

Erste Schritte in Motion Control

Ein Praktikum mit maxon ESCON
und EPOS4 Steuerungen

Urs Kafader, Walter Schmid, Stefan Enz





Erstausgabe 2020

© 2020 und 2017 academy.maxongroup.com, Sachseln

Dieses Werk ist urheberrechtlich geschützt. Alle Rechte vorbehalten, einschliesslich, aber nicht beschränkt auf die Rechte der Übersetzung in Fremdsprachen, der Reproduktion, der Speicherung auf elektronischen Medien, des Nachdrucks und der öffentlichen Präsentation. Die Verwendung von geschützten Namen, gebräuchlichen Namen usw. in diesem Werk bedeutet nicht, dass diese Namen nicht im Sinne des Markenrechts geschützt sind. Alle Informationen in diesem Werk, einschliesslich, aber nicht beschränkt auf numerische Daten, Anwendungen, quantitative Daten usw. sowie Ratschläge und Empfehlungen, wurden sorgfältig recherchiert, obwohl die Genauigkeit dieser Informationen und die völlige Abwesenheit von Druckfehlern nicht garantiert werden kann. Die Genauigkeit der bereitgestellten Informationen muss vom Benutzer in jedem Einzelfall überprüft werden. Der Autor, der Herausgeber und/oder ihre Vertreter können nicht für Personen-, Vermögens- oder Sachschäden haftbar gemacht werden.

Version 1.02, Januar 2021

maxon ESCON und EPOS4 Praktikum

Erste Schritte in Motion Control

Herangehensweise

Der Ansatz dieses Lehrbuchs ist ein praktischer und experimenteller. Anstatt zuerst die Theorie der Bewegungssteuerung zu erklären und sie dann an konkreten Beispielen anzuwenden, beginnen wir hier mit praktischen Experimenten. Unsere Untersuchungen werden mit Hilfe der Software ESCON oder EPOS Studio auf einem maxon ESCON Servo-Kontroller oder EPOS4 Positioniersteuerung durchgeführt. Wir erklären alle relevanten Motion-Control-Prinzipien und Features. Der Text enthält deshalb hauptsächlich die Übungen und praktischen Arbeiten. Hintergrundinformationen finden Sie in den farbigen Kästchen. Ich möchte betonen, dass das "Spielen" mit dem System und das Erleben des Systems wichtige Aspekte des Lernens sind. Nutzen Sie daher die Vorschläge für zusätzliche Übungen!

Motion Control ist Mechatronik. Sie ist die Kombination von Mechanik und Elektronik, von Aktuatoren (Motoren) und Sensoren, die alle durch Software gesteuert werden. In diesem Lehrbuch identifizieren und definieren wir die Rolle, das Verhalten und die gegenseitige Interaktion der verschiedenen Elemente eines Motion-Control-Systems. Allerdings müssen wir den grundlegenden inneren Aufbau und die Arbeitsprinzipien der Elemente nur in begrenztem Umfang kennen, wir betrachten sie eher als Black Box. Was kommt bei einem bestimmten Input aus der Box heraus? Wie wird dieser Output erzeugt und mit welchem Parameter kann der Output beeinflusst werden? Wir werden zum Beispiel nicht erklären, wie das Feed Forward implementiert werden muss, aber wir werden die Grundidee erläutern und wie sich die Parameter beim Tuning verändern.

Kästchen

Die farbigen Kästchen enthalten zusätzliche allgemeine Hintergrundinformationen über Motion Control und Programmierung, sowie spezielle Aspekte über das verwendete System und die Software.

	Hintergrund zu Motion Control Allgemeine Informationen und Grundlagenwissen zu Motion Control Konzepten.
 	ESCON oder EPOS Info Zusätzliche Angaben zu speziellen Eigenschaften und über das Verhalten von <i>EPOS4</i> oder <i>ESCON</i> Systemen.
	Best Practice Tipps und Hinweise, wie man das <i>ESCON</i> oder <i>EPOS4</i> System und das <i>ESCON</i> oder <i>EPOS Studio</i> effizient benützt und bedient.

Inhaltsverzeichnis

maxon ESCON und EPOS4 Praktikum	3
Herangehensweise.....	3
Kästchen.....	3
Inhaltsverzeichnis.....	4
Teil 1: ESCON Vorbereitung.....	6
1 Einrichten des ESCON Starter Kit	6
1.1 Installation der ESCON Studio Software.....	7
1.2 Kabel anschliessen.....	8
1.3 Starten des ESCON Studio	9
1.4 Firmware Update Assistent (optional)	9
1.5 Startup Assistent	10
1.6 Regler Tuning Assistent	12
Teil 2: Erkundung der ESCON-Funktionalitäten	15
2 Betrieb	15
2.1 Eigenständiger Betrieb	15
2.2 Kontroller Überwachung: Übersicht.....	16
2.3 Kontroller Überwachung: Kontroller	16
2.4 Kontroller Überwachung: Eigenschaften	17
2.5 Datenaufzeichnung und Drehzahl	17
3 Eingänge / Ausgänge	22
3.1 Freigabe Optionen.....	22
3.2 Sollwert Optionen	23
3.3 Comparator.....	23
4 Spezielle Betriebsarten (optional)	25
4.1 Offener Regelkreis (Open Loop) Drehzahlsteller	25
4.2 Stromregler.....	25
5 Spezielle ESCON Werkzeuge	27
5.1 Diagnose	27
5.2 Virtueller Kontroller	27
5.3 Standard-Parameter wiederherstellen.....	28
Teil 3: EPOS4 Vorbereitung.....	29
6 Einrichten des EPOS4 Starter Kit	29
6.1 Installation der EPOS Studio Software.....	30
6.2 Kabel anschliessen.....	31
6.3 Die EPOS4 als Black Box und deren Dokumentation.....	32
6.4 Das EPOS Studio starten	35
6.5 Herunterladen der aktuellsten Firmware	39
6.6 Zurücksetzen auf Standardparameter.....	40
7 Einrichten des EPOS4 Motion-Control Systems.....	41

7.1	Startup Wizard.....	41
7.2	Tuning.....	50
Teil 4: EPOS4 Positionsregler.....		58
8	Exploring the Motion Controller	59
8.1	Profile Position Modus.....	59
8.2	Homing	66
8.3	Cyclic Synchronous Position (CSP) Modus	69
8.4	Profile Velocity Modus.....	72
8.5	Cyclic Synchronous Velocity (CSV) Modus	76
8.6	Cyclic Synchronous Torque (CST) Modus.....	78
9	Verwendung des I/O Monitor Werkzeuges	79
9.1	Eingänge	79
9.2	Ausgänge	83
Teil 5: CANopen und EtherCAT Kommunikation.....		84
9.3	CAN	85
9.4	CANopen	85
9.5	CANopen Geräteprofil	85
9.6	EPOS4 Object Dictionary Werkzeug.....	89
9.7	Parameter Import und Export.....	92
9.8	CAN Kommunikation	92
10	Bemerkungen zu EtherCAT.....	94
Teil 6: Anhänge, Referenzen und Index.....		95
11	Anhänge	95
11.1	Motor und Encoder Datenblätter	95
12	Referenzen, Glossar.....	97
12.1	Abbildungsverzeichnis.....	97
12.2	Liste der Kästchen.....	98
12.3	Literature	99
12.4	Index.....	100

Teil 1: ESCON Vorbereitung

Die nächsten beiden Kapitel dienen dem Kennenlernen des Systems und der Vorbereitung des *ESCON* Starter Kit.

Im Kapitel 1 geben wir einen Überblick über die Hardware, gehen auf die Installation der *ESCON Studio* Software ein und erklären kurz die Hauptelemente des Servo-Controller Systems.

Im Kapitel 2 wird das Servo-Controller System für die folgenden Übungen vorbereitet. Die Eigenschaften des Motors und des Encoders müssen definiert und der Regelkreis abgestimmt werden.

1 Einrichten des ESCON Starter Kit

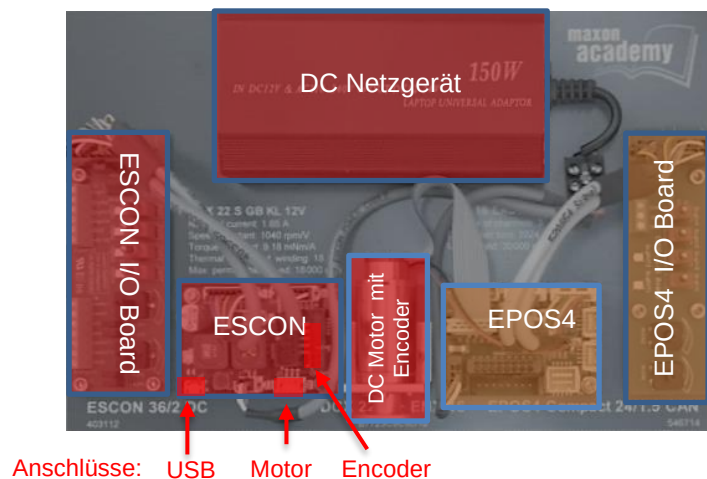


Abbildung 1: Die Komponenten des ESCON/EPOS4 Starter Kit.
Die rot markierten Komponenten werden für das ESCON Training benötigt.

Im Starter Kit enthalten ist ein DC Motor mit einem Encoder, links davon einen *ESCON 36/2 DC* Servo-Controller mit einer angeschlossenen Leiterplatte mit Schaltern, LED Lampen und Potentiometer. Diese Leiterplatte dient als Ein- und Ausgangsschnittstelle (I/O Board) mit digitalen und analogen Eingängen und Ausgängen. Ausserdem ist ein DC Netzgerät für die Stromversorgung enthalten. Die benötigten Kabel befinden sich in der Box unter dem Kit.

Rechts befindet sich eine *EPOS4* Positioniersteuerung mit einer eigenen Ein- und Ausgang Schnittstelle. Die *EPOS4* Steuerung wird im Teil 3 dieses Buches behandelt.

1.1 Installation der ESCON Studio Software

Als Erstes muss der Computer vorbereitet werden. Die Software für das ESCON Studio kann von der maxon Webseite escon.maxongroup.com heruntergeladen und installiert werden. Auf der Produkteseite wird der Knopf *Download ESCON Setup* angeklickt.

Die Installation enthält alle notwendigen Informationen und Werkzeuge (ESCON Studio, Dokumentation und Firmware), welche benötigt werden für den maxon ESCON Servo-Controller. Das *ESCON Studio* ist das Benutzerinterface, welches Zugriff auf alle Parameter und Einstellungen des Servo-Controllers ermöglicht.



Aktuellstes ESCON Studio

Für den Fall, dass Sie ESCON Studio bereits installiert haben, überprüfen Sie die Version. Klicken Sie dazu auf das Menü Hilfe und wählen Sie dann das Menü *Über ESCON Studio*. Falls Ihre Version älter ist als die Version auf der maxon Website (escon.maxongroup.com), deinstallieren Sie die alte Version und laden Sie die neuste Version von der maxon Website herunter. Stellen Sie nach der Installation sicher, dass Sie ESCON Studio 2.2 (Revision 5) oder höher verwenden.

Installation

Schritt 1: Folgen Sie den Anweisungen, die beim Ausführen des Installationsprogramms gegeben werden. Bitte lesen Sie jede Anleitung sorgfältig durch. Geben Sie den Ort des Arbeitsverzeichnisses an, wenn Sie dazu aufgefordert werden.



Arbeitsverzeichnis

Wir empfehlen den folgenden Ort als Arbeitsverzeichnis:
C:\Programme (x86)\maxon motor ag
Beachten Sie, dass die Bezeichnung des Programmverzeichnisses je nach der installierten Systemsprache variieren kann.

Schritt 2: Es werden neue Verknüpfungen und Symbole im Startmenü Ihres Computers angelegt. Die Dateien wurden in das Menü *maxon motor ag* kopiert, wo Sie auf das Programm sowie auf die gesamte Dokumentation zugreifen können. Ein Klick auf die *ESCON Studio* Verknüpfung auf Ihrem Desktop startet das Programm.



ESCON Studio Sprachauswahl

Das ESCON Studio unterstützt verschiedene Sprachen, die bei der Installation des EPOS Studios ausgewählt wurden. Sie können diese jedoch später ändern.

Im Menü Datei wählen Sie Optionen und dann Sprache oder das entsprechende Wort in der aktuellen Studio-Sprache. Wählen Sie die Sprache aus der Dropdown-Liste und starten Sie das ESCON Studio für die zu aktivierende Sprache neu.

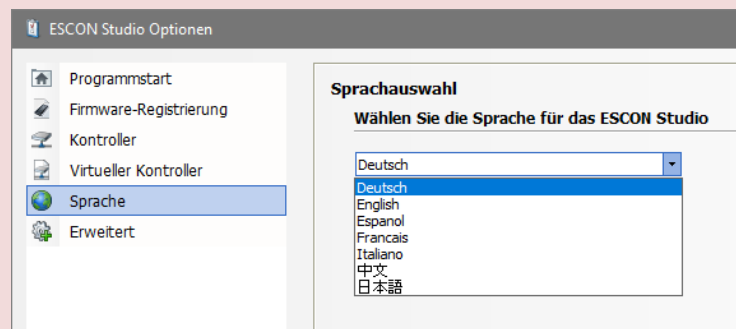


Abbildung 2: Sprachauswahl

1.2 Kabel anschliessen

Die Kabel anzuschliessen ist einfach und grundsätzlich fehlersicher. Schliessen Sie zunächst den DCX-Motor (2 Drähte: rot, schwarz mit einem kleinen schwarzen Stecker) an die einzige passende weiße Buchse an. Dann machen Sie dasselbe mit dem Encoder-Flachbandkabel. Die Platine mit den I/O's ist bereits angeschlossen. Schliesslich - und es ist gute Ingenieurpraxis, dies zuletzt zu tun - schliessen Sie die Stromversorgung an. Zum Schluss stecken Sie das USB-Kabel in einen der USB-Ausgänge Ihres Computers und in den *ESCON Micro USB*-Stecker. Ihr Setup sollte dann ähnlich aussehen wie in Abbildung 3.



Abbildung 3: Anschliessen des ESCON Starter Kit.

1.3 Starten des ESCON Studio

Verbinden Sie die ESCON über USB mit Ihrem Computer. Starten Sie das *ESCON Studio*, indem Sie auf die entsprechende Verknüpfung auf Ihrem Desktop doppelklicken. Der *Startup-Assistent* wird automatisch gestartet. Beenden Sie ihn, indem Sie auf Abbrechen klicken.

Überprüfen Sie den aktiven Controller, der in der Dropdown-Liste oben angezeigt wird. Er sollte *ESCON 36/2 DC - USB0* wie in Abbildung 4, eventuell mit einer anderen USB-Port-Nummer. Wenn ein anderer aktiver Controller angezeigt wird, wählen Sie *ESCON 36/2 DC - USB0* aus der Liste.

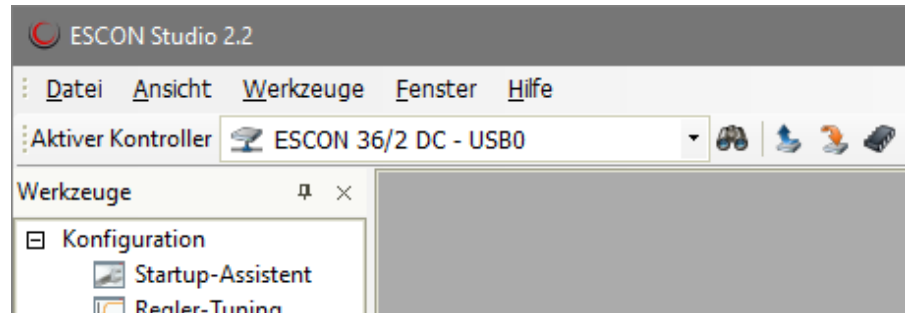


Abbildung 4: Aktiver Controller Auswahlmenü

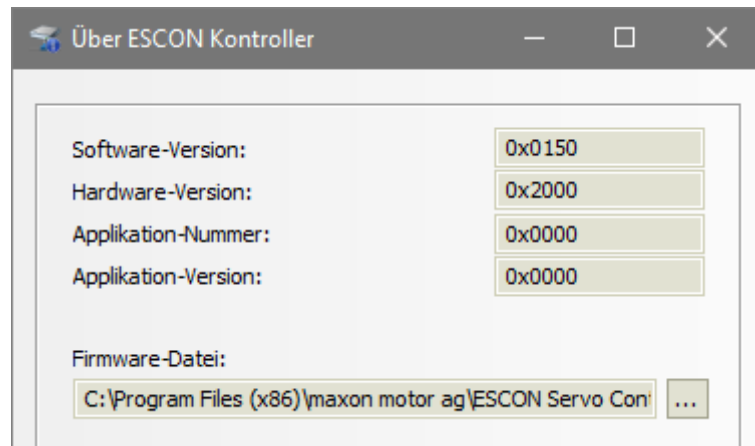
1.4 Firmware Update Assistent (optional)

Zielsetzung: Herunterladen der aktuellen Firmware-Version auf die ESCON.

