kHz sampling rate)
- Configurable sampling rate
- Total buffer size: 512 words

While the data recorder is running, data are sampled to a ring buffer until a trigger is set. After a trigger, data will be recorded until the buffer is full.

Variables

- Max. four variables of the Object Dictionary
- 16-bit and 32-bit variables are allowed (one word)
- 8-bit variables need 16-bits in the data recorder memory

Trigger

Following automatic trigger modes are supported:

- Manuel Trigger – set by communication
- Movement Trigger – set at movement start
- Error Trigger – set by error
- Digital Input Trigger – set by digital input
- End of Profile Trigger – set at movement stop

8.5.2 Object Description

8.5.2.1 Data Recorder Control

Description

The data recorder is controlled by write access.

Name	Data Recorder Control	
Index	0x2010	
Subindex	0x00	
Type	UNSIGNED16	
Access	RW	
Default Value	0	
Value Range	0	3

Bit	Description
15…2	reserved
1	0 = no trigger 1 = force trigger
0	0 = stop recorder 1 = start recorder

Table 8-84 Data Recorder Control – Bits

8.5.2.2 Data Recorder Configuration

Description

Configures the auto trigger functions.

Name	Data Recorder Configuration
Index	0x2011
Subindex	0x00
Type	UNSIGNED16
Access	RW
Default Value	0
Value Range	➔Table 8-85

Bit	Description
15…4	reserved
3	1 = trigger at end of profile
2	1 = trigger upon digital input
1	1 = trigger by error state
0	1 = trigger at movement start

Table 8-85 Data Recorder Configuration – Bits

8.5.2.3 Data Recorder Sampling Period

Description

Sampling period as a multiple of the current regulator cycle (n multiplied by 0.1ms).

Name	Data Recorder Sampling Period	
Index	0x2012	
Subindex	0x00	
Type	UNSIGNED16	
Access	RW	
Default Value	10	
Value Range	0	65535

8.5.2.4 Data Recorder Number of Preceding Samples

Description

Number of preceding samples defines the trigger position in the data recorder buffer.

Name	Data Recorder Number of Preceding Samples	
Index	0x2013	
Subindex	0x00	
Type	UNSIGNED16	
Access	RW	
Default Value	0	
Value Range	0	65535

8.5.2.5 Data Recorder Number of Sampling Variables

Description

Number of variables (max. 4) to be recorded.

Name	Data Recorder Number of Sampling Variables	
Index	0x2014	
Subindex	0x00	
Type	UNSIGNED16	
Access	RW	
Default Value	0	
Value Range	0	4

8.5.2.6 Data Recorder Index of Variables

Description

Index of Object Dictionary.

Related Objects

➔Data Recorder Subindex of Variables

Name	Data Recorder Index of Variables
Index	0x2015
Number of entries	0x05

Names	Data Recorder Index of Variable 1 Data Recorder Index of Variable 2	Data Recorder Index of Variable 3 Data Recorder Index of Variable 4
Index	0x2015	
Subindex	0x01…0x04	
Type	UNSIGNED16	
Access	RW	
Default Value	0	
Value Range	➔Object Dictionary	

8.5.2.7 Data Recorder Subindex of Variables

Description

Subindex of Object Dictionary.

Related Objects

➔Data Recorder Index of Variables

Name	Data Recorder Subindex of Variables
Index	0x2016
Number of entries	0x05

Names	Data Rec… Subindex of Variable 1 Data Rec… Subindex of Variable 2	Data Rec… Subindex of Variable 3 Data Rec… Subindex of Variable 4
Index	0x2016	
Subindex	0x01…0x04	
Type	UNSIGNED16	
Access	RW	
Default Value	0	
Value Range	➔Object Dictionary	

8.5.2.8 Data Recorder Status

Description

Data recorder's status.

Name	Data Recorder Status
Index	0x2017
Subindex	0x00
Type	UNSIGNED16
Access	RO
Default Value	0
Value Range	➔Table 8-86

Bit	Description
15…2	reserved
1	0 = not triggered 1 = triggered
0	0 = stopped 1 = running

Table 8-86 Data Recorder Status – Bits

8.5.2.9 Data Recorder Max. Number of Samples

Description

Defines maximal number of samples per variable. The parameter is dynamically calculated by the data recorder.

The maximal number of samples is the memory size (512 words) divided by the sum of the variable size (in words) of all configured variables.

Name	Data Recorder max. Number of Samples	
Index	0x2018	
Subindex	0x00	
Type	UNSIGNED16	
Access	RO	
Default Value	0	
Value Range	–	–

Example:

Sum of Variable Size [word]	Example	Number of Samples
1	1 x 16-bit variable or 1 x 8-bit variable	512
2	1 x 32-bit variable	256
3	1 x 16-bit and 1 x 32-bit variable	170
…	…	…
8	4 x 32-bit variables	64

Table 8-87 Data Recorder Max. Number of Samples – Example

8.5.2.10 Data Recorder Number of recorded Samples

Description

Offset to the start of the recorded data vector within the ring buffer.