Überprüfen Sie die aktuelle Firmware der ESCON (klicken Sie auf *Hilfe - Über ESCON Controller*) und vergleichen Sie diese mit der neuesten im Standard-Dateiverzeichnis C:\Programdateien (x86)\maxon motor ag\ESCON Servo Controller\ESCON 36-2 DC\Firmware

Wenn die aktuelle Firmware nicht die neueste verfügbare sein sollte, starten Sie den *Firmware-Update* Assistenten im Konfigurations-Abschnitt des Werkzeuge-Fensters auf der linken Seite

- Schritt 1 Lesen Sie den Warnhinweis durch und bestätigen Sie. Klicken Sie auf *Weiter*.
- Schritt 2 Wählen Sie die Update-Optionen für den Firmware- und Parameter-Update-Modus. (Verwenden Sie in der Regel die Standardeinstellungen: Update mit neuester installierter Firmware und Standard-Parameter). Klicken Sie auf *Weiter*.
- Schritt 3 Wählen Sie die neue Firmware Version. Klicken Sie auf *Weiter*.
- Schritt 4 Starten Sie den Download mit dem *Start* Knopf.
- Schritt 5 Klicken Sie nach dem erfolgreichen Firmware-Update auf *Beenden*.



motor ag > ESCON Servo Controller > ESCON 36-2 DC > Firmware	
☐ Name	Date modified
History	14.02.2020 10:07
Escon_0150h_2000h_0000h_0000h.bin	13.12.2018 11:13

Abbildung 5: Firmware Versionsvergleich
 Oben: Die installierte Firmware Version auf der ESCON wie im Hilfe Menü angezeigt.
 Unten: Die aktuellste Firmware Version
 Beide zeigen die selbe Software- und Hardware-Version an.

Anmerkung: Sehen Sie sich auch das ESCON Video-Tutorial über das Firmware-Update-Tool in der Registerkarte Videos auf der Produktseite des ESCON 36/2-Kontrollers an auf: www.escon.maxongroup.com.

1.5 Startup Assistent

Zielsetzung: Konfigurieren Sie das *ESCON 36/2 DC*-System mit *Drehzahlregelung* für diesen speziellen DC-Motor mit Inkremental-Encoder Feedback. Richten Sie einen externen *Freigabe* Schalter und ein externes Potentiometer als *analogen Sollwert* ein.

- Schritt 1 *Startup-Assistent:* Wenn der Startup-Assistent nicht automatisch starten sollte, doppelklicken Sie auf *Startup-Assistent* in den *Werkzeugen* auf der linken Seite.
- Schritt 2 *Sicherheitshinweise:* Lesen Sie die Sicherheitshinweise und bestätigen Sie diese. Drücken Sie auf *Weiter*.
 Bemerkung: Wenn Sie auf Hilfe klicken, öffnet sich der entsprechende Abschnitt des Dokuments *Hardware-Referenz*.
- Schritt 3 *Einleitung:* Vergewissern Sie sich, dass die Beschreibung des Controllers mit der von Ihnen angeschlossenen übereinstimmt. Drücken Sie auf *Weiter*.
- Schritt 4 *Motor-Daten:* Geben Sie die Motorparameter des DCX 22 S Motors ein. Die entsprechenden Motordaten finden Sie direkt auf der Platine oder im Anhang 12.1 (Motor und Encoder Datenblätter). Klicken Sie auf *Weiter*.

- Schritt 5 *System-Daten*: Geben Sie die System-Eigenschaften ein. Klicken Sie auf *Weiter*. Beachten Sie: Drehzahl und Strom (oder das entsprechende Drehmoment) können durch jede Komponente des Antriebssystems begrenzt sein: Motor, Getriebe, Mechanik, Stromversorgung.... In unserem Fall sind die maximale Drehzahl und der Nennstrom Grenzen des Motors; der maximale Strom wird durch den ESCON Controller begrenzt.
- Schritt 6 *Drehzahlgeber*: Wählen Sie: *Digitaler Inkremental-Encoder*. Auch hier finden Sie die Encoder-Auflösung (1024 Impulse/Umdrehung) auf dem Kit aufgedruckt. Die *Encoder-Drehrichtung* ist gemäss dem maxon Standard. Klicken Sie auf *Weiter*.
- Schritt 7 *Betriebsart*: Wählen Sie Drehzahlregler und *Unterlagerter Stromregler (Standard)* als Betriebsart. Klicken Sie auf *Weiter*.
- Schritt 8 *Freigabe*: Wählen Sie *Freigabe* aus dem Scroll-Down Menü und wählen Sie den *Digitaleingang 2* mit *Hoch (High)* aktiv für die Freigabe Funktionalität. Klicken Sie auf *Weiter*.
- Schritt 9 *Sollwert*: Wählen Sie *Analoger Sollwert* und *Analogeingang 1* für die Sollwert Funktionalität. Geben Sie die gewünschte Skalierung der Sollwerte ein, z.B. min. Drehzahl -1000 min^{-1} und max. Drehzahl $+1000 \text{ min}^{-1}$. Beachten Sie: Der Spannungsbereich des externen Potentiometers (*Analogeingang 1*) beträgt ca. 0...5 V. Klicken Sie auf *Weiter*.
- Schritt 10 *Strombegrenzung*: Wählen Sie *Fixe Strombegrenzung* für unsere einfache Konfiguration und setzen Sie diese auf 2.0000 A. Klicken Sie auf *Weiter*.
- Schritt 11 *Drehzahlrampe*: Wählen Sie *Fixe Rampe* für unsere Konfiguration und setzen Sie die gewünschten Werte für Beschleunigung und Verzögerung, z.B. $500 \text{ min}^{-1}/\text{s}$. (Dies ist ein sehr tiefer Wert, aber er erlaubt uns, die Beschleunigung zu verfolgen.) Klicken Sie auf *Weiter*.
- Schritt 12 *Offset*: Für eine einfache Grundkonfiguration belassen Sie den Wert auf 0.0 min^{-1} . Klicken Sie auf *Weiter*.
- Schritt 13 *Digitaleingänge/ -ausgänge*: Wählen sie die Stopp Funktionalität für den Digitaleingang 1. Klicken Sie auf *Weiter*.
- Schritt 14 *Analogeingänge*: Für unsere einfache Grundkonfiguration ist keine weitere Konfiguration der Analogeingänge nötig. Klicken Sie auf *Weiter*.
- Schritt 15 *Analogausgänge*: Für unsere einfache Grundkonfiguration ist keine weitere Konfiguration der Analogausgänge nötig. Klicken Sie auf *Weiter*.
- Schritt 16 *Digitaleingang 1 - Stopp*: Setzen Sie eine relativ hohe Verzögerung, z.B. $20'000 \text{ min}^{-1}/\text{s}$. Klicken Sie auf *Weiter*.
- Schritt 17 *Zusammenfassung der Konfiguration*: Überprüfen Sie Ihre Einstellungen.
- Schritt 18 *Verdrahtung anzeigen*: Klicken Sie auf *Verdrahtungsübersicht anzeigen* (oder über *Datei* → *Verdrahtungsübersicht*).
Anmerkung: Weitere Informationen finden Sie auch im Kapitel 4 der *Hardware Reference*.
- Schritt 19 *Beenden der Konfiguration*: Klicken sie auf *Beenden* um die Konfiguration auf den ESCON Controller zu speichern.



Motor- und Sensorparameter der vorliegenden Einheit

Der verwendete Motor ist ein DCX 22 S, 12 V Nennspannung, mit Graphitbürsten und Kugellager. Seine genaue Spezifikation ist auf der [maxon Webseite](http://www.maxon.ch) oder im [Anhang](#) aufgeführt. Wichtig für uns ist die Tatsache, dass es sich um einen bürstenbehafteten DC-Motor handelt.

Dieser Motor hat eine maximale Drehzahl von $18'000 \text{ min}^{-1}$ und liegt damit unter dem, was wir mit der gegebenen Spannung der Stromversorgung (24 VDC) erreichen können. Typischerweise ist der Motor in der Lage, bei 24 V Versorgung Drehzahlen von bis zu ca. $24'400 \text{ min}^{-1}$ zu erreichen (Drehzahlkonstante des Motors mal Motorspannung = $1040 \text{ min}^{-1}/\text{V} * 0.98 * 24\text{V}$).

Der Nennstrom beträgt 1,65 A. Der Motor kann diesen Strom kontinuierlich ohne Überhitzung ziehen. Für einige Sekunden kann er einen viel höheren Strom aufnehmen, möglicherweise bis zu 10 A. Wir sind jedoch nicht in der Lage, so viel Strom aus unserer ESCON 36/2 DC herauszuholen, welche eine Strombegrenzung von 4 A aufweist.

Der Encodertyp ist ENX 16 EASY-Encoder mit 1024 Impulsen pro Umdrehung auf jedem der beiden Kanäle. Daraus ergeben sich 4096 Signalfanken (Zustände) pro Umdrehung der Motorwelle, die als Inkremente (inc) bezeichnet werden. Die inc sind die internen Positionseinheiten, die von den EPOS-Steuerungen verwendet werden. Daher haben wir eine Nennaufösung von $1/4096$ einer Umdrehung oder $0,088^\circ$.

1.6 Regler Tuning Assistent

Bemerkung: Schauen Sie auch das ESCON Video Tutorial auf der Produktseite des ESCON 36/2 Kontrollers auf www.escon.maxongroup.ch in der Registerkarte Videos über den *Auto Tuning Modus*. Wenn Sie während dem Tuning auf einen Fehler stossen, versuchen Sie es mit dem *Diagnose Assistenten* (siehe Kapitel 5.1).

Ziel: Erfahren Sie, wie das Auto Tuning in ESCON-Systemen durchgeführt wird. Bereiten Sie die optimale Systemreaktion auf Sollwertbefehle vor.

- Schritt 1 Wenn das *Regler-Tuning* nach der Konfiguration nicht automatisch startet, doppelklicken Sie auf *Regler-Tuning* im Fenster "Werkzeuge" auf der linken Seite.
- Schritt 2 *Tuning-Methode*: Wählen Sie die *Auto Tuning* Methode. Drücken Sie auf *Weiter*.
- Schritt 3 *Auto Tuning*: Drücken sie *Start*. Lesen Sie die Warnmeldung aufmerksam durch. Stellen Sie sicher, dass die Motorwelle sich frei drehen kann. Drücken auf *Ja*, um das Auto Tuning zu starten.
- Schritt 4 Nach erfolgreichem Abschluss des Auto-Tuning wechseln alle Statusleisten auf grün. Klicken Sie auf *Beenden*, um die Tuning-Parameter dauerhaft zu speichern.

Ihr System ist nun betriebsbereit. Aktivieren Sie den Eingangsschalter *Freigabe*, um den Motor laufen zu lassen und die Drehzahl mit dem Sollwertpotentiometer zu ändern.



ESCON Auto Tuning

Für jeden Regelkreis (Strom, geschlossener und offener Drehzahlregler), eine 2-Schritt Prozedur wird ausgeführt:

1. Zuerst werden die optimalen Regelparameter identifiziert. Während dieses Vorgangs schwingt die Motorwelle. Die entsprechenden roten Statusbalken bewegen sich.
2. Zweitens werden die identifizierten Kontrollparameter durch Auswertung einer Sprungantwort verifiziert. Die Motorwelle bewegt sich, und die sich bewegenden Statusbalken wechseln auf grün. Das Ergebnis der Auswertung wird grafisch dargestellt.

Option: Expert Tuning

Expert Tuning ist nicht Teil dieser ESCON-Grundausbildung. Zögern Sie jedoch nicht, sie ebenfalls zu erforschen. Sehen Sie sich auch das ESCON Video Tutorial auf der Produktseite des ESCON 36/2 Controllers auf <http://www.maxongroup.ch> auf der Registerkarte Videos über die *Expert Tuning Methode* an.

- Schritt 1 Starten Sie den *Regler-Tuning Assistenten* mit Doppelklick auf *Regler-Tuning* im *Werkzeuge* Fenster auf der linken Seite.
- Schritt 2 *Tuning-Methode*: Wählen Sie Expert Tuning. Drücken Sie auf *Weiter*.
- Schritt 3 *Tuning Modus*: Wählen Sie Drehzahlregler (Unterlagerter Stromregler). Drücken Sie auf *Weiter*.
- Schritt 4 *Expert Tuning*: Aktivieren Sie beide *Identifizieren* Checkboxes. Die grünen Anzeigen für Drehzahl und Strom oben werden rot. Klicken Sie auf *Start*. Beachten Sie die Warnmeldung. Klicken Sie auf *Ja*, um das Tuning zu starten. Das *Expert Tuning* wird nun starten (siehe Kästchen unten).
- Schritt 5 Stellen Sie unterschiedliche *Regler-Steifigkeiten* ein und beobachten Sie, wie sich die Parameterwerte und das Ergebnis der Verifikation ändern. Beachten Sie: es ist nicht notwendig, den Schritt der *Identifizierung* zu wiederholen.
- Schritt 6 Wenn Sie mit dem Tuningergebnis zufrieden sind, klicken Sie auf *Beenden*, um die Tuning-Parameter zu speichern.

Ihr System ist nun betriebsbereit.



ESCON Expert und manuelles Tuning

Expert Tuning

Beim Tuning des Strom- und Drehzahl-Regelkreises werden 3 Schritte ausgeführt:

1. Zunächst werden die Eigenschaften des Systems (Reibung, Trägheit) bewertet. Während dieses Vorgangs schwingt die Motorwelle. Die entsprechenden roten Statusbalken bewegen sich. In der Expertenabstimmung können Sie diesen Schritt beeinflussen, indem Sie eine *Amplitude* für die Identifikation einstellen.
Bemerkung: Wählen Sie die Amplituden nicht zu gross, da dies zu einer starken Systemreaktion führt. Die gewählte Drehzahlamplitude sollte jedoch eine typische Drehzahländerung Ihrer Anwendung widerspiegeln.
2. Zweitens werden aus dem Ergebnis des ersten Schrittes die optimalen Steuerungsparameter ermittelt. Diese Parametrisierung kann beeinflusst werden, indem die Schieberegler für die *Regler-Steifigkeit* mehr auf die weiche oder harte Seite eingestellt werden. Es ist nicht erforderlich, den Identifikationsschritt zu wiederholen, wenn verschiedene Parametrisierungen ausgewertet werden.
Anmerkungen: Sie können die ermittelten Regler-Parameter überprüfen, indem Sie auf *Zeige Parameter* klicken. Hier könnten Sie die Parameter auch manuell ändern. Die genaue Bedeutung der ESCON-Steuerparameter ist geistiges Eigentum von maxon und wird nicht bekannt gegeben.
3. Der letzte Schritt verifiziert die Abstimmung durch Ausführen einer Sprungantwort. Das Ergebnis wird im *Verifizierungsfenster* angezeigt. Nach erfolgreichem Abschluss der Abstimmung wechseln alle Statusbalken auf grün.
Bemerkung: Sie können in die Verifizierungskurven zoomen, indem Sie den interessanten Bereich mit der Maus auswählen. Ein Rechtsklick setzt das ursprüngliche Diagramm zurück.

Manuelles Tuning

Sollten Sie mit dem Abstimmungsergebnis nicht zufrieden sein oder auf ein Problem oder einen Fehler stossen, können Sie die Regler-Parameter manuell ändern. *Start* führt dann einen Verifizierungstest mit den manuell eingestellten Parametern durch. (Es erfolgt natürlich keine Identifizierung und automatische Parametrierung)

Teil 2: Erkundung der ESCON-Funktionalitäten

In Teil 1 haben wir das System konfiguriert, den Motion Controller abgestimmt und alle Einstellungen auf dem Gerät gespeichert. Der ESCON Servo-Kontroller ist nun bereit für weitere Untersuchungen.

- Kapitel 2 geht auf die Hauptfunktionalität des ESCON Studios ein, einschliesslich der Überwachung des Regelverhaltens und der Parametereinstellung.
- Kapitel 3 zeigt spezielle Werkzeuge für die Diagnose und untersucht die mit dem Kontroller verbundenen Ein- und Ausgänge.
- Kapitel 4 untersucht alternative Betriebsarten
- Kapitel 5 enthält Informationen über Spezialwerkzeuge.

Das *ESCON Studio* ist eine grafische Benutzeroberfläche (GUI) zur Einrichtung, Parametrierung und Überwachung der Leistung der ESCON. Die USB Kommunikations-Schnittstelle ist nur für diese Zwecke gedacht; es handelt sich nicht um eine serielle Kommunikation zu einem Mastersystem. Der Betrieb der ESCON wird durch seine digitalen und analogen Eingänge gesteuert (siehe Kapitel 2.1).