Name	Data Recorder Number of recorded Samples	
Index	0x2019	
Subindex	0x00	
Type	UNSIGNED16	
Access	RO	
Default Value	0	
Value Range	–	–

8.5.2.11 Data Recorder Data Buffer

Description

Memory for the different data recorder’s ring buffers. Memory allocation is dynamically calculated when the recorder is started.

Name	Data Recorder Data Buffer	
Index	0x201B	
Subindex	0x00	
Type	Domain	
Access	RO	
Default Value	0	
Value Range	–	–

Data Buffer Segmentation (Example: 2 x 16-bit variables, 1 x 32-bit variable)

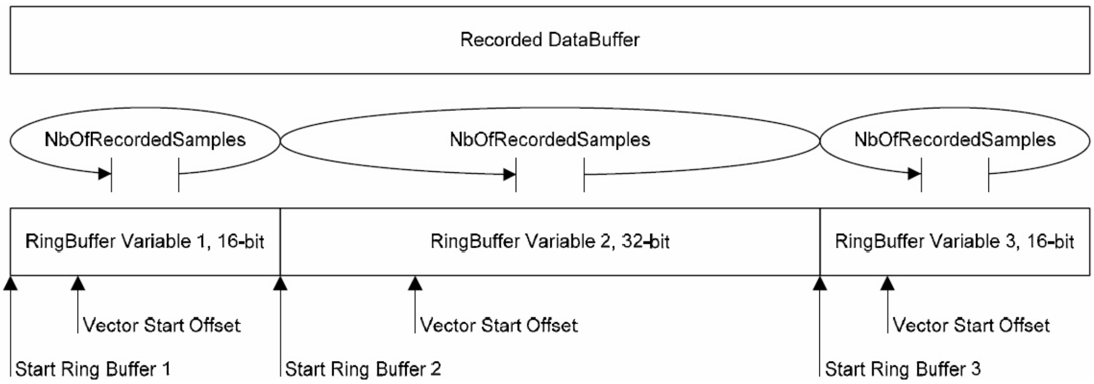


Figure 8-89 Data Recorder Data Buffer – Segmentation

StartRingBuffer1 = 0

StartRingBuffer2 = MaxNbOfSamples * nbOfWords(Variable1)

StartRingBuffer3 = MaxNbOfSamples * (nbOfWords(Variable1) + nbOfWords(Variable2)

9 Extended Encoders Configuration

9.1 In Brief

In addition to standard incremental digital encoders to detect the actual position, a number of other sensor types may be used:

- SSI absolute encoder (single or multi turn, 6 to 32 bit resolution, Gray or binary code, RS422)
- Analog incremental encoder (2-channel, max. 10 bit interpolation, Sinus-Cosinus 1 Vss)
- Digital incremental encoder (2-channel or 3-channel, up to 2 500 000 impulse, RS422)

9.1.1 Objective

The present Application Note explains the configuration of extended encoders and features “in practice examples” suitable for daily use.

Contents

9.1.2 Scope

Hardware	Order #	Firmware Version	Reference
EPOS3 EtherCAT		2200h	Firmware Specification
EPOS3 70/10 EtherCAT	411146	2200h or higher	Cable Starting Set Hardware Reference

Table 9-88 Extended Encoders Configuration – covered Hardware and required Documents

9.1.3 Tools

Tools	Description
Crimper	Molex hand crimper (63819-0000)
Software	«EPOS Studio» Version 2.00 or higher

Table 9-89 Extended Encoders Configuration – recommended Tools

9.2 Hardware Signals

The extended position sensors can be connected to the EPOS3 EtherCAT Positioning Controllers's digital inputs and outputs.

9.2.1 EPOS3 70/10 EtherCAT

Signal 1 Connector (J5)

Contains differential "High Speed" digital inputs and outputs.

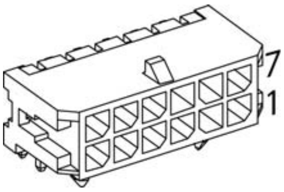


Figure 9-90 EPOS3 70/10 EtherCAT – Signal 1 Connector (J5)

Pin	Signal	Description
1	DigIN10/	Digital Input 10 "High Speed Command" complement
2	DigIN10	Digital Input 10 "High Speed Command"
3	DigIN9/	Digital Input 9 "High Speed Command" complement
4	DigIN9	Digital Input 9 "High Speed Command"
5	DigIN7/	Digital Input 7 "High Speed Command" complement
6	DigIN7	Digital Input 7 "High Speed Command"
7	DigIN8/	Digital Input 8 "High Speed Command" complement
8	DigIN8	Digital Input 8 "High Speed Command"
9	$+V_{AUX}$	Auxiliary output voltage (+5 VDC / 150 mA)
10	D_Gnd	Digital signal ground
11	DigOUT5/	Digital Output 5 "High Speed Output" complement
12	DigOUT5	Digital Output 5 "High Speed Output"

Table 9-90 EPOS3 70/10 EtherCAT – Signal 1 Connector (J5)

9.3 Sensor Types

9.3.1 SSI Absolute Encoder

9.3.1.1 General Description

The Synchronous Serial Interface (SSI) is an interface to connect an absolute position sensor to a controller, such as EPOS3 70/10 EtherCAT. This interface uses a clock signal from the controller to the sensor and a data signal from the sensor back to the controller. The serial data stream from the sensor begins with the most significant bit.