Alle folgenden Übungen sind lediglich Ausgangspunkte für eigene Experimente. Meine Empfehlung: Spielen Sie einfach!

2 Betrieb

Zielsetzung: Erfahren Sie, was Sie in der *Kontroller-Überwachung* des ESCON Studios sehen können.

2.1 Eigenständiger Betrieb

Nachdem Sie den *Startup Assistenten* und das *Regler-Tuning* durchgeführt haben, ist Ihre ESCON betriebsbereit. Wie oben erwähnt, wird die ESCON nur durch ihre I/Os gesteuert. Lassen Sie uns untersuchen, wie die konfigurierten Eingänge zum Betrieb des Motors verwendet werden können.

- Schritt 1 Trennen Sie die USB-Verbindung zu Ihrem PC.
- Schritt 2 Aktivieren Sie die Endstufe der ESCON durch *Freigabe (Enable)* des Schalters *Digitaleingang 2* (zweiter von oben auf dem I/O Board, neben der grünen LED). Für eine dauerhafte Freigabe schieben Sie den Schalter nach rechts. Der Motor beginnt sich zu drehen.
- Schritt 3 Deaktivieren (*Disable*) Sie den Motor durch Deaktivieren des Schalters *Digitaleingang 2* (zweiter von oben auf dem I/O Board, neben der grünen LED). Der Motor läuft aus und stoppt.
- Schritt 4 Aktivieren Sie den Motor wieder. Ändern Sie Drehzahl und Richtung mit dem Potentiometer für *Analogeingang 1* auf dem I/O Board. Können Sie den Motor mit dem eingestellten Wert zum Stillstand bringen? Beachten Sie die langsame Reaktion aufgrund der niedrigen Rampeneinstellungen.

- Schritt 5 Stellen Sie eine hohe Drehzahl ein. Aktivieren und deaktivieren Sie die Stopp-Funktion mit dem Schalter *Digitaleingang 1*. Die Abbremsung bis zum Stillstand ist aufgrund der hohen Abbremsrampeneinstellung sehr schnell.
- Schritt 6 Stellen Sie eine hohe Drehzahl ein und erhöhen Sie die Last, indem Sie Ihren Finger leicht auf die drehende Scheibe am Motor drücken. Was passiert mit der Drehzahl?
- Schritt 7 Deaktivieren Sie den Motor.

2.2 Kontroller Überwachung: Übersicht

Das *ESCON Studio* erlaubt die Überwachung aller Einstellungen und Parameter. Bitte verbinden Sie hierzu erneut die USB-Schnittstelle mit Ihrem Rechner.

- Schritt 1 Wählen Sie auf der linken Seite im *ESCON Studio* *Kontroller-Überwachung*. Aktivieren Sie unten die Registerkarte *Übersicht*.
- Schritt 2 Aktivieren Sie auf den I/O-Board den Schalter *Digitaleingang 1 (Stopp)* und beobachten Sie, wie sich seine Darstellung in der *Kontroller-Überwachung* ändert.
Drehen Sie das analoge Eingangspotentiometer 1 (*Sollwert*). Beobachten Sie, wie sich die Anzeige der Eingangsspannung und der entsprechende Drehzahlsollwert ändern.
- Schritt 3 Aktivieren Sie die *ESCON (Freigabe)* Schalter *Digitaleingang 2* auf dem I/O-Board) und stellen Sie verschiedene Drehzahlwerte ein. Beobachten Sie die Reaktion des Motors und die Messwerte im linken Panel (*Eingänge/Ausgänge*), sowie im rechten Panel (*Motor/Sensoren*).
- Schritt 4 Wie reagiert der Motor auf schnelle Sollwertänderungen? Ändern Sie die Beschleunigungs- und/oder Abbremsrampen im zentralen Panel (*Kontroller*) auf sehr niedrige (100 min^{-1}) oder viel grössere ($10'000 \text{ min}^{-1}$) Werte. Wie schnell reagiert der Motor auf abrupte Sollwertänderungen? Verfolgen Sie die Reaktion der Motorwelle und die gemessene Drehzahl in der *Kontroller-Überwachung*.
- Schritt 5 Erzeugen Sie mit dem Finger ein wenig Reibung an der Motorwelle. Wie reagiert der Motorstrom? Was passiert mit der Drehzahl?
- Schritt 6 Erhöhen Sie die Reibung oder blockieren Sie sogar den Motor. Was spüren Sie an der Motorwelle? Wie gross ist der aktuelle Messwert? Was zeigen die Statusanzeigen im zentralen Panel der *Kontroller-Überwachung* an?
- Schritt 7 Aktivieren Sie den digitalen Eingang 1 (*Stopp*) und beobachten Sie die Motorreaktion. Hat die Einstellung der Verzögerungsrampe einen Einfluss auf die Stoppdauer?

2.3 Kontroller Überwachung: Kontroller

Der Reiter *Kontroller* in der *Kontroller-Überwachung* zeigt die Eingänge und die Parameter des Reglers im Detail. Wiederholen Sie die Schritte von vorhin und vergleichen Sie die Anzeigen.

Als Option, erkunden Sie die Parameter in dieser Anzeige. Welche Werte können geändert werden?

2.4 Kontroller Überwachung: Eigenschaften

Zielsetzung: Erfahren Sie, wie Sie Konfigurationsparameter ändern können, ohne den *Startup-Assistenten* durchlaufen zu müssen.

- Schritt 1 Aktivieren Sie Eigenschaften im unteren Bereich der Kontroller-Überwachung Anzeige. Das Dialogfenster *Eigenschaften* wird geöffnet.
Beachten Sie bitte: Der ESCON Kontroller muss zum Ändern von Konfigurationsparametern auf *Gesperrt*, also deaktiviert sein.
- Schritt 2 Ändern Sie einige der Parameter. Beispielsweise den *Sollwert*-Drehzahlbereich, die *Drehzahlrampe* oder den Stopp-Verzögerungswert in den digitalen Ein- und Ausgängen. Drücken Sie zur Bestätigung und zum Speichern der neuen Konfigurationseinstellungen auf *OK*.
- Schritt 3 Betreiben Sie den Motor und überprüfen Sie auf in der Kontroller-Überwachung die neuen Einstellungen.



Werkzeug: Parameter
Die Parameter des Werkzeugs geben einige grundlegende schreibgeschützte Informationen über das Gerät (Name, Serien- und Versionsnummern) sowie Einstell- und tatsächliche Strom- und Drehzahlwerte.
Die vollständige Liste der Parameter kann zur Verfügung gestellt werden, indem Sie in den erweiterten Optionen die Benutzerstufe auf *Expert* ändern.

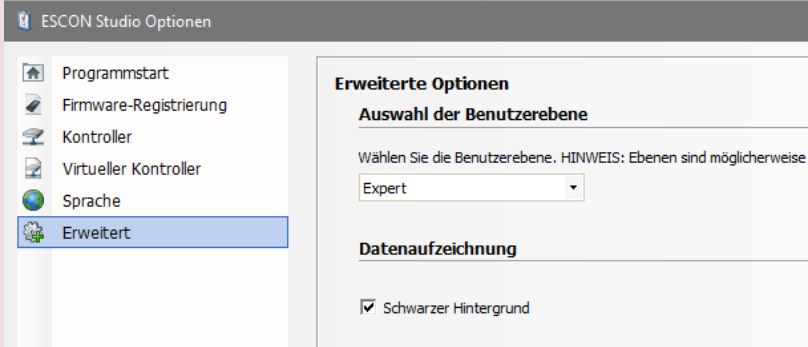


Abbildung 6: Erweiterte Optionen

2.5 Datenaufzeichnung und Drehzahl

Zielsetzung: Erfahren Sie, wie die *Datenaufzeichnung* eingerichtet wird und wie die angezeigten Drehzahlinformationen zu interpretieren sind.

Einrichten der Datenaufzeichnung für die Drehzahlmessung.

- Schritt 1 Öffnen Sie die *Datenaufzeichnung* im Abschnitt *Werkzeuge* auf der linken Seite des *ESCON Studios* (Doppelklick) oder wählen Sie aus dem Hauptmenü *Werkzeuge*.
- Schritt 2 Klicken Sie auf die Schaltfläche *Einstellungen* unten rechts.
- Schritt 3 Aktivieren Sie die Kanäle 1 bis 3. Klicken Sie auf die entsprechenden Schaltflächen. Wählen Sie Kanal 1, um die *Soll-Drehzahl* anzuzeigen, Kanal 2, um die *Ist-Drehzahl gemittelt* anzuzeigen, und Kanal 3 *Ist-Drehzahl*.
- Schritt 4 Deaktivieren Sie für alle drei Kanäle die Kontrollkästchen für die *automatische Skalierung* und stellen Sie die minimale Skalierung auf 850 min^{-1} und die maximale auf 950 min^{-1} ein.
- Schritt 5 Wählen Sie im Abschnitt *Datenerfassung* auf der rechten Seite die minimal mögliche *Abtastzeit* (0,187 ms). Dies entspricht einer Abtastrate des Datenrekorders, welche der Abtastrate des Drehzahlregelkreises entspricht.
- Schritt 6 Stellen Sie die *Trigger-Konfiguration* auf *Kontinuierliche Aufzeichnung* ein. Drücken Sie OK.

Auswertung der Drehzahlinformationen in der Datenaufzeichnung

- Schritt 1 Stellen Sie eine Drehzahl von etwa 900 min^{-1} ein und aktivieren Sie die ESCON.
- Schritt 2 Untersuchen Sie die Signale auf dem Bildschirm des Datenrekorders. Sie sehen ein ähnliches Bild wie in Abbildung 7, oberes Diagramm. Beobachten Sie die Skaleninformationen unten.
- Schritt 3 *Stoppen* Sie die Aufzeichnung, klicken Sie auf *Ist-Drehzahl* im Abschnitt Verfügbare Kurven rechts (was passiert, wenn Sie das Kontrollkästchen deaktivieren und wieder aktivieren?). Ändern Sie die *Skalierung* der *Ist-Drehzahl* auf $25 \text{ min}^{-1}/\text{div}$ und bestätigen Sie mit *Enter*. (Sie können den Offset so lassen, wie er ist.) Das Bild kann nun wie das untere Diagramm in Abbildung 7 aussehen.

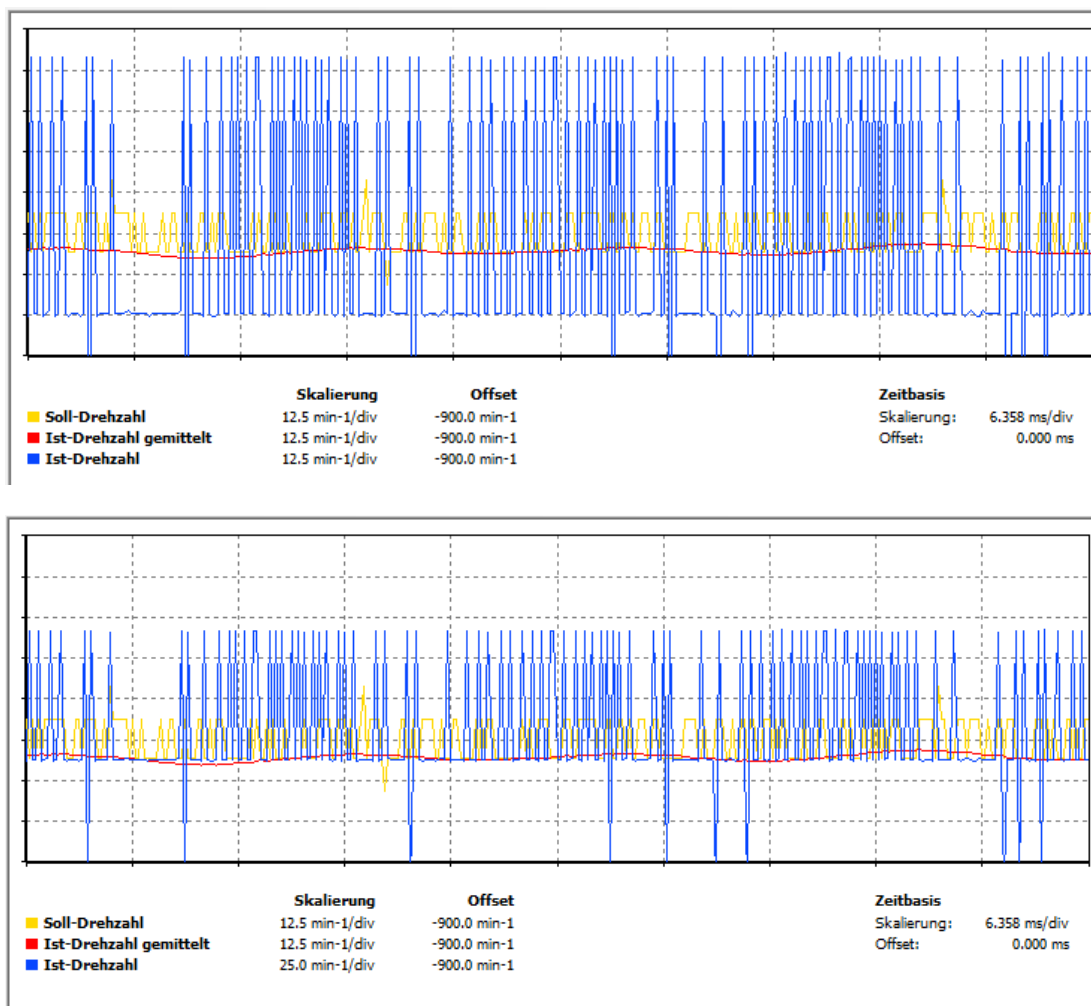


Abbildung 7: Drehzahlssignale auf ESCON-Datenaufzeichnung
 Oben: Alle Kurven mit der gleichen Skalierung
 Unten: Ist-Drehzahlskalierung geändert.

Interpretation der Kurven im Diagramm

- Die *Soll-Drehzahl* (gelbe Kurve) ist nahezu konstant und weist Spitzen von etwa 4 min^{-1} auf. Diese Spitzen spiegeln das Rauschen des externen Potentiometers wider, das als Sollwertquelle fungiert. Sobald das Rauschen die Auflösung des 12-Bit-AD-Wandlers übersteigt, sehen wir eine Spitze.
- Die *Ist-Drehzahl* (blaue Kurve) scheint zwischen 3 Hauptpegeln zu springen, die durch etwas mehr als 3 Teilungen von 25 min^{-1} ($>78 \text{ min}^{-1}$) getrennt sind. Dies können keine echten Drehzahländerungen sein; der Motor kann nicht so schnell reagieren. Was wir stattdessen sehen, ist der Effekt der Drehzahlauswertung durch einen digitalen Inkrementalencoder. Die Drehzahl kann nur in Schritten von ± 1 Encoder-Inkrement pro Abtastperiode gemessen werden, d.h. 1 Inkrement pro 0.187 ms oder etwa $320'000 \text{ Inc/min}$. Der hier verwendete Encoder hat 4096 Inc/Umdrehungen, was zu einer beobachteten Drehzahlauflösung von etwa 78 min^{-1} führt.
- Daher ist die *gemittelte Ist-Drehzahl* (rote Kurve) ein besserer Indikator für die tatsächliche Drehzahl. Sie zeigt vier Schwankungen pro Motorumdrehung von etwa $\pm 3 \text{ min}^{-1}$. Bei 900 min^{-1} dauert eine Motorumdrehung 67 ms ; das ist etwa der gesamte angezeigte Zeitbereich. Vier Fluktuationen pro Umdrehung sind typisch für die hier verwendeten EASY-Encoder. Daher spiegelt die rote Kurve eher die Eigenschaften des Encoders wider als die tatsächlichen Drehzahlschwankungen.



ESCON Datenaufzeichnung Anzeige Optionen

Hintergrundfarbe

Zur besseren Sichtbarkeit können Sie die Hintergrundfarbe der Anzeige mit der rechten Maustaste auf dem Anzeigebereich auf schwarz ändern: *Hintergrund wechseln*.

Cursor

Rechts im Fenster *Datenaufzeichnung* können Sie einen Cursor ein- und ausschalten. Die Werte an der Cursorposition werden auch unterhalb des Diagramms angezeigt.

Datenaufzeichnung im Einzel-Trigger Modus

In analoger Weise kann die Datenaufzeichnung für alle Arten von Signalen eingerichtet werden. Das Verfahren umfasst die gleichen Hauptschritte wie oben. Üben wir dies, um die Systemreaktion beim Aktivieren des *Stopp*-Eingangs zu untersuchen.