The number of data bits (for multi turn and single turn resolution) and the clock rate can be configured.

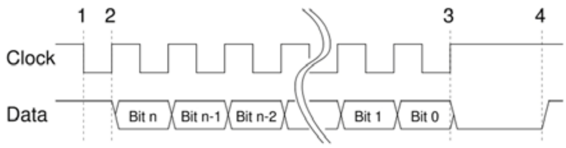


Figure 9-91 SSI Principle

9.3.1.2 Implementation

The EPOS3 EtherCAT's SSI interface uses DigOUT5 and DigOUT5/ as differential clock output and DigIN 9 and DigIN 9/ as differential data input.

If the supply voltage of the SSI sensor is 5 V and the current is less than 150 mA, it can be directly supplied from the $+V_{AUX}$ signal (J5-9). Otherwise, an external power supply must be connected to power the sensor.

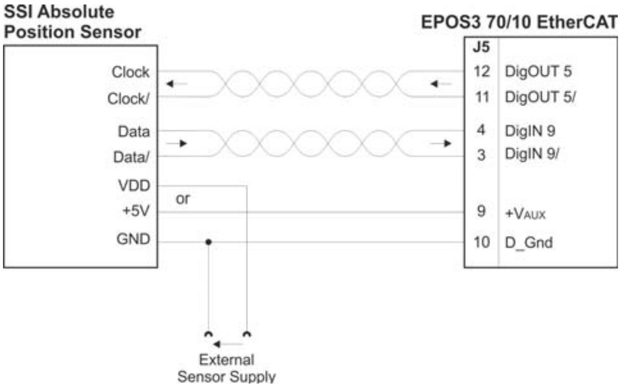


Figure 9-92 EPOS3 70/10 EtherCAT – SSI Encoder Connection

Differential	
DigIN9 “High Speed Command”	Connector [J5] Pins [3] / [4]
Min. differential input voltage	±200 mV
Line receiver (internal)	EIA RS422 Standard
Max. input frequency	5 MHz

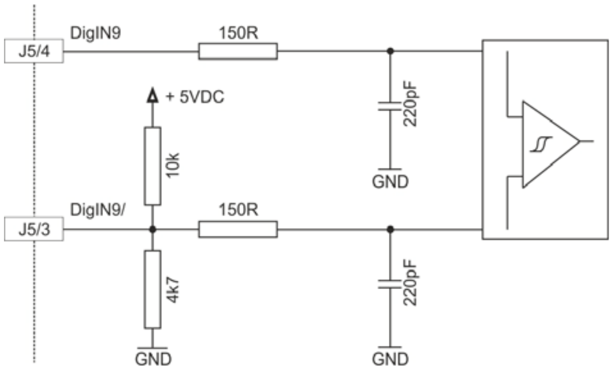


Figure 9-93 EPOS3 70/10 EtherCAT – DigIN9 “Differential” Circuit

Differential	
DigOUT5 “High Speed Output”	Connector [J5] Pins [11] / [12]
Differential output voltage	min 1.5 V @ R_L = 54 Ω
Output current	max. 60 mA
Line transceiver (internal)	EIA RS422 Standard
Max. output frequency	5 MHz

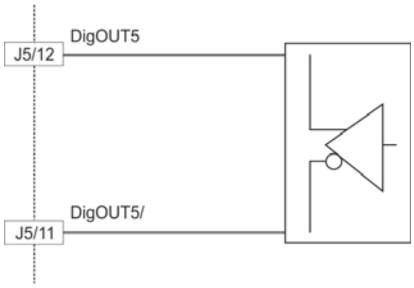


Figure 9-94 EPOS3 70/10 EtherCAT – DigOUT5 “Differential” Circuit

9.3.1.3 Choice of Manufacturers for SSI Absolute Encoders

Manufacturer	Contact
Baumer	Baumer Electric AG, CH-8501 Frauenfeld www.baumerelectric.com
Heidenhain	DR. JOHANNES HEIDENHAIN GmbH, DE-83292 Traunreut www.heidenhain.de
Hengstler	HENGSTLER GmbH, DE-78554 Aldingen www.hengstler.com
Posital Fraba	POSITAL GmbH, DE-51063 Cologne www.posital.de
and others	

Table 9-91 SSI Absolute Encoder – Manufacturers (not concluding)

9.3.2 Incremental Encoder 2

9.3.2.1 General description

The incremental signals are transmitted as square-wave pulse trains A and B, phase-shifted by 90° electrical. The signals A and B and their inverted signals typically have TTL levels.

9.3.2.2 Implementation

A second incremental encoder can be connected to the EPOS3 70/10 EtherCAT's digital inputs DigIN7 to DigIN9.

If the supply voltage of the incremental encoder is 5 V and the current is less than 150 mA, it can be directly supplied from the $+V_{AUX}$ signal (J5-9). Otherwise, an external power supply must be connected to power the sensor.