- Schritt 1 Öffnen Sie die Einstellungen in der Datenaufzeichnung
- Schritt 2 Aktivieren Sie die Kanäle 1 bis 3 wählen Sie Kanal 1 für die Anzeige der *Soll-Drehzahl*, Kanal 2 für *Ist-Drehzahl*, Kanal 3 für *Ist-Drehzahl gemittelt*
- Schritt 3 Für alle Kanäle, aktivieren Sie die *Automatische Skalierung* Checkbox.
- Schritt 4 Setzen Sie die *Abtastzeit* auf ca. 5 ms . Die *Gesamtzeit* hängt von der Anzahl der aktivierten Kanäle und der gewählten Abtastzeit ab.

- Schritt 5 Stellen Sie die Trigger-Konfiguration auf *Einzel-Trigger* ein und definieren Sie den *Typ* als *Digitalsignal* mit *Stopp* und *Betrieb* -> *Angehalten* als Modus. Stellen Sie eine *Zeitverzögerung* von einigen Prozent oder etwa 500 ms ein. Drücken Sie OK.
- Schritt 6 Aktivieren Sie die ESCON und stellen Sie eine relativ hohe Drehzahl ein. Sie können sehen, dass die *Datenaufzeichnung* ständig die ausgewählten Parameter aufzeichnet.
- Schritt 7 Aktivieren Sie den Schalter Stopp. Die Aufzeichnung stoppt und zeigt die ausgewählten Parameter an. Die Soll-Drehzahl fällt sehr schnell auf Null, wie von der hohen Verzögerungsrampe für den Stopp-Eingang gefordert. Die gemittelte *Ist-Drehzahl* Kurve zeigt, dass der Motor viel länger braucht, um wirklich anzuhalten. Es sieht so aus, als ob der tatsächliche Strom ziemlich klein ist.

3 Eingänge / Ausgänge

In diesem Kapitel untersuchen wir einige der Eingabe- und Ausgabefunktionalitäten ohne Anspruch auf Vollständigkeit.

3.1 Freigabe Optionen

Bislang haben wir eine einfache Eingabe zum Aktivieren und Deaktivieren verwendet. Die Funktionalität besteht darin, die Leistungsendstufe der ESCON mit dem Motor zu verbinden und zu trennen. Es gibt weitere Möglichkeiten, die Freigabe mit Drehrichtung beinhalten.

Freigabe & Drehrichtung

Freigabe & Drehrichtung erlauben zusätzlich die Drehrichtung des Motors zu konfigurieren.

- Schritt 1 Öffnen Sie das *Eigenschaften* Register in der *Kontroller-Überwachung*.
- Schritt 2 Klicken Sie auf *Freigabe* im linken Menü und wählen Sie *Freigabe & Drehrichtung* aus der Dropdown-Liste. Der *Freigabe*-Eingang ist immer noch auf *Digitaleingang 2*, der *Drehrichtung*-Eingang ist auf *Digitaleingang 3*. Drücken Sie *OK*, um die neuen Einstellungen zu speichern.
- Schritt 3 Beobachten Sie das Motorverhalten (Drehzahl und Richtung) für die folgenden Fragen und beobachten Sie die Anzeige in der *Kontroller-Überwachung*.
 - Wie dreht sich die Motorwelle bei positiven oder negativen Drehzahlsollwerten und den 2 Positionen des *Drehrichtung*-Einganges?
 - Was passiert, wenn Sie die Drehrichtung bei laufendem Motor ändern?

Freigabe CW

- Schritt 4 Öffnen Sie das *Eigenschaften* Register erneut und konfigurieren Sie die ESCON mit *Freigabe CW*.
- Schritt 5 Wie dreht sich die Motorwelle bei positiven oder negativen Drehzahlsollwerten. Beobachten Sie auch die Signale in der *Kontroller-Überwachung*.
- Schritt 6 Wie ist die Sollwertpolarität (positiv oder negativ) für die CW-Bewegung?

Freigabe CW & CCW

- Schritt 7 Konfigurieren Sie die ESCON mit *Freigabe CW & CCW*.
- Schritt 8 Wie dreht sich die Motorwelle bei positiven oder negativen Drehzahlsollwerten. Beobachten Sie auch die Signale in der *Kontroller-Überwachung*.
- Schritt 9 Was sind die Bedingungen für die Ausführung eines Freigabebefehls in einer bestimmten Richtung? Was ist der Unterschied zu *Freigabe & Drehrichtung*.



Freigabe und Gesperrt der Leistungs-Endstufe

Die Bezeichnungen *Freigegeben* und *Gesperrt* beziehen sich auf den Status der Leistungs-Endstufe des Kontrollers.

1. *Freigegeben* Die Endstufe des Kontrollers bestromt den Motor. Der Motor bekommt Energie und kann gesteuert werden.
2. *Gesperrt* Die Endstufe ist abgeschaltet, der Motor ist nicht bestromt; es wird kein Drehmoment erzeugt. Der Controller ist aber nach wie vor bestromt.

3.2 Sollwert Optionen

Wir haben einen Analogeingang verwendet, um den Drehzahlsollwert mit einem Potentiometer einzustellen. Es gibt Anwendungen, die immer mit der gleichen Drehzahl oder vielleicht mit zwei verschiedenen festen Drehzahlen laufen, denken Sie z.B. an einen Lüfter mit einer niedrigen und einer hohen Drehzahleinstellung. Feste Drehzahlen können auch in der ESCON eingestellt werden.

- Schritt 1 Öffnen Sie das Dialogfeld *Eigenschaften* am unteren Rand der *Kontroller-Überwachung*. Beobachten: Die ESCON muss *gesperrt* werden. Rekonfigurieren Sie die ESCON auf lediglich *Freigabe*.
- Schritt 2 Ändern Sie den Typ *Sollwert* auf *2 fixierte Sollwerte*. Konfigurieren Sie Digitaleingang 3, um zwischen den 2 Werten hin- und her zu schalten. Definieren Sie 2 Drehzahlwerte, die Sie klar unterscheiden können, z.B. einen negativen niedrigen Wert (z.B. -200 min^{-1} für den *inaktiven Zustand*) und einen positiven hohen Wert (z.B. $+2000 \text{ min}^{-1}$ für den *aktiven Zustand*). Drücken Sie *OK*, um die neuen Konfigurationseinstellungen zu speichern.
- Schritt 3 Betreiben Sie den Motor und überprüfen Sie in der *Kontroller-Überwachung* die neuen Einstellungen.
- Schritt 4 Untersuchen Sie das Verhalten der Drehzahlregelung bei Drehzahländerung. Vergleichen Sie mit den Einstellungen der *Drehzahlrampen*.

3.3 Komparator

Zielsetzung: Informieren Sie sich über die Möglichkeiten, die Drehzahl (oder den Strom) mit Hilfe der Komparator-Funktionalität zu überwachen.

- Schritt 1 Konfigurieren Sie den *Analogeingang 1* auf *Sollwert*.
- Schritt 2 Konfigurieren Sie einen der Digitaleingänge als *Drehzahl-Komparator*.
- Schritt 3 Erkunden Sie nacheinander die verschiedenen Modi: *Limite*, *Bereich* oder *Abweichung*.
Stellen Sie vernünftige Drehzahlbereichsparameter ein, die mit Ihrem Motor kompatibel sind (z.B. ein paar hundert min^{-1}).
Um die Reaktion von Motor und Ausgang leicht verfolgen zu können, wählen Sie Rampen für relativ niedrige Drehzahlen (z.B. $25 \text{ min}^{-1}/\text{s}$).
Wählen Sie die Richtung, die Sie erforschen möchten.

Schritt 4 Ändern Sie den eingestellten Wert und beobachten Sie den Ausgang des LED-Drehzahlkomparators.

Beachten: Die LED auf dem I/O-Board ist nur funktionsfähig, wenn der entsprechende Eingangsschalter aktiviert ist. Ein HOCH- oder Aktiv-Pegel des Ausgangs führt zum Erlöschen der LED. Ein TIEF- oder Inaktiv-Pegel des Ausgangs lässt die LED aufleuchten).



DC Tacho Feedback

Die ESCON kann auch mit einem DC-Tacho als Feedbacksignal betrieben werden. Wählen Sie einfach den entsprechenden *Drehzahlgeber* aus und folgen Sie den Anweisungen:

- Definieren Sie den Analogeingang für den anzuschliessenden Tacho
- Stellen Sie die DC-Tachokonstante ein, d.h. die Gleichspannung pro Drehzahl
- Wählen Sie die DC-Tachorichtung.

4 Spezielle Betriebsarten (optional)

4.1 Offener Regelkreis (Open Loop) Drehzahlsteller

Zielsetzung: Erfahren Sie, dass Open Loop mit der ESCON einer IxR-Kompensation entspricht.

Die ESCON kann eine Drehzahlregelung mit offenem Regelkreis durchführen, d.h. ohne Encoder oder DC-Tacho als Feedbackgeber. Stattdessen erhält die ESCON einfache Drehzahlinformationen aus dem Motorstrom und der angelegten Spannung mit der sogenannten IxR-Kompensationsmethode.

Wir können diese Betriebsart ausprobieren, indem wir unseren DC-Motor ohne Sensor einrichten (verwenden Sie den *Startup-Assistenten* oder die Registerkarte *Eigenschaften*).

- Schritt 1 Konfigurieren Sie die ESCON als *Drehzahlsteller* (Open Loop), d.h. ohne Sensor als Drehzahlgeber.
- Schritt 2 Führen Sie ein *Expert Tuning* durch und untersuchen Sie die Parameter für die Drehzahlabstimmung (sie unterscheiden sich vollständig von den Standard-PI-Parametern im geschlossenen Regelkreis).
- Schritt 3 Vergleichen Sie das Verhalten der Drehzahlregelung mit dem Fall mit Sensor Feedback. Können Sie in der *Datenaufzeichnung* einen Unterschied zur Betriebsart *Drehzahlregler* (geschlossener Regelkreis) fühlen oder sehen oder beobachten?
Beachten Sie: Für die Steuerung mit offenem Regelkreis können Sie den Sensor angeschlossen lassen. Ändern Sie einfach die *Betriebsart* im Abschnitt *Eigenschaften*.

4.2 Stromregler

Zielsetzung: Lernen Sie wie die Stromregelung funktioniert.

ESCON für Stromregler-Modus einstellen.

- Schritt 1 Öffnen Sie den *Startup-Assistenten* oder die Registerkarte *Eigenschaften*. Ändern Sie die Betriebsart auf *Stromregler*.
- Schritt 2 Notieren Sie sich die Parameter: Nennstrom, Thermische Zeitkonstante der Wicklung und Grenzdrehzahl.
- Schritt 3 Konfigurieren des Analogeingang 1 auf *Sollwert*. Definieren Sie den Strombereich von 0 A bei 0 V bis maximal etwa dem 3-fachen Nennstrom des Motors bei 5 V.
- Schritt 4 Achten Sie auf die maximale Strombelastbarkeit Ihrer Stromversorgung.

Erkunde das Motorverhalten im Stromregler-Modus.

- Schritt 1 Stellen Sie einen niedrigen Strom von ca. 50...100 mA ein. Freigeben. Verfolgen Sie die physikalische Motorreaktion und in der *Kontroller-Überwachung*.
 - Wie gross ist die Motordrehzahl?

- Wie gross ist der reale Motorstrom im Vergleich zum eingestellten Wert?
- Sind Statusindikatoren aktiv (mittlerer Abschnitt des *Kontroller-Überwachung* Bildschirms)?

Schritt 2 Was ändert sich, wenn Sie den Sollwert erhöhen?

Schritt 3 Stellen Sie Strom auf Null ein, fixieren Sie die Motorwelle von Hand und erhöhen Sie den eingestellten Wert.

- Was empfinden Sie beim Erhöhen des Sollwerts?
 - Was passiert, wenn der eingestellte Wert höher als der Motornennstrom ist?
- Achtung! Bei grösseren Motoren kann dies aufgrund des hohen Drehmoments schwierig sein.

Sie sollten gelernt haben, dass der Stromregelkreis immer eine Begrenzung durch einen externen, höher liegenden Drehzahl- oder Positionsregelkreis oder durch einen mechanischen Anschlag benötigt, um eine "Durchdreh"-Situation zu vermeiden.



Mehr Hintergrund zu Regelkreisen

Beachten Sie die Informationen, im Zusammenhang mit der EPOS4-Steuerung

5 Spezielle ESCON Werkzeuge

5.1 Diagnose

Anmerkung: Sehen Sie sich auch das ESCON Video Tutorial in der Registerkarte Videos auf der Produktseite des ESCON 36/2 Controllers auf www.maxongroup.com an.

Zielsetzung: Informieren Sie sich über das *Diagnose*-Werkzeug des ESCON Studios.

- Schritt 1 Machen Sie absichtlich einen Verdrahtungsfehler oder eine falsche Einstellung der Sensorrichtung. Die einfachste ist eine *invertierte* Encoder-Drehrichtung.
- Schritt 2 Arbeiten Sie das *Diagnose*-Werkzeug des ESCON Studios durch.
- Schritt 3 Korrigieren Sie Ihren Verdrahtungsfehler oder die falsche Einstellung der Encoder-Drehrichtung.
- Schritt 4 Arbeiten Sie sich erneut durch das *Diagnose*-Werkzeug.

5.2 Virtueller Controller

Zielsetzung: Erfahren Sie, wie Sie den *virtuellen Controller* zur Vorbereitung ohne Hardware verwenden können.

Konfiguration des virtuellen Controllers

Starten Sie ohne angeschlossene Hardware (Motor, ESCON), arbeiten Sie einfach mit Ihrem Computer und dem ESCON Studio.

- Schritt 1 Öffnen Sie im ESCON Studio das Dialogfeld *Optionen* im Menü Datei. Wählen Sie *Virtueller Controller* und aktivieren Sie das Kontrollkästchen *Nach Virtuellen Controllern suchen*. Drücken Sie OK.

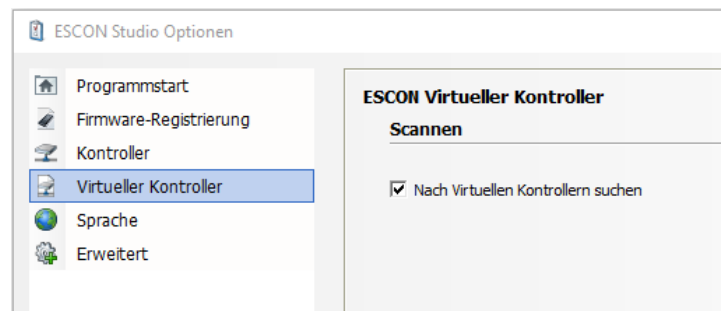


Abbildung 8: Nach Virtuellen Controllern suchen

- Schritt 2 *Kontroller suchen...* entweder im Menü *Datei* oder drücken Sie das Fernglas-Symbol. Wählen Sie aus dem Dropdown-Menü den ESCON-Typ aus, den Sie verwenden möchten. In unserem Fall ist es *ESCON 36/2 DC - IPC0*. Der Kommunikationstyp ist als Erweiterung des Controller-Namens gekennzeichnet. Erweiterungen virtueller Controller werden als IPCx bezeichnet.

- Schritt 3 Konfigurieren Sie den virtuellen ESCON Regler mit dem Startup-Assistenten zur Verwendung mit unserem DCX-Motor. Die Betriebsart ist Drehzahlregler mit Encoder-Feedback.
- Schritt 4 Konfigurieren Sie andere Parameter als zuvor. Z.B. ein grosser Drehzahl Sollwertbereich ($-5000 \dots +5000 \text{ min}^{-1}$) auf *Analogeingang 2* (statt 1), Freigabe & Drehrichtung auf *Digitaleingang 1* und 2, *Stopp* auf *Digitaleingang 3*.

Übertragen der Konfiguration auf den realen ESCON-Kontroller

- Schritt 5 Speichern Sie die virtuelle Konfiguration auf Ihren Computer. Verwenden Sie den Befehl *Parameter Hochladen* im Menü *Datei* oder das entsprechende Symbol mit dem blauen Upload-Pfeil.
- Schritt 6 Verbinden Sie Ihre echte ESCON über USB mit Ihrem Computer. Stellen Sie den aktiven Kontroller mit der Erweiterung USBx als den echten ESCON Kontroller Typ ein.
- Schritt 7 Laden Sie die gespeicherten Parameter auf die ESCON herunter. Verwenden Sie den entsprechenden Befehl im Menü *Datei* oder das Symbol mit dem orangefarbenen Download-Pfeil.
- Schritt 8 Bei korrekter Verdrahtung und nach einem Tuning sollten Sie in der Lage sein, den Motor entsprechend der in der virtuellen Steuerung vorgenommenen Konfiguration zu betreiben.

5.3 Standard-Parameter wiederherstellen

Bitte stellen Sie am Ende der Schulung die Werkseinstellungen in der ESCON wieder her, damit der nächste Kursteilnehmer wieder bei null anfangen kann.

Verwenden Sie entweder das Symbol für die Wiederherstellung der Parameter oder den entsprechenden Befehl im Menü *Datei*



Abbildung 9: Standard-Parameter wiederherstellen

Teil 3: EPOS4 Vorbereitung

Die nächsten beiden Kapitel dienen dem Kennenlernen des EPOS4-Systems und der Vorbereitung für die Übungen.

In Kapitel 6 geben wir einen Überblick über die Hardware, gehen auf die Installation der Software *EPOS Studio* ein und erläutern kurz die Hauptelemente des Motion-Control-Systems.

In Kapitel 7 wird das Motion-Control System für die folgenden Übungen vorbereitet. Die Eigenschaften des Motors und des Encoders müssen definiert und der Regelkreis abgestimmt werden.

6 Einrichten des EPOS4 Starter Kit

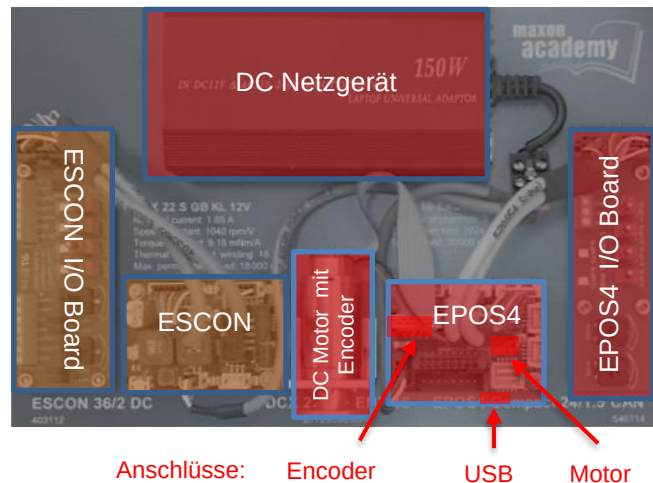


Abbildung 10: Die Komponenten des ESCON/EPOS4 Starter Kit
Die rot markierten Komponenten werden für das EPOS4 Training benötigt..

Im Starter Kit enthalten ist ein DC Motor mit einem Encoder, rechts davon eine *EPOS4 Compact 24/1.5 CAN* Positioniersteuerung mit einer angeschlossenen Leiterplatte mit Schaltern, LED-Lampen und Potentiometer. Diese Leiterplatte dient als Ein- und Ausgangs Interface (I/O Board) mit digitalen und analogen Eingängen und Ausgängen. Ausserdem ist ein DC Netzgerät für die Stromversorgung enthalten. Die benötigten Kabel befinden sich in der Box unter dem Kit.

Links befindet sich ein *ESCON* Servo-Controller mit einem eigenen Ein- und Ausgangs Interface. Der *ESCON* Controller wird im Teil 1 dieses Buches behandelt.

6.1 Installation der EPOS Studio Software

Zuerst gilt es den Computer vorzubereiten. Die Software für das EPOS Studio kann von der maxon Webseite epos.maxongroup.com heruntergeladen und installiert werden. Auf der Produktseite kann der Knopf *Download EPOS Setup* angeklickt werden.

Die Installation umfasst alle nötigen Informationen und Werkzeuge (wie zum Beispiel Handbücher, Firmware, Windows DLLs, Treiber), die für die Installation und den Betrieb des *EPOS4 Motion Controllers* nötig sind. Das wichtigste Werkzeug ist das *EPOS Studio*. Es ist die Bedienoberfläche für den vollen Zugriff auf alle Eigenschaften und Parameter des Motion Controllers.

Schritt 1: Folgen Sie den Anweisungen, die beim Ausführen des Installationsprogramms gegeben werden. Bitte lesen Sie jede Anleitung sorgfältig durch. Geben Sie den Ort des Arbeitsverzeichnisses an, wenn Sie dazu aufgefordert werden.



Arbeitsverzeichnis

Wir empfehlen den folgenden Ort als Arbeitsverzeichnis:

C:\Programme (x86)\maxon motor ag

Beachten Sie, dass die Bezeichnung des Programmverzeichnisses je nach der installierten Systemsprache variieren kann.

Schritt 2 Es werden neue Verknüpfungen und Symbole im Startmenü Ihres Computers angelegt. Die Dateien wurden in das Menü *maxon EPOS4* kopiert, wo Sie auf das Programm sowie auf die gesamte Dokumentation zugreifen können. Ein Klick auf die *ESCON Studio* Verknüpfung auf Ihrem Desktop startet das Programm.



Das neueste EPOS Studio

Nach der Installation, überprüfen Sie, dass Sie die Version *EPOS Studio 3.6* (Revision 2) oder höher verwenden.

Klicken Sie im Menüpunkt *Help* auf *About EPOS Studio*. Falls Sie nicht die neueste Ausgabe installiert haben, laden Sie diese von der maxon Website herunter: epos.maxongroup.com.

Komponente	Minimale Anforderungen
Betriebssystem	Windows 10, 8, 7
Prozessor	Core2Duo 1.5 GHz
Laufwerk	Harddisk oder SSD Laufwerk, 1.5 GB verfügbarer Speicher
USB Port	USB 2.0 / USB 3.0
Speicher	1 GB RAM
Bildschirm	Auflösung 1024 x 768 Pixels mit hoher Farbauflösung (16-Bit)
Web Browser	Microsoft Internet Explorer 8.0
Software	Microsoft .NET Framework 2.0 & 4.0

*Tabelle 1: Minimale Systemanforderungen
(Vgl. Release Notes.txt im EPOS Studio Installationsverzeichnis)*

6.2 Kabel anschliessen

Die Kabel anzuschliessen ist einfach und grundsätzlich fehlersicher. Schliessen Sie zunächst den DCX-Motor (2 Drähte: rot, schwarz mit einem kleinen schwarzen Stecker) an die einzige weisse Buchse an, die passt. Dann machen Sie dasselbe mit dem Encoder-Flachbandkabel. Die Platine mit den I/O's ist bereits angeschlossen. Dann - und es ist gute Ingenieurpraxis, dies zuletzt zu tun - schliessen Sie die Stromversorgung an. Zum Schluss stecken Sie das USB-Kabel in einen der USB-Ausgänge Ihres Computers und in den *EPOS Micro USB-Stecker*. Ihr Setup sollte dann ähnlich aussehen wie in Abbildung 11.

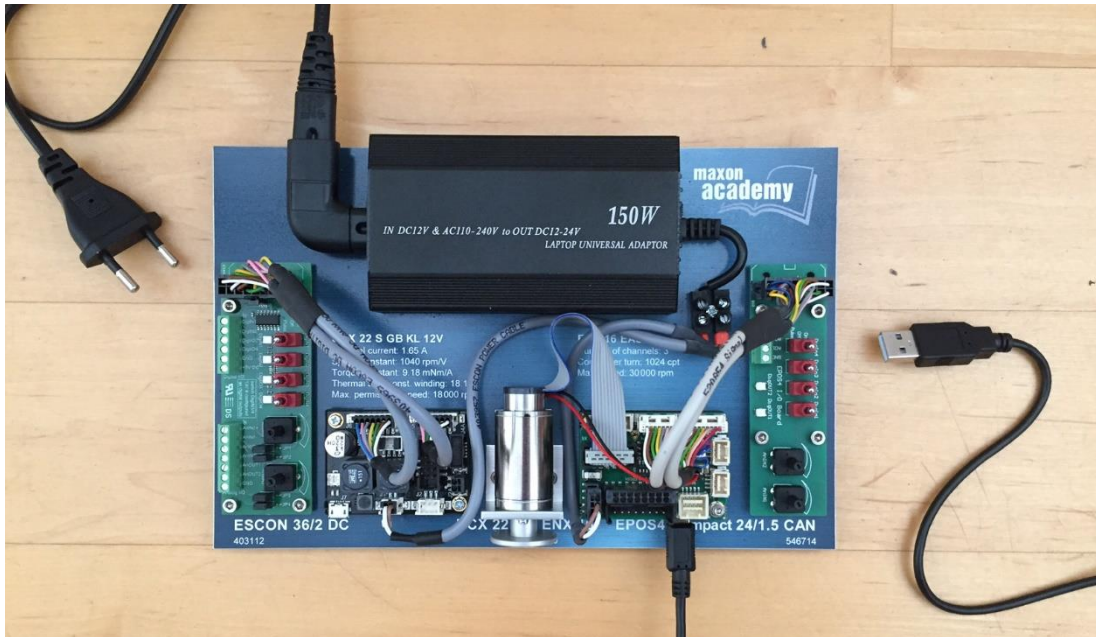


Abbildung 11: Anschliessen des EPOS4 Starter Kit

Best Practice

Den USB-Treiber finden

Beim ersten Anschliessen der EPOS4 am USB-Port des Computers erscheint die Meldung, dass ein neues Gerät gefunden wurde und verlangt nach dem entsprechenden Treiber. Dieser wird automatisch installiert und benötigt eine gewisse Zeit. Sie können die Installation jedoch beschleunigen, wenn Sie den Treiber manuell aus dem Installationsverzeichnis der EPOS4 Software installieren:

maxon motor ag\EPOS IDX\Driver Packages\EPOS USB Driver. Befolgen Sie die Schritte gemäss Dokument *EPOS USB Driver Installation.pdf*. Die Installation kann leicht abweichen, abhängig vom verwendeten Betriebssystem auf dem Computer.

6.3 Die EPOS4 als Black Box und deren Dokumentation

EPOS ist ein Akronym für "**E**asy to use **P**ositioning **S**ystem" und die 4 steht für das Design der vierten Generation. Es liegt jedoch an Ihnen zu beurteilen, ob es wirklich "EPOS" oder nur "POS" ist, d.h. ob es einfach zu benutzen ist oder nicht.

Betrachtet man die EPOS4 als Black Box, so erhält man ein grundlegendes Verständnis ihrer Funktionalität und ihrer Ein- und Ausgaben. Es handelt sich um einen Motion-Controller, der Position, Drehzahl oder Strom von bürstenbehafteten oder bürstenlosen DC-Motoren mit Encoder-Feedback regeln kann. Es gibt auch mehrere digitale und analoge Ein- und Ausgänge, die für verschiedene Zwecke im Zusammenhang mit der Bewegungsachse verwendet werden können.

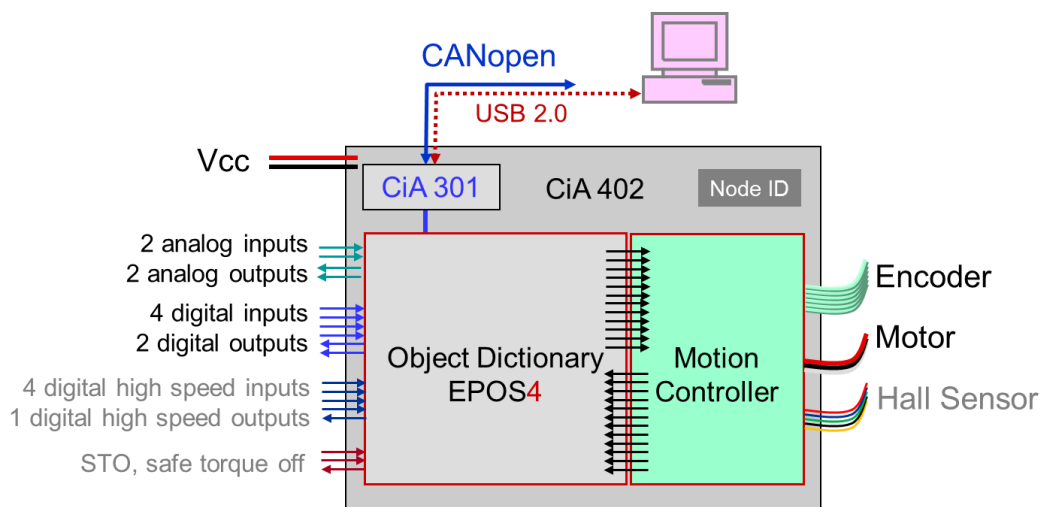


Abbildung 12: Schematische Übersicht der EPOS4 mit externen Anschlüssen

Aus der Sicht der Systemumgebung ist die *EPOS4* ein CANopen-Gerät für *Antriebe und Bewegungssteuerung* (CiA 402). Das *EPOS Studio* auf Ihrem Computer kommuniziert mit ihm über eine serielle USB 2.0/3.0 Verbindung. Alternative Wege für die Kommunikation sind serielle RS232- oder CANopen-Busverbindung. Höchstwahrscheinlich hat Ihr Computer keine CAN-Schnittstelle, so dass wir diesen Bus nicht verwenden können.

EPOS4-Bewegungssteuerungen sind CANopen-Slaves und benötigen einen Master mit einem Programm, das ihnen und anderen Slaves Befehle erteilt. Master können SPS (PLC), PC, Mikrocontroller und andere programmierbare Geräte mit einer geeigneten Kommunikation sein. Sie steuern den Prozessablauf und alle Slave Einheiten im Netzwerk. In unserer Situation wird es das EPOS-Studio sein, das die Rolle des Master-Programms übernimmt. Mit einer Schnittstelle ermöglichen die EPOS4-Steuerungen sogar die Adressierung durch EtherCAT von einem EtherCAT-Master-System.



Handbücher und Software Dokumentation

Die Installation richtet nicht nur die *EPOS Studio* Software auf Ihrem Computer ein, sondern kopiert auch alle Handbücher und Software, die Sie zur Programmierung des vorliegenden oder eines anderen maxon EPOS-Gerätes benötigen.

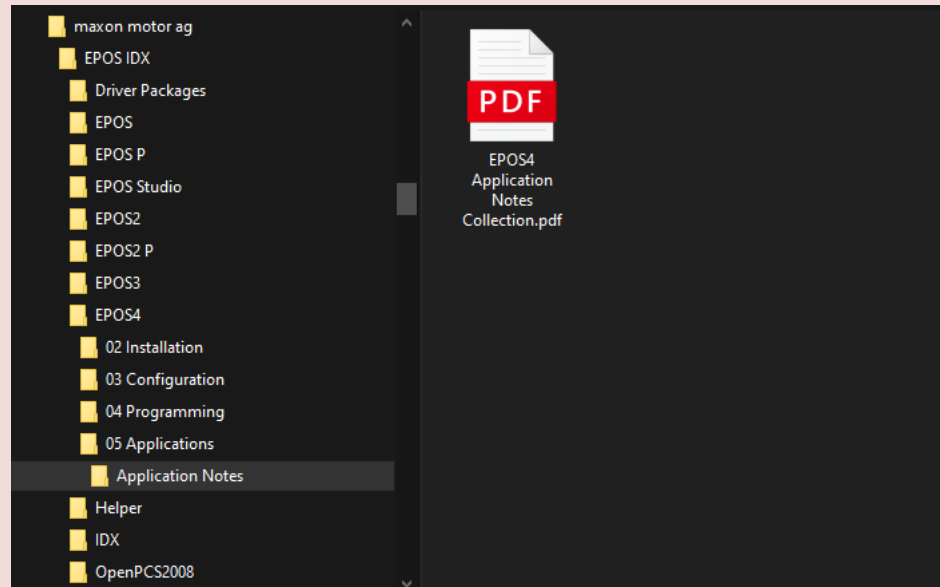


Abbildung 13: Die Ablagestruktur der maxon EPOS Software Installation

Für die EPOS4-Produkte gibt es 4 Unterordner mit den Dokumenten wie in *Abbildung 14*:

3. *02 Installation*: Informationen über die Hardware und Verkabelung.
4. *03 Configuration*: Aktuelle und ältere *Firmware Files*; dies ist die Software, welche auf dem EPOS Controller läuft.
5. *04 Programming*: Enthält wichtige Dokumente für Programmierer: Die *Firmware Specification* (siehe Kapitel 0) wo alle Eigenschaften und Parameter im Detail beschrieben sind. Ein Programmierer findet hier weitere nützliche Informationen wie Programmierbeispiele und Bibliotheken im Dokument *Programming Reference*.
6. *05 Application*: Die *Application Notes Collection* enthält Informationen zu besonderen Anwendungsbedingungen und Betriebsmodi.