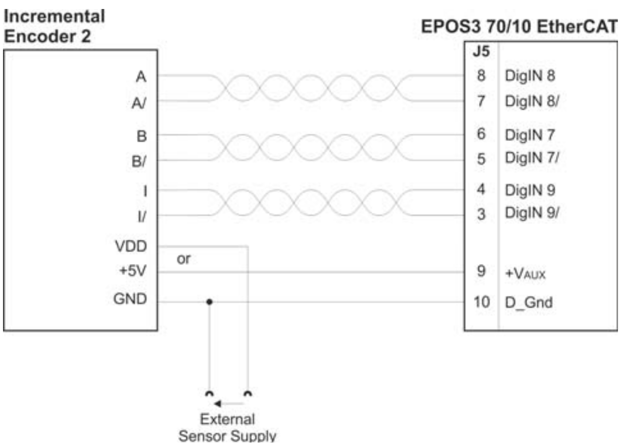


Figure 9-95 EPOS3 70/10 EtherCAT – Incremental Encoder 2 Connection

Differential	
DigIN7 “High Speed Command” DigIN8 “High Speed Command” DigIN9 “High Speed Command”	Connector [J5] Pins [5] / [6 Connector [J5] Pins [7] / [8] Connector [J5] Pins [3] / [4]
Min. differential input voltage	±200 mV
Line receiver (internal)	EIA RS422 Standard
Max. input frequency	5 MHz

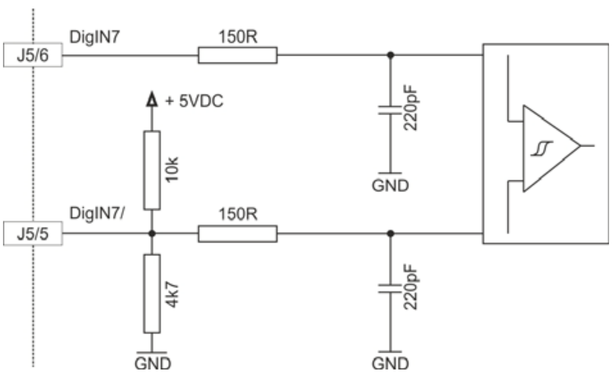


Figure 9-96 EPOS3 70/10 EtherCAT – DigIN7 “Differential” Circuit (analogously valid also for DigIN8 and DigIN9)

9.3.2.3 Choice of Manufacturers for Incremental Encoder 2

Manufacturer	Contact
maxon	maxon motor ag, CH-6072 Sachseln www.maxonmotor.com
Baumer	Baumer Electric AG, CH-8501 Frauenfeld www.baumerelectric.com
Heidenhain	DR. JOHANNES HEIDENHAIN GmbH, DE-83292 Traunreut www.heidenhain.de
Hengstler	HENGSTLER GmbH, DE-78554 Aldingen www.hengstler.com
Scancon	SCANCON A/S, DK-3450 Alleroed www.scancon.dk
and others	

Table 9-92 Incremental Encoder 2 – Manufacturers (not concluding)

9.3.3 Sinus Incremental Encoder 2

9.3.3.1 General Description

The sinusoidal incremental signals A and B are phase-shifted by 90° electrical. The differential signal has an amplitude of typically 1 Vpp. The number of periods per turn can be configured.

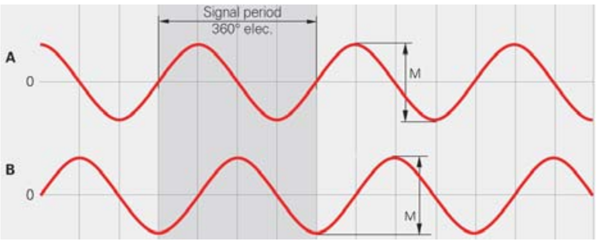


Figure 9-97 Sinus Incremental Encoder Principle

9.3.3.2 Implementation

A sinus incremental encoder can be connected to the EPOS3 70/10 EtherCAT's digital inputs DigIN7 and DigIN8.

If the supply voltage of the SSI sensor is 5 V and the current is less than 150 mA, it can be directly supplied from the $+V_{AUX}$ signal (J5-9). Otherwise, an external power supply must be connected to power the sensor.