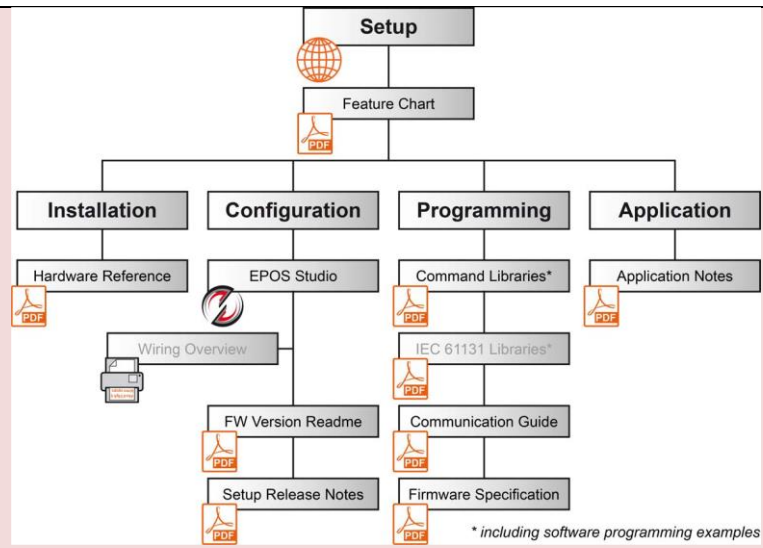


Abbildung 14: Die Dokumentationsstruktur der maxon EPOS4

6.4 Das EPOS Studio starten

Wir sind nun bereit! Starten Sie das *EPOS Studio* durch Anklicken des entsprechenden Symbols auf Ihrem Computerdesktop.

Das *EPOS Studio* ist die grafische Benutzeroberfläche für alle maxon EPOS-Produkte. Das Schöne am *EPOS Studio* ist, dass es die direkte Kommunikation mit jedem Gerät im Netzwerk ohne Programmieraufwand ermöglicht. Dies ist nützlich für das Einrichten des Systems, für Schulungen und für alle anderen Untersuchungen, die Sie durchführen möchten. Seien Sie sich jedoch bewusst, dass Sie in einer realen Anwendung das Mastersystem programmieren müssen, um die richtigen Befehle an Ihre EPOS zu senden.

Um die Kommunikation mit der eigentlichen Hardware herzustellen, wählen Sie im Assistenten für neue Projekte das richtige EPOS-Projekt, welches sich automatisch öffnen sollte. Falls nicht, klicken Sie auf das Symbol *New Project* in der Menüleiste.

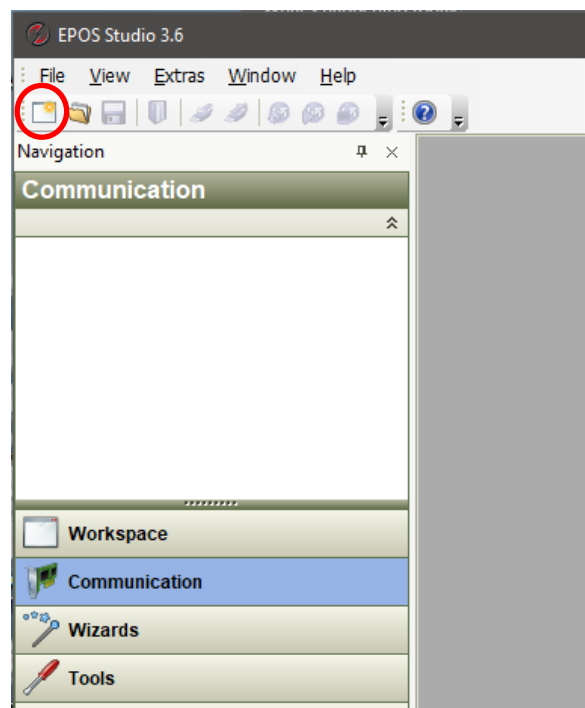


Abbildung 15: Wo Sie das Symbol für ein neues Projekt finden



Projekt im *EPOS Studio*

Das *EPOS Studio* ermöglicht es, die Einstellungen Ihres Projekts zu speichern und es wieder zu öffnen. Ein Projekt im *EPOS Studio* enthält Information über die Hardware und den benützten Kommunikationskanal. Falls mehrere EPOS in einem Netzwerk angeschlossen sind, so wird dies ebenfalls gespeichert. Projekte unter einem eigenen Namen zu speichern ist immer dann nützlich, wenn sie nicht ein Standardprojekt wie in diesem Fall hier verwenden.

Schritt 1 Wählen Sie *EPOS4 Project* von der Liste. Drücken Sie *Weiter*.

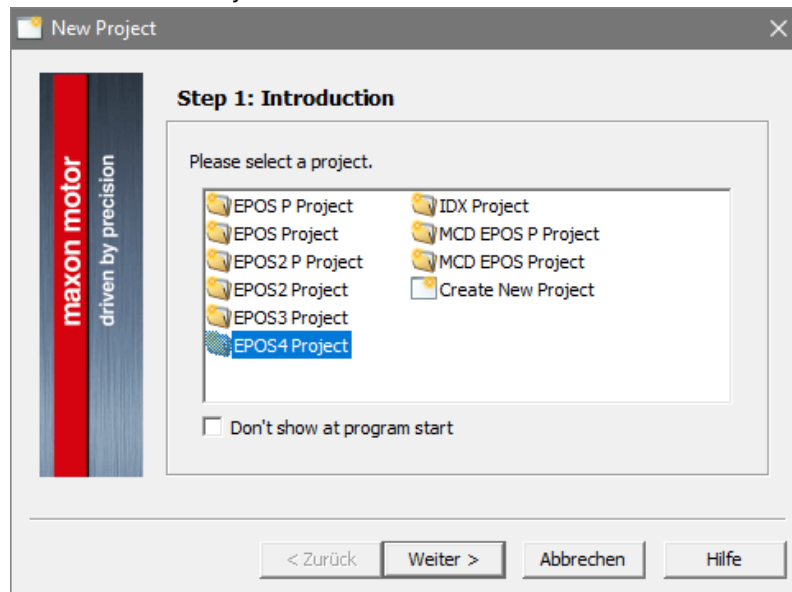


Abbildung 16: New Project Wizard, Schritt 1

Schritt 2 Nun definieren Sie einen Namen und einen Pfad für Ihr Projekt oder verwenden Sie den vorgeschlagenen Namen. Drücken sie *Fertigstellen*, um das Projekt zu erzeugen.

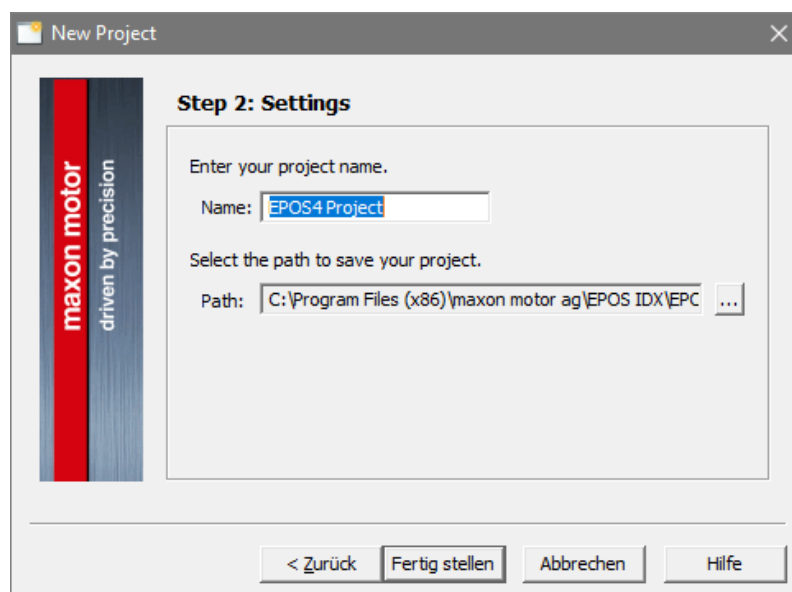


Abbildung 17: New Project Assistant, Schritt 2

Wenn alles geklappt hat, meldet das *EPOS Studio*, dass die Projektdatei gespeichert wurde - Sie müssen diese Meldung bestätigen - und stellt die Kommunikation mit der *EPOS4* her. Letzteres ist daran zu erkennen, dass die Einstellbalken in einem Pop-up-Fenster von links nach rechts verlaufen und anzeigen, dass die Parameter der *EPOS4* gelesen werden.

Das Workspace Register

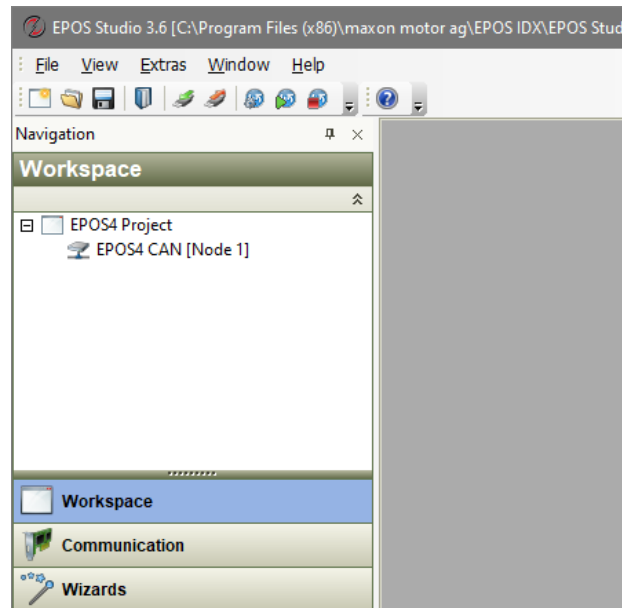


Abbildung 18: Das EPOS Studio mit geöffnetem Workspace Register

Das *Workspace Register* im *Navigation* Fenster links zeigt die Regler Hardware des Projekts. In unserem Fall enthält das Projekt einen *EPOS4 [Node 1]* Motion-Controller.



Fehler und Warnungen

Im *Statusfenster* unten werden möglicherweise Fehler und Warnungen angezeigt. Wenn z.B. keine *CAN-Kommunikation* aufgebaut wird, kann der *Can Passive Error* erscheinen; oder ein *Hallsensor-Error* in unserem Fall, wo ein bürstenbehafteter DC-Motor ohne Hallsensoren angeschlossen ist.

Im Moment können Sie die Fehler noch ignorieren und sie löschen, sobald Sie eine ordnungsgemäße Einrichtung vorgenommen haben. Die Fehler werden gelöscht, indem Sie mit der rechten Maustaste auf die Fehlerliste klicken und *Clear All Entries* wählen.

Falls andere Fehler oder Warnungen erscheinen, überprüfen Sie die Verdrahtung. Einzelheiten zu Fehlern und Warnungen finden Sie im separaten Dokument *EPOS4 Firmware Specification*.

Status				
Type	Node	Code	Name	Description
 Error	EPOS4 [Node 1]	0x7388	Hall sensor error	Motor hall sensor report

Abbildung 19: Fehler und Warnungen im Statusfenster

Das Communication Register

Das *Communication Register* zeigt den Aufbau Ihres Projekts aus der Sicht der Kommunikation. Schematisch können Sie sehen, dass Ihr Computer (*LocalHost*) über *USB* mit der *EPOS4 [Node 1]* verbunden ist. Es ist eine unbenutzte *CAN*-Verbindung vorhanden, wo andere CANopen-Knoten (z.B. andere EPOS4 Controller) hinzugefügt werden könnten.

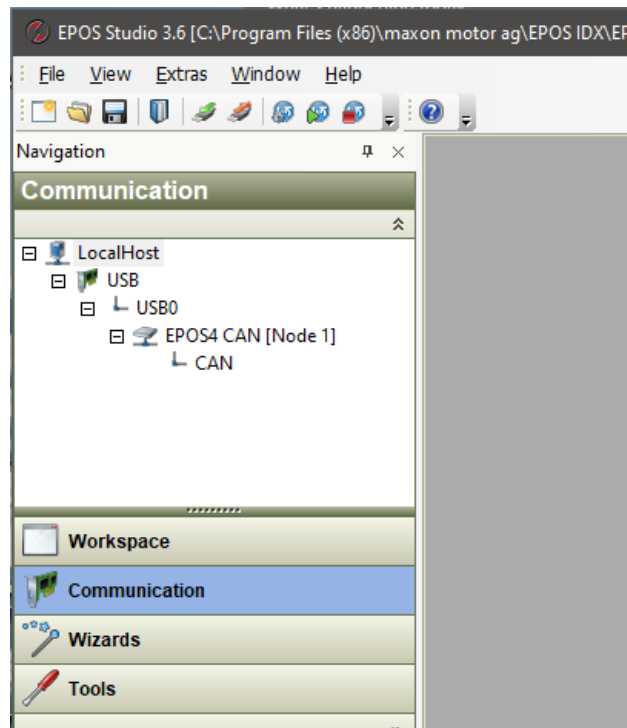


Abbildung 20: Das Communication Register im EPOS Studio

Das Wizards und Tools Register

Es besteht derzeit keine Notwendigkeit, die Assistenten und Werkzeuge im Detail zu betrachten. Viele von ihnen werden später auf unserer Reise benutzt und erklärt werden, wenn sie auftauchen.

Nur eine Bemerkung: Die Tools und Wizards sind den verschiedenen Hardwarekomponenten im Projekt zugeordnet. Wählen Sie daher immer das richtige Device aus dem Dropdown-Menü oben aus. Hier haben wir allerdings nur ein Device zur Hand: die *EPOS4 [Node 1]*.

6.5 Herunterladen der aktuellsten Firmware

Bevor wir starten, stellen Sie sicher, dass unsere EPOS4 die aktuelle Firmware geladen hat. Dazu starten wir den *Firmware Update Wizard* mittels Doppelklick.

Schritt 1 Bestätigen Sie, dass Sie die Warnmeldung gelesen haben und drücken Sie *Next*.

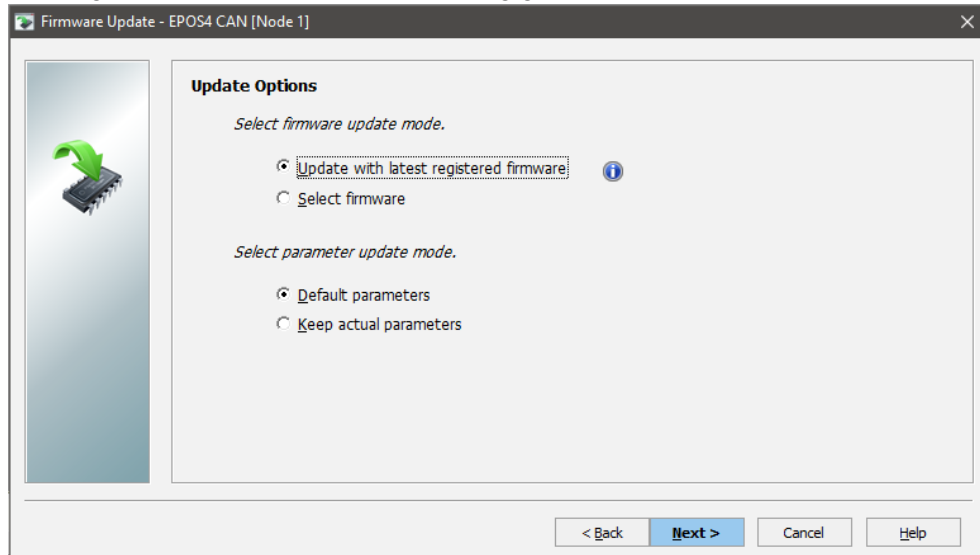


Abbildung 21: Firmware Update Wizard: Einstellung der Optionen

Schritt 2 Es besteht keine Notwendigkeit, die Standard-Aktualisierungsoptionen zu ändern, drücken Sie einfach erneut auf *Next*.

Schritt 3 Starten Sie das Firmware-Update und warten Sie, bis die Schaltfläche *Finish* aktiv wird. Unterbrechen Sie den Vorgang nicht! Drücken Sie dann *Finish*.

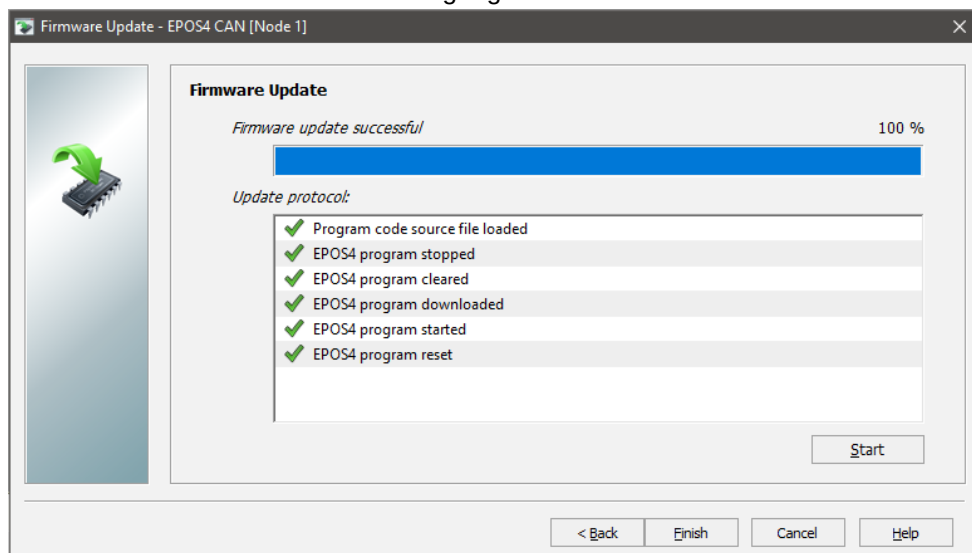


Abbildung 22: Firmware Update Assistent: Nach erfolgreichem Firmware Update

6.6 Zurücksetzen auf Standardparameter

Es ist immer ein guter Ratschlag, alle Standardparameter in der EPOS4 wiederherzustellen, bevor irgendwelche Experimente durchgeführt werden. Klicken Sie mit der rechten Maustaste auf den EPOS-Controller im Kommunikations-Gerätebaum.