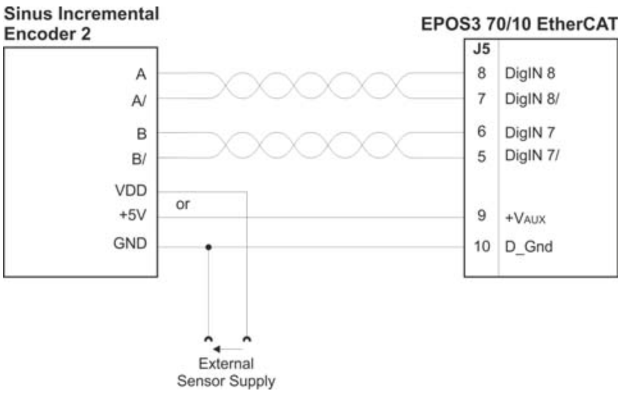


Figure 9-98 EPOS3 70/10 EtherCAT – Sinus Incremental Encoder Connection

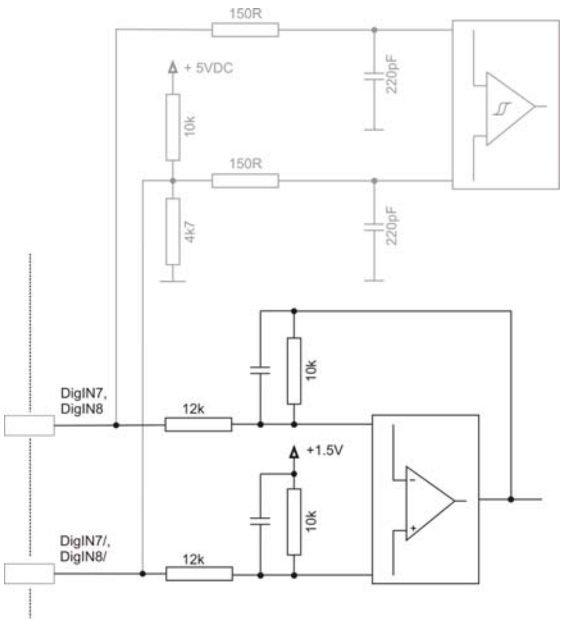


Figure 9-99 EPOS3 70/10 EtherCAT – DigIN7/DigIN8 “Differential” Input Circuit of Sinus Incremental Encoder 2

9.3.3.3 Choice of Manufacturers for Sinus Incremental Encoder 2

Manufacturer	Contact
Baumer	Baumer Electric AG, CH-8501 Frauenfeld www.baumerelectric.com
Heidenhain	DR. JOHANNES HEIDENHAIN GmbH, DE-83292 Traunreut www.heidenhain.de
Hengstler	HENGSTLER GmbH, DE-78554 Aldingen www.hengstler.com
and others	

Table 9-93 Sinus Incremental Encoder 2 – Manufacturers (not concluding)

9.4 Configuration Objects

Note
The subsequent information is an extract of the ➔separately available document «EPOS3 EtherCAT Firmware Specification» showing the configuration objects for the extended encoders.

- *Some combinations of sensors can only be configured if the controller structure is set to “1” (velocity auxiliary controller).*
- *With a single loop structure, the main sensor will be used regardless if it is mounted to the motor or to the load.*

9.4.1 Controller Structure

Description

Used to define the dual loop controller structure. Without auxiliary controller, the structure is single loop.

Remarks

If a controller structure will be set to a value that is in conflict with the actual position sensor type, the sensor type will be set to “0” (Unknown sensor).

Can only be changed in “Disable” state.

Name	Controller Structure
Index	0x2220
Subindex	0x00
Type	UNSIGNED16
Access	RW
Default Value	–
Value Range	➔Table 9-94

Value	Description
0	no auxiliary controller
1	velocity auxiliary controller

Table 9-94 Controller Structure

9.4.2 Sensor Configuration

Name	Sensor Configuration
Index	0x2210
Number of entries	4

Description

Used to define the main and the auxiliary controller's sensor type.

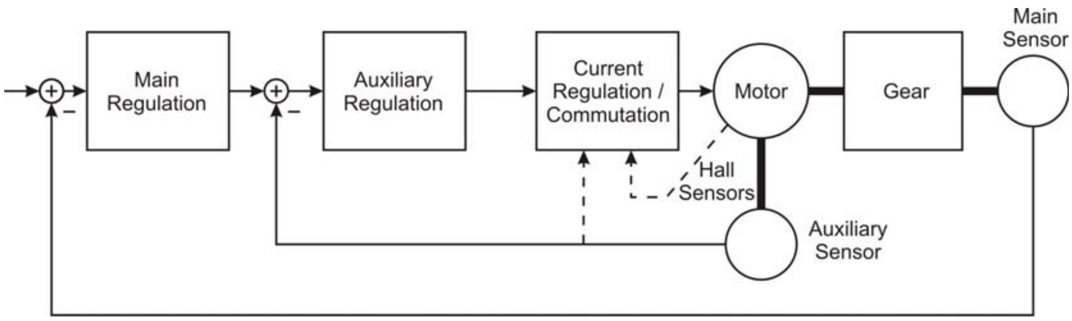


Figure 9-100 Regulation, Sensor and Gear Overview

Name	Position Sensor Type
Index	0x2210
Subindex	0x02
Type	UNSIGNED16
Access	RW
Default Value	0x01
Value Range	➔Table 9-95 and Table 9-96

Bit	Description
15…12	reserved (0)
11…8	Sensor type of auxiliary controller
7…4	reserved (0)
3…0	Sensor type of main controller

Table 9-95 Position Sensor Type – Bits

Value	Description	Abbreviation
0	Unknown sensor (undefined)	–
1	Incremental Encoder 1 with index (3-channel)	Inc Enc1
2	Incremental Encoder 1 without index (2-channel)	
3	Hall Sensors (Remark: consider worse resolution)	Hall
4	Absolute encoder SSI	SSI
5	reserved	–
6	Incremental Encoder 2 with index (3-channel)	Inc Enc2
7	Incremental Encoder 2 without index (2-channel)	
8	Sinus Incremental Encoder 2	Sin Inc Enc2

Table 9-96 Supported Sensor Types

Description

Used to change the position sensor polarity.

Remarks

Can only be changed in “Disable” state.

The absolute position may be corrupted after changing this parameter.