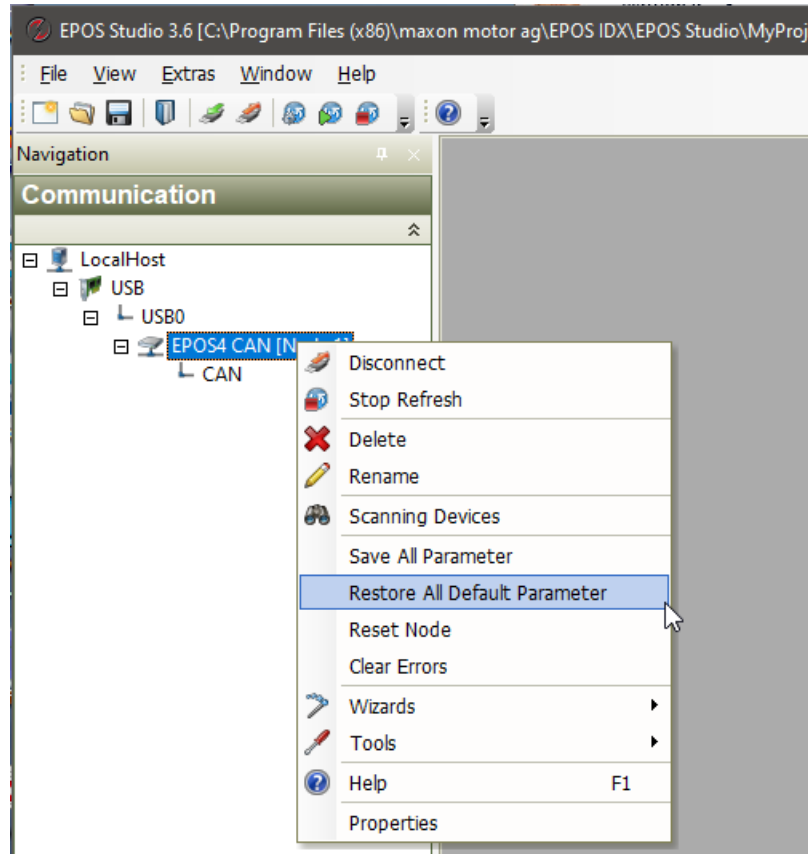


Abbildung 23: Restore All Default Parameters

7 Einrichten des EPOS4 Motion-Control Systems

7.1 Startup Wizard

Eine EPOS4 kann viele verschiedene Antriebskonfigurationen mit einer Vielzahl von Motoren und möglichen Sensoren und - wie wir bereits gelernt haben - mit mehreren Kommunikationsmöglichkeiten steuern. Daher muss als Erstes der eigentliche Aufbau definiert werden. Zu diesem Zweck führen wir den *Startup Wizard* aus, der auf der Registerkarte *Wizards* im *EPOS4 Studio* zu finden ist.

Doppelklicken Sie auf das entsprechende Symbol.

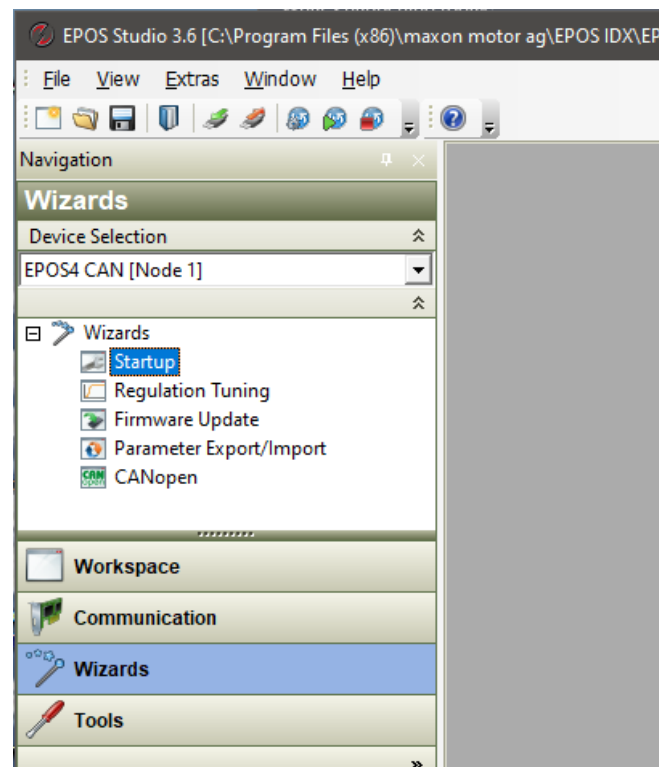


Abbildung 24: Doppelklick auf *Startup* im *Wizards* Register des *EPOS Studio*

Die linke Spalte im Wizard-Dialog zeigt die verschiedenen Schritte der Systemdefinition, beginnend mit den *Sicherheitshinweisen*. maxon möchte sicherstellen, dass Sie eine qualifizierte Person sind. Wenn Sie diesem Lehrbuch folgen, werden Sie eine solche werden. Haben Sie also keine Angst und kreuzen Sie unten das Kästchen "Yes, I have read the above instructions" an, sonst können Sie nicht fortfahren. Bei der Konfiguration folgen wir der vorgegebenen Reihenfolge - von oben nach unten in der linken Spalte. Nachdem Sie jeden Schritt abgeschlossen haben, klicken Sie auf die Schaltfläche *Next* am unteren Rand

Glücklicherweise haben wir ein recht einfaches Setup und nur einige wenige Parameter weichen von den Standardeinstellungen ab. Es gibt nicht allzu viele Optionen zur Auswahl. Wenn Sie ein komplexeres System haben (z.B. Dual-Loop-Konfiguration mit Feedback-Sensoren am Motor und an der Last), arbeiten Sie den *Startup Wizard* sorgfältig durch.

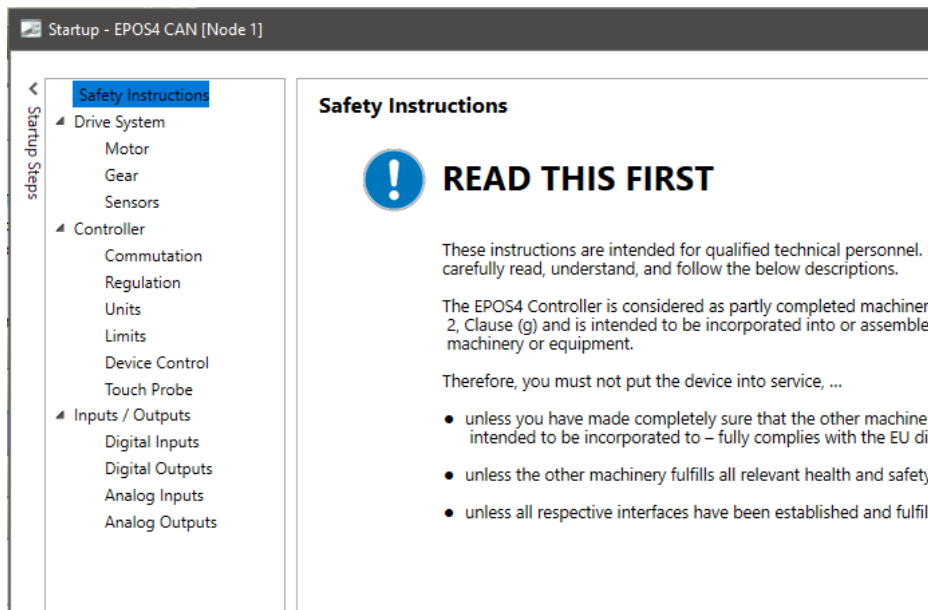


Abbildung 25: Startup Wizard, Sicherheitshinweise

- Schritt 1 Der erste Schritt ist die Definition des verwendeten Motortyps. Die EPOS Motion Controller können mit bürstenbehafteten DC-Motoren (maxon DC Motor) oder mit bürstenlosen DC-Motoren (maxon EC-Motor) eingesetzt werden. Da es sich bei dem vorliegenden DCX-Motor um einen maxon DC Motor handelt, wählen Sie die erste Option. Entsprechend passen sich die folgenden Schritte automatisch an. Als nächstes müssen Sie einige Schlüsselparameter unseres DCX 22 S GB KL 12V Motors angeben, die auf der Hauptplatine des Starter Kits aufgedruckt sind. Da der Dauerstrom des Motors (1650 mA) grösser ist als der Dauerstrom der EPOS4 (1500 mA), beschränken wir uns auf den letzteren, d.h. niedrigeren Wert. Durch Deaktivierung des entsprechenden Kontrollkästchens können Sie sogar die Drehmomentkonstante des Motors eingeben.

Motor

Enter motor type and characteristics (consult maxon catalog).

Motor type	maxon DC motor
Nominal current	1500 mA
Torque constant	☐ Identify during mechanical system identification
	9.180 mNm/A
Thermal time constant winding	18.1 s
Max speed	18000 rpm

Abbildung 26: Startup Wizard, Schritt 1, Motorparameter

- Schritt 2 Getriebe (Gear). Wir können Schritt 2 auslassen, da kein Getriebe am Motor montiert ist.

- Schritt 3 In Schritt 3 definieren wir die Art des Sensors, der verwendet wird, um den Positions- oder Drehzahlregelkreis zu schliessen. Das Standardgerät für maxon EPOS-Steuerungen ist ein inkrementeller digitaler Encoder. Diesen haben wir an unserem Motor und er heisst ENX 16 EASY. Seine Parameter sind wiederum auf der Hauptplatine zu finden: Es handelt sich um einen 3-Kanal-Encoder mit 1024 cpt (Counts per turn oder Impulse pro Umdrehung). Klicken Sie auf *Next*, um fortzufahren.

Sensors

Enter sensor data for connected sensors.

Connector	Type	Mounting Position
X4 - Hall sensor	None	-
X5 - Encoder	Digital incremental encoder 1	On motor shaft
X6 - Sensor	None	-

X5 - Digital incremental encoder 1

Number of pulses: 1024 pulses/revolution

Type: Encoder with index (3-channel)

Direction: ☒ maxon ☐ Inverted

Channel A:  90°
 Channel B: 
 CW CCW

Abbildung 27: Startup Wizard, Schritt 3, Sensor Parameter

- Schritt 4 *Regulation*. In unserem Fall gibt es nicht viele Regulierungsoptionen zur Auswahl, wie Sie sehen können, wenn Sie die Dropdown-Buttons untersuchen. Im Wesentlichen könnten wir eine Drehzahlregelung auf *Observer*-Basis wählen, was zu einem besseren Verhalten der Drehzahlregelung bei extrem niedriger Drehzahl und niedriger Sensorauflösung führen könnte. Unsere Encoder-Auflösung ist jedoch recht gut. Daher sind die Standardeinstellungen perfekt. Lassen Sie die Einstellungen so, wie sie sind, und klicken Sie einfach auf *Next*, um fortzufahren.
- Schritt 5 *Units*. Es besteht keine Notwendigkeit, die Einheiten (Units) zu ändern.
- Schritt 6 *Limits*. Hier können Sie allgemeine Betriebsgrenzwerte definieren, die jederzeit eingehalten werden sollen. Wenn ein Kommando eine solche Grenze überschreitet, tritt ein Fehler auf.
- Die Strombegrenzung wurde bereits auf 1500 mA festgelegt, dem maximal möglichen Wert für diese EPOS4-Steuerung. Wir werden den maximalen Ausgangsstrom bei 4500 mA halten, um eine schnelle Beschleunigung zu ermöglichen.
- Der Standardwert der Max. Beschleunigung ist extrem hoch. Wir belassen ihn jedoch so, um unseren Betrieb nicht unnötig einzuschränken.
- Da die maximal zulässige Drehzahl des Motors $18'000 \text{ min}^{-1}$ beträgt, kann die maximale *profile velocity* nicht schneller gewählt werden.
- Bemerkung: Die maximale Drehzahl des Encoders ($30'000 \text{ min}^{-1}$) ist viel höher

und wird nicht der begrenzende Faktor sein. Im Falle eines Getriebes wird die maximale Drehzahl oft durch die Eingangsdrehzahl des Getriebes begrenzt. Wir belassen den das *Following error window* (Schleppfehlerfenster-Fenster) auf seinem Standardwert und begrenzen den möglichen Positionsbereich (*position limit*) nicht - wie es z.B. bei einer Linearachse sinnvoll wäre. Klicken Sie auf *Next*, um fortzufahren.

Limits

Enter the drive system operating limits (consider the component with the lowest limit value).

Max continuous current	1500 mA
Max output current	4500 mA
Max acceleration	4294967295 rpm/s
Max profile velocity	18000 rpm
Following error window	2000 inc
☐ Use software position limit.	
Min position limit	0 inc
Max position limit	0 inc

Abbildung 28: Startup Wizard, Schritt 6, Limitierungen

- Schritt 7 *Device Control*. In diesem Konfigurationsschritt legen Sie fest, wie das System auf spezielle Vorfälle wie Herunterfahren, Deaktivieren, Schnellstopp und andere reagieren soll. Sie können auch einige Parameter einstellen, die diese Reaktionen regeln. Wir wollen uns jetzt nicht mit diesen Einstellungen beschäftigen und behalten die Standardwerte bei. Weitere Informationen finden Sie im Dokument *Firmware Specification*.
- Schritt 8 *Digital Inputs*. Konfiguration der digitalen Eingänge. Auch hier behalten wir die Konfiguration so, wie sie ist.
- Schritt 9 Dasselbe tun wir für die Konfiguration der *digitalen Outputs* (Ausgänge). Es muss nur noch *Finish* gedrückt werden, um die Setup-Parameter in der EPOS4 zu speichern.

Das ist es, was wir mit einfach zu benutzen meinen. Mit wenigen Klicks und Parametern ist das System eingerichtet.



Motor- und Sensor-Parameter der vorliegenden Einheit

Der verwendete Motor ist ein *DCX 22 S*, 12 V Nennspannung, mit Graphitbürsten und Kugellager. Seine genaue Spezifikation ist auf der maxon [Website](#) oder im [Anhang](#) zu finden. Wichtig für uns ist die Tatsache, dass es sich um einen bürstenbehafteten DC-Motor handelt.

Dieser Motor hat eine maximale Drehzahl von $18'000 \text{ min}^{-1}$ und liegt damit unter dem, was wir mit der gegebenen Spannung der Stromversorgung (24 VDC) erreichen können. Typischerweise ist der Motor in der Lage, bei 24 V Versorgung Drehzahlen von bis zu ca. $22'500 \text{ min}^{-1}$ zu erreichen (Drehzahlkonstante des Motors mal Motorspannung = $1040 \text{ min}^{-1}/\text{V} * 0.9 * 24\text{V}$).

Der Nennstrom beträgt 1,65 A. Der Motor kann diesen Strom kontinuierlich ohne Überhitzung ziehen. Für einige Sekunden kann er einen viel höheren Strom aufnehmen, möglicherweise bis zu 10 A. Wir sind jedoch nicht in der Lage, so viel Strom aus unserer EPOS4 herauszuholen, die eine Stromgrenze von 4,5 A aufweist.

Der Sensortyp ist ein *ENX 16 EASY*-Encoder mit 1024 Impulsen pro Umdrehung auf jedem der beiden Kanäle. Daraus ergeben sich 4096 Signalfanken (Zustände) pro Umdrehung der Motorwelle, die als Inkremente (inc) bezeichnet werden. Die inc sind die internen Positionseinheiten, die von den EPOS-Steuerungen verwendet werden. Daher haben wir eine Nennauflösung von $1/4096$ einer Umdrehung oder $0,088^\circ$.

Darüber hinaus verfügt der vorliegende Encoder über einen Indexkanal, der für eine präzise Referenzfahrt verwendet werden kann.



Kommutierung mit bürstenbehafteten und bürstenlosen Motoren

Der Begriff Kommutierung beschreibt die Art und Weise, wie die Versorgungsspannung an die verschiedenen Wicklungssegmente angelegt wird.

Bei bürstenbehafteten Motoren – z.B. maxon DCX Motor – erfolgt die *Kommutierung* im Motor mittels Graphit- oder Edelmetallbürsten. Um den Motor in Drehung zu versetzen, muss der Regler lediglich eine Spannung anlegen.