Name	Position Sensor Polarity
Index	0x2210
Subindex	0x04
Type	UNSIGNED16
Access	RW
Default Value	0x00
Value Range	➔Table 9-97

Bit	Value	Name	Description
0	0	Incremental Encoder 1	normal Enc1 polarity (CCW counts positive)
	1		inverted Enc1 polarity (or encoder mounted on motor shaft)
1	0	Hall sensors	normal Hall sensor polarity (maxon standard)
	1		inverted Hall sensor polarity
2	0	SSI Encoder	normal SSI polarity (CCW counts positive)
	1		inverted SSI polarity
3	0	Incremental Encoder 2	normal Enc2 polarity (CCW counts positive)
	1		inverted Enc2 polarity (or encoder mounted on motor shaft)
4	0	Sinus Incremental Encoder	normal Enc2Sin Encoder polarity (CCW counts positive)
	1		inverted Enc2Sin Encoder polarity
5…15	(0)	reserved	–

Table 9-97 Position Sensor Polarity

9.4.3 SSI Encoder Configuration

Description

Used to configure the interpretation of the SSI Encoder.

Remark

Changes are only supported in “Disable” state.

Name	SSI Encoder Configuration
Index	0x2211
Number of entries	4

Description

SSI data rate (SSI clock frequency) in [kbit/s].

Remark

The maximal data rate depends on the actual line length and the employed SSI encoders’ specifications. Typically are 400 kbit/s for cable lengths <50 m.

Name	SSI Encoder Datarate	
Index	0x2211	
Subindex	0x01	
Type	UNSIGNED16	
Access	RW	
Default Value	500	
Value Range	400	2 000

Description

Defines the number of multi-turn and single-turn bits. The maximal number of bits for both values combined is 32. The resolution is $2^{\text{number of bits single-turn}}$.

Name	SSI Encoder Number of Data Bits
Index	0x2211
Subindex	0x02
Type	UNSIGNED16
Access	RW
Default Value	3085 (0x0C0D)
Value Range	➔Table 9-98

Bit	Name	Value		
		Minimal	Maximal	Default
15…8	number of bits multi-turn	0	26	12
7…0	number of bits single-turn	6	23	13

Table 9-98 SSI Encoder Number of Data Bits

Description

Position received from encoder [Position units] (➔page 1-11).

Name	SSI Encoder Actual Position	
Index	0x2211	
Subindex	0x03	
Type	INTEGER32	
Access	RO	
Default Value	–	
Value Range	–	–

Description

Defines the SSI’s encoding type.

Name	SSI Encoding Type
Index	0x2211
Subindex	0x04
Type	UNSIGNED16
Access	RW
Default Value	0
Value Range	➔Table 9-99

Value	Description
0	SSI Encoder binary type
1	SSI Encoder Gray coded

Table 9-99 SSI Encoding Type

9.4.4 Incremental Encoder 2 Configuration

Description

Used to configure the interpretation of the Incremental Encoder 2.

Remarks

Can only be changed in “Disable” state.

The absolute position may be corrupted after changing this parameter.

Name	Incremental Encoder 2 Configuration
Index	0x2212
Number of entries	3

Description

The encoder’s pulse number must be set to number of pulses per turn of the connected Incremental Encoder.

Name	Incremental Encoder 2 Pulse Number	
Index	0x2212	
Subindex	0x01	
Type	UNSIGNED32	
Access	RW	
Default Value	500	
Value Range	16	2 500 000

Description

Holds the internal counter register of the Incremental Encoder 2. It shows the actual encoder position in quad counts [qc].

Name	Incremental Encoder 2 Counter	
Index	0x2212	
Subindex	0x02	
Type	UNSIGNED32	
Access	RO	
Default Value	–	
Value Range	–	–

Description

Holds the Incremental Encoder 2 counter reached upon last detected encoder index pulse. It shows the actual encoder index position in quad counts [qc].

Name	Incremental Encoder 2 Counter at Index Pulse	
Index	0x2212	
Subindex	0x03	
Type	UNSIGNED32	
Access	RO	
Default Value	–	
Value Range	–	–

9.4.5 Sinus Incremental Encoder 2 Configuration

Description

Used to configure the Sinus Incremental Encoder 2 Configuration's interpretation.

Remarks

Can only be changed in "Disable" state.

The absolute position may be corrupted after changing this parameter.

Name	Sinus Incremental Encoder 2 Configuration
Index	0x2213
Number of entries	2

Description

Defines the resolution of "Sinus Incremental Encoder 2". The parameter pulses per turn must be set to the number of pulses per revolution of the connected Sinus Incremental Encoder.

This value multiplied by $2^{\text{number of interpolation bits}}$ is the total resolution of the Sinus Incremental Encoder.

The values are further limited as follows:

Max. resolution: $2^{\text{number of interpolation bits}}$ * pulses per turn ≤ 10 000 000

Min. resolution: $2^{\text{number of interpolation bits}}$ * pulses per turn ≥ 64

Name	Sinus Incremental Encoder 2 Resolution
Index	0x2213
Subindex	0x01
Type	UNSIGNED32
Access	RW
Default Value	0x00800006
Value Range	➔Table 9-100

Bit	Name	Value		
		Minimal	Maximal	Default
15…8	pulses per turn	1	2 500 000	2048
7…0	number of interpolation bits	2	10	4

Table 9-100 Encoder 2 Resolution

Description

Position received from Sinus Incremental Encoder [Position units] (➔page 1-11).

Name	Sinus Incremental Encoder 2 Actual Position	
Index	0x2213	
Subindex	0x02	
Type	INTEGER32	
Access	RO	
Default Value	–	
Value Range	–	–

9.5 Application Examples

Best Practice

The system should work correct if you employ components as listed and configure them as described.

If not the case, check the objects' configuration after executing the described wizards and adjust/tune them according to the actual components employed.

9.5.1 Example 1: Single Loop DC Motor / Gear / SSI Absolute Encoder

Equipment	Type / Specifications
Controller	maxon motor controller EPOS3 70/10 EtherCAT (411146)
Motor	maxon DC motor (any)
Gear	maxon gear (any) reduction 23:1 (absolute 576:25), recommended input speed <6000 rpm
Absolute SSI Encoder	Baumer BMMH (42S105C 12/13 B25) Coding: Gray Interface Data Rate: 500 kbit/s Singleturn Data Bits: 12 Multiturn Data Bits: 13

Table 9-101 Example 1 – Setup

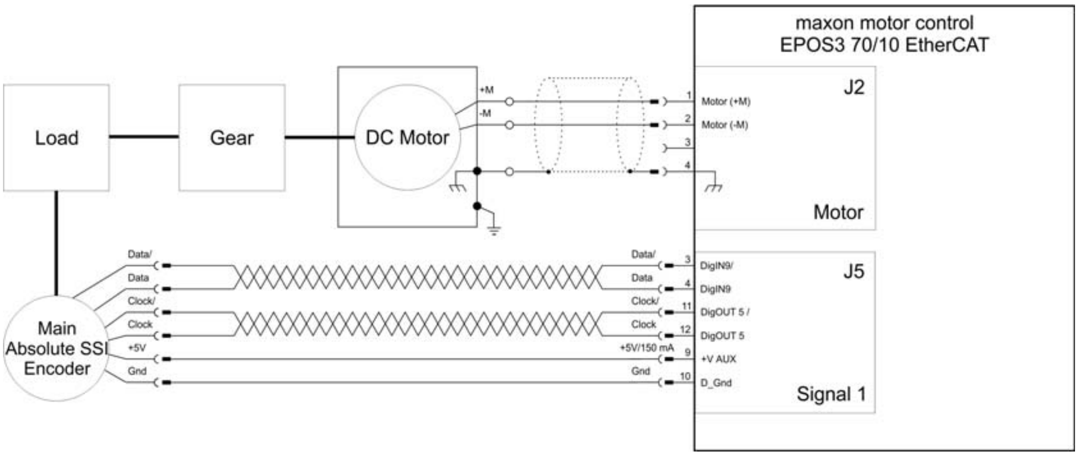
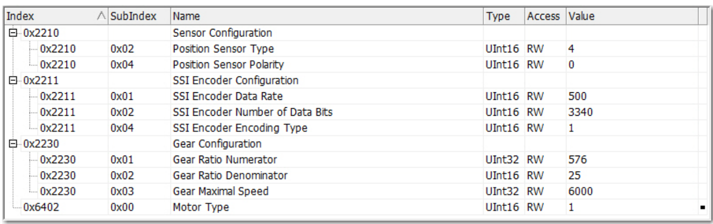


Figure 9-101 Example 1 – Wiring Diagram

1) Wire the system according to the wiring diagram (➔Figure 9-101).
2) Follow the configuration steps in the "Startup Wizard" of «EPOS Studio».
3) Upon successful configuration, start the "Regulation Tuning Wizard".
4) Now your system is ready to use.
   For verification purposes: The related objects should have been set as follows:



Index	SubIndex	Name	Type	Access	Value
0x2210		Sensor Configuration			
0x2210	0x02	Position Sensor Type	UInt16	RW	4
0x2210	0x04	Position Sensor Polarity	UInt16	RW	0
0x2211		SSI Encoder Configuration			
0x2211	0x01	SSI Encoder Data Rate	UInt16	RW	500
0x2211	0x02	SSI Encoder Number of Data Bits	UInt16	RW	3340
0x2211	0x04	SSI Encoder Encoding Type	UInt16	RW	1
0x2230		Gear Configuration			
0x2230	0x01	Gear Ratio Numerator	UInt32	RW	576
0x2230	0x02	Gear Ratio Denominator	UInt16	RW	25
0x2230	0x03	Gear Maximal Speed	UInt32	RW	6000
0x6402	0x00	Motor Type	UInt16	RW	1

Figure 9-102 Example 1 – Object Configuration

9.5.2 Example 2: Dual Loop Incremental Encoder (2 Ch) / EC Motor / Gear / Incremental Encoder (3 Ch)

Equipment	Type / Specifications
Controller	maxon motor controller EPOS3 70/10 EtherCAT (411146)
Motor	maxon EC motor (any)
Encoder	maxon Encoder MR Counts per turn: 1000 inc. Number of Channels: 2 (or 3)
Gear	maxon gear (any) reduction 5.8:1 (absolute 23:4), recommended input speed <8000 rpm
Auxiliary Encoder	Baumer BHF (16.05A 7200-E2-5) Counts per turn: 7200 inc. Number of Channels: 3

Table 9-102 Example 2 – Setup

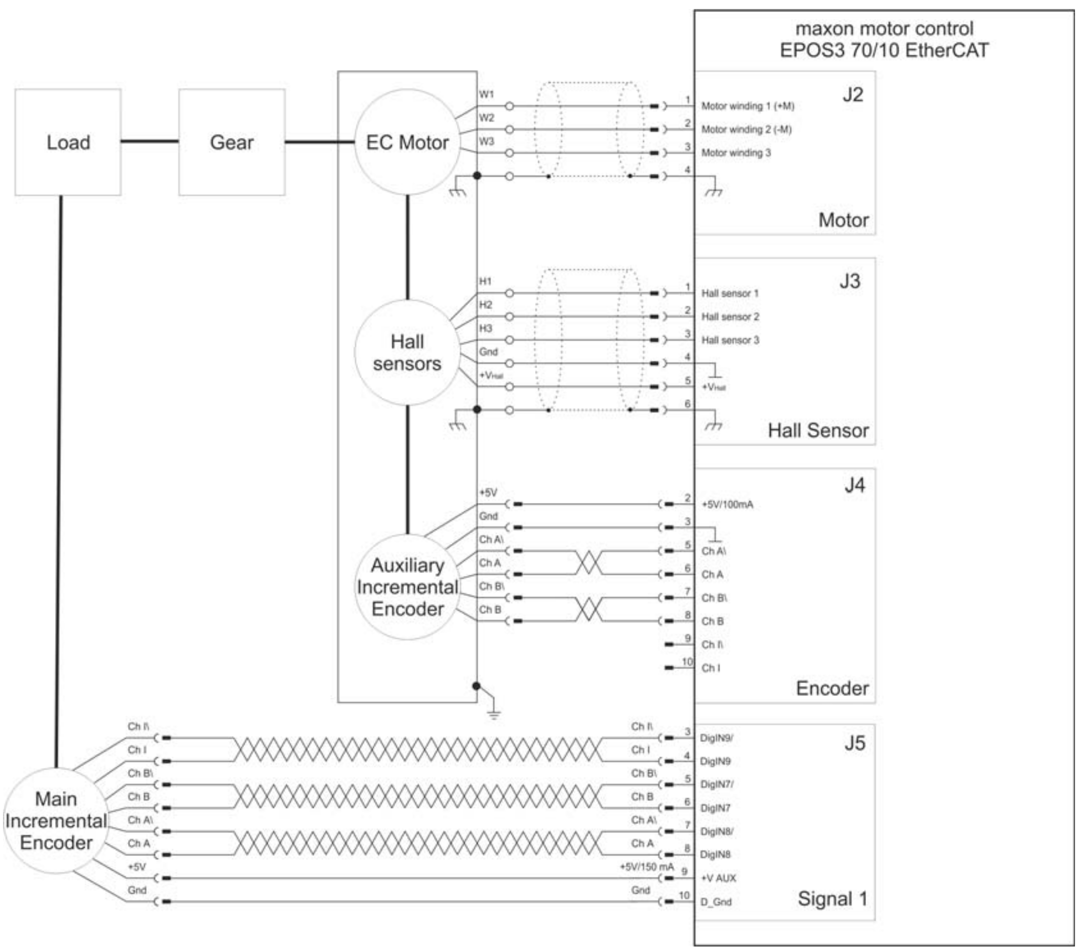


Figure 9-103 Example 2 – Wiring Diagram

1) Wire the system according to the wiring diagram (➔Figure 9-103).
2) Follow the configuration steps in the “Startup Wizard” of «EPOS Studio».
3) Upon successful configuration, start the “Regulation Tuning Wizard”.
4) Now your system is ready to use.
   For verification purposes: The related objects should have been set as follows:

Index	SubIndex	Name	Type	Access	Value
0x2210		Sensor Configuration			
0x2210	0x01	Pulse Number Incremental Encoder 1	UInt32	RW	1000
0x2210	0x02	Position Sensor Type	UInt16	RW	518
0x2210	0x04	Position Sensor Polarity	UInt16	RW	8
0x2212		Incremental Encoder 2 Configuration			
0x2212	0x01	Incremental Encoder 2 Pulse Number	UInt32	RW	7200
0x2220	0x00	Controller Structure	UInt16	RW	1
0x2230		Gear Configuration			
0x2230	0x01	Gear Ratio Numerator	UInt32	RW	23
0x2230	0x02	Gear Ratio Denominator	UInt16	RW	4
0x2230	0x03	Gear Maximal Speed	UInt32	RW	8000
0x6402	0x00	Motor Type	UInt16	RW	10

Figure 9-104 Example 2 – Object Configuration

List of Figures

••page intentionally left blank••

LIST OF TABLES

••page intentionally left blank••

INDEX

A

C

D

E

F

H

I

M

N

O

P

R

S

T

U

V

maxon motor ag
Brünigstrasse 220
P.O.Box 263
CH-6072 Sachseln
Switzerland

Phone +41 41 666 15 00
Fax +41 41 666 16 50

www.maxonmotor.com