Bei bürstenbehafteten Gleichstrommotoren ist für eine korrekte Kommutierung kein zusätzliches Feedback erforderlich. Für die Messung von Drehzahl und Position in einem Regelkreis ist jedoch ein Encoder oder eine andere Feedbackvorrichtung erforderlich.

Bei bürstenlosen DC Motoren (BLDC) – z.B. maxon EC Motoren - gibt es drei Wicklungssegmente (Phasen). Die Kommutierung, d.h. die richtige Spannungsversorgung dieser drei Phasen entsprechend der tatsächlichen Rotorlage, ist eine zusätzliche Aufgabe des Reglers.

Die einfachste Art der Kommutierung für BLDC-Motoren, genannt *Blockkommutierung* wird durch die Rotorpositionsinformation erreicht, die von 3 digitalen Hallsensoren geliefert wird. Kurz und ohne ins Detail zu gehen, die Hallsensoren sind notwendig, um den BLDC-Motor korrekt anzusteuern. Hallsensoren liefern jedoch nur eine Rückmeldung mit niedriger Auflösung, und die Blockkommutierung hat ihre Grenzen, wenn es um den reibungslosen Betrieb bei niedrigen Drehzahlen geht.

Eine ausgeklügeltere Art der Kommutierung wird als *Sinuskommutierung* oder *Field Oriented Control (FOC)* bezeichnet. Diese führt zu einem ruhigen Motorbetrieb auch bei sehr niedrigen Drehzahlen und ergibt einen etwas höheren Motorwirkungsgrad. Sie erfordert jedoch ein zusätzliches, hochauflösendes Feedback eines Encoders. Glücklicherweise ist dies bei Servosystemen mit Positions- oder Drehzahlregelung in der Regel ohnehin vorhanden.

Hinweis: Bei bürstenlosen Gleichstrommotoren wird das Feedback für zwei separate Aufgaben benötigt: für die Kommutierung - d.h. für den Motorbetrieb - und für die Drehzahl- oder Positionsregelung.

Die EPOS4-Motion Controller erlauben Blockkommutierung für Motordrehzahlen bis zu $100'000 \text{ min}^{-1}$, während die Sinuskommutierung eine maximale Drehzahl von $50'000 \text{ min}^{-1}$ aufweist. Diese Werte gelten für BLDC-Motoren mit 1 Magnetpolpaar. Höhere Polpaarzahlen reduzieren die maximale Drehzahl entsprechend. Ein EC-4-pole Motor mit 2 Polpaaren kann nur mit der Hälfte dieser Drehzahlwerte betrieben werden. Flachmotoren haben meistens noch mehr Polpaare und - entsprechend - eine geringere maximale Drehzahl (siehe Tabelle 2).



Positions- und Drehzahlauswertung mit Inkr. Encoder

Inkrementelle Digital-Encoder sind der gebräuchlichste Sensortyp zur Messung von Drehzahl und Position in Kleinstantrieben. Sie sind in verschiedenen Bauformen und mit unterschiedlichen Funktionsprinzipien zu finden, aber alle unterteilen grundsätzlich eine Umdrehung in eine Anzahl von Schritten (Inkremente) und senden jedes Mal, wenn ein Inkrement erkannt wird, Signalimpulse an die Steuerung. Inkrementelle Encoder liefern in der Regel Rechtecksignale in zwei Kanälen, A und B, die um ein Viertel der Signallänge (oder 90 elektrische Grad) versetzt sind.

Der **charakteristische Parameter** ist die Anzahl der Rechteck-Impulse N pro Umdrehung, angegeben in Imp/U (Impulszahl pro Umdrehung). Zählt man die Anzahl der Zustandsübergänge in beiden Kanälen, ist die erreichbare Auflösung viermal höher. Diese Zustände werden als Inkremente (inc) oder Quad Counts (qc) bezeichnet und in EPOS-Systemen als Positionseinheit verwendet.

Hinweis: Wenn über die Auflösung von inkrementellen Encodern gesprochen wird, ist darauf zu achten, ob die Zählungen vor oder nach der Quadratur gemeint sind.

Je höher die Auflösung, desto genauer kann die Position gemessen werden, und desto genauer kann die Drehzahl aus der Positionsänderung als Funktion der Zeit abgeleitet werden.

Die **Drehrichtung** ergibt sich aus der Abfolge der Signalimpulse von Kanal A und B: Kanal A führt in eine Richtung, Kanal B führt in die andere Richtung.

Zusätzlich kann der Encoder über einen **Index Kanal** verfügen. Dies ist ein schmaler Impuls pro Umdrehung. Der Index kann verwendet werden, um eine sehr genaue Positionsreferenz zu erhalten (siehe Homing, Kapitel 3.2). Einige Regler für bürstenlose Motoren benötigen den Index auch als absolute Rotorreferenz bei sinusförmiger Kommutierung.

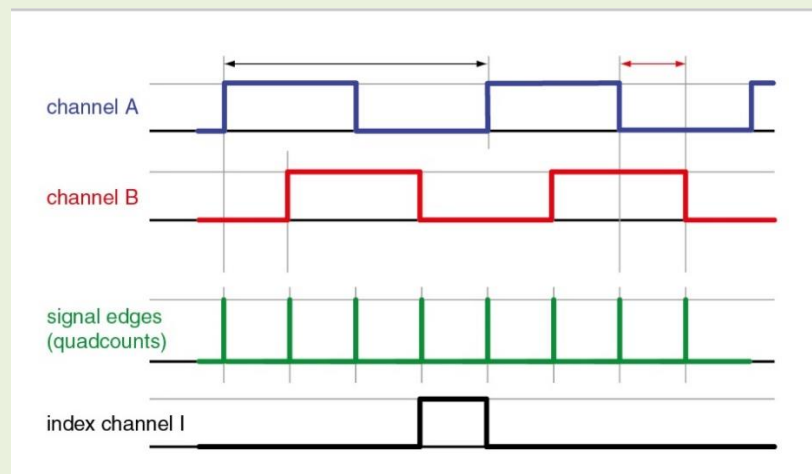


Abbildung 29: Die Signale eines Inkremental-Encoders

Die **Positionsauswertung** wird durch Zählen der Signalfanken (Inkmente) ab einer definierten Referenzposition (Home-Position, siehe Referenzfahrt in Kapitel 8.2) ermittelt. Die Einheit der Positionierung wird inc genannt.

Die **Drehzahlauswertung** wird aus der Positionsänderung pro Abtastintervall berechnet. In unserem Fall ist die natürliche Einheit der Drehzahl Inkmente pro Regelzykluszeit. Daher kann die Drehzahl nur in Schritten von 1 Inkrement pro 0,4 ms ausgedrückt werden, was üblicherweise in die gebräuchlichere und praktischere Einheit rpm (min^{-1}) übersetzt wird. Die Auswirkung der Drehzahl-Quantisierung des Inkrementalgebers lässt sich gut an den Diagrammen zur Drehzahlmessung erkennen (siehe Kapitel 0).

Beschleunigungs- und **Verzögerungswerte** werden in der praktischen und leicht zu interpretierenden Einheit rpm/s (min^{-1}/s) angegeben.



Encoder-Impulszahl und Regeldynamik

Eine hohe Encoder-Impulszahl N führt nicht nur zu einer hohen Positionsauflösung, sondern verbessert auch die Regeldynamik. Die Rückmeldung über eine Änderung des Positions-Istwerts erfolgt schneller und der Regelkreis kann umso schneller korrigierend eingreifen. Ausserdem kann der Regler steifer, d. h. mit höherer Regelverstärkung aufgebaut werden, was die Gefahr des instabilen Schwingens verringert.

Bei niedriger Impulszahl und hoher Einstellung der Regelparameter hat der Antrieb mehr Zeit zum Beschleunigen, bis ein neues Positionsrückmeldesignal, d.h. ein neuer Encoder-Impuls, empfangen wird. In einem solchen Fall kann der Antrieb die Endlage überschwingen und einen proportionalen Korrekturwert in die entgegengesetzte Richtung erzeugen. Dies kann wiederum zu einem Überschwingen in die neue Richtung führen. Die Bewegung wird instabil und der Antrieb schwingt.

Die Positionierung allein mit den Signalen der Hallsensoren führt typischerweise zu diesem instabilen Verhalten. Es wird dringend empfohlen, einen Encoder mit ausreichend hoher Auflösung zu verwenden.



EPOS4 maximale Drehzahl bei bürstenlosen Motoren

Die Mikrocontrollerleistung der EPOS4 begrenzt die maximal mögliche Drehzahl von bürstenlosen Motoren. Die Sinuskommutierung (FOC) benötigt mehr Rechenleistung als die Blockkommutierung, was zu einer geringeren möglichen Kommutierungsdrehzahl führt. Die erreichbare Motordrehzahl hängt auch von der Anzahl der Polpaare des Motors ab.

maxon EC Motoren (Beispiele)	Anzahl Polpaare	Max. Drehzahl mit Blockkommutierung	Max. Drehzahl mit Sinuskommutierung
maxon EC motor, EC-max	1	100'000 min ⁻¹	50'000 min ⁻¹
EC-4pole	2	50'000 min ⁻¹	25'000 min ⁻¹
EC 20 flat, EC 32 flat	4	25'000 min ⁻¹	12'500 min ⁻¹
EC-i 40, EC 60 flat	7	14'280 min ⁻¹	7'140 min ⁻¹
EC-i 52, EC 45 flat	8	12'500 min ⁻¹	6'250 min ⁻¹
EC 90 flat*	11 (neu) 12 (alt)	9'090 min ⁻¹ 8'330 min ⁻¹	4'540 min ⁻¹ 4'160 min ⁻¹

Tabelle 2: Maximale Drehzahl von EPOS4-Systemen für bürstenlose DC-Motoren in Abhängigkeit von der Anzahl der Polpaare und der Art der Kommutierung. Beachten Sie, dass einige der Motoren eine niedrigere maximale Drehzahlangabe haben, als bei der Blockkommutierung möglich ist. (Prüfen Sie die Grenzdrehzahl auf dem Motordatenblatt)*

7.2 Tuning

Der nächste Schritt zur Systemvorbereitung ist das Tuning. Der Zweck ist derselbe wie beim Tuning eines Sportwagens: Wir möchten eine schnelle, starke und optimierte Reaktion des Antriebssystems auf beliebige Bewegungsbefehle. Auch hier können wir einen Assistenten verwenden, um uns die Arbeit zu erleichtern. Starten Sie den Assistenten für das *Regulation Tuning* (Doppelklick) auf der Registerkarte *Wizards*.

Auf der linken Seite sehen Sie die 3 möglichen Tuning Schritte: Current (Strom), Velocity (Drehzahl), Position. Die einfachste Art, ein Tuning mit EPOS4 durchzuführen, ist die Verwendung der Autotuning-Funktion. Genau das werden wir hier tun.

Tuning des elektrischen Systems

Das Tuning des elektrischen Systems sollte zuerst gemacht werden.

Klicken Sie auf die Schaltfläche "Auto tune", um die nächste Meldung mit "Yes" zu bestätigen. Mit den angezeigten Parametern wird ein automatisches Tuning durchgeführt, welches das Verhalten des Motorstromregelkreises optimiert. Ziel ist es, dass der Strom möglichst schnell und präzise anliegt, wenn er von der übergeordneten Steuerungsebene, wie z.B. der Positions- oder Drehzahlregelung, gefordert wird. Ein gut abgestimmter Stromregelkreis ist die Voraussetzung für eine dynamisch Systemreaktion.

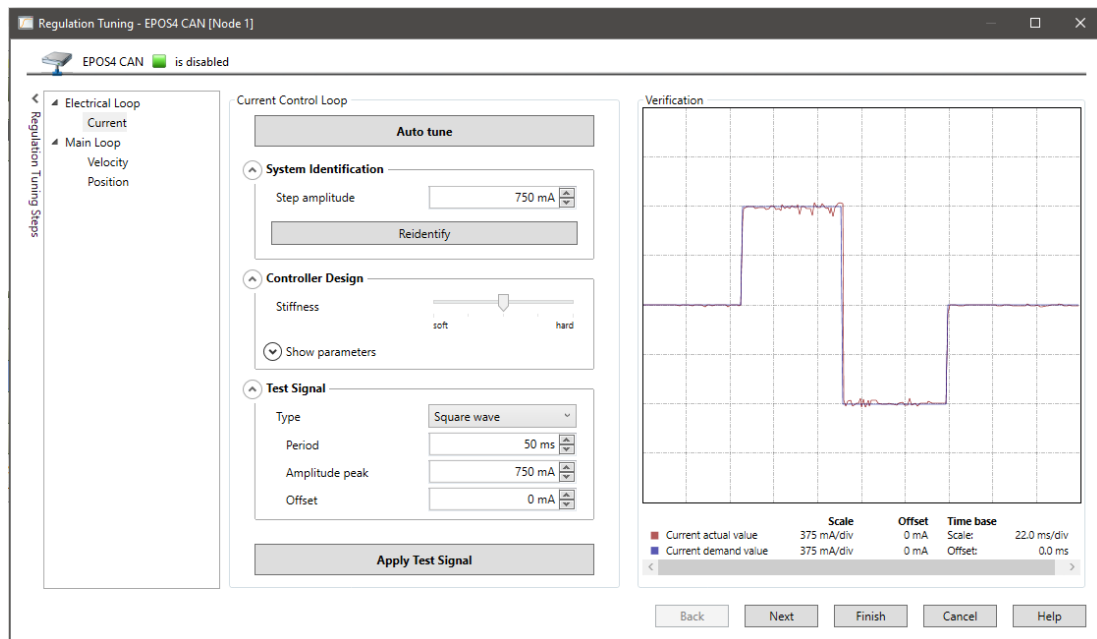


Abbildung 30: Regulation Tuning Assistent: Current (Strom)
Das Resultat des Auto-Tuning für den Strom. Das Verifikationsdiagramm zeigt die perfekte Reaktion auf einen aktuellen Sollwertsprung.

Die *Auto-Tuning* Funktion beginnt mit einem Schritt zur *Systemidentifikation*. Ein schwingendes Signal wird an den Motor angelegt. Es dient dazu, Informationen über das dynamische Verhalten des Systems zu erhalten. Die Systemidentifikation muss nur einmal durchgeführt werden, es sei denn, es gibt eine Änderung im *Antriebslayout*. In einem

nächsten Tuning-Schritt werden die optimalen Regler-Entwurfsparameter entsprechend der erforderlichen Steifigkeit berechnet. Der Regler-Entwurf basiert auf den Ergebnissen der Identifikation und verwendet ein allgemeines Modell der Auslegung des Antriebssystems. Zum Schluss wird beim Regler-Tuning ein *Testsignal* angelegt, d.h. es wird eine Testbewegung ausgeführt und die tatsächliche Systemreaktion aufgezeichnet. Das Diagramm zeigt das Ergebnis an.

Wenn Sie an der Bedeutung aller Parameter interessiert sind und daran, wie sie zur Verbesserung der Systemreaktion eingesetzt werden können, lesen Sie bitte unten in der EPOS4-Box *Tuning Parameter und manuelles Tuning* nach.

Drücken Sie *Next* oder wählen Sie *Velocity* im Menü auf der linken Seite.



Best Practice

Diagramm Zoom

Sie können in die Diagramme hineinzoomen, indem Sie den interessanten Bereich mit der Maus auswählen. Ein Doppelklick zoomt heraus.

Tuning des mechanischen Systems

Für das einfache Tuning der Drehzahl- und Positionsregelung verwenden Sie wieder die *Auto tune* Funktionalität. Das Verfahren folgt den gleichen Prinzipien wie oben beschrieben. Die Systemidentifikation dauert jedoch länger. Sie gewinnt Informationen über das mechanische Verhalten des Systems, indem sie den Motor mit steigender Amplitude und bei verschiedenen Frequenzen schwingen lässt. Wie zuvor wird das Ergebnis im *Verification* Diagramm rechts dargestellt.

Die *Auto tune* Prozesse liefern mit nur einem Mausklick recht gute Tuning-Parameter für die meisten Anwendungen. Umständliches Ausprobieren oder zeitaufwändiges Auswerten der aufgezeichneten Bewegungsantwort entfällt.

Hinweis: Das Tuning sollte immer am vollständig eingerichteten System mit der originalen Spannungsversorgung und allen Mechaniken, Trägheiten und Reibungen durchgeführt werden. Schliesslich soll ja die gesamte Maschine optimal reagieren und nicht nur der Motor allein.

Wenn Sie beim Verlassen des Assistenten für die Regelungsabstimmung auf *Finish* klicken, werden die neu gefundenen Abstimmungsparameter dauerhaft gespeichert. Die Parameter gelten für jeden zukünftigen Fahrbefehl. Sie werden dauerhaft gespeichert und sind auch nach einem Neustart des Systems funktionsfähig.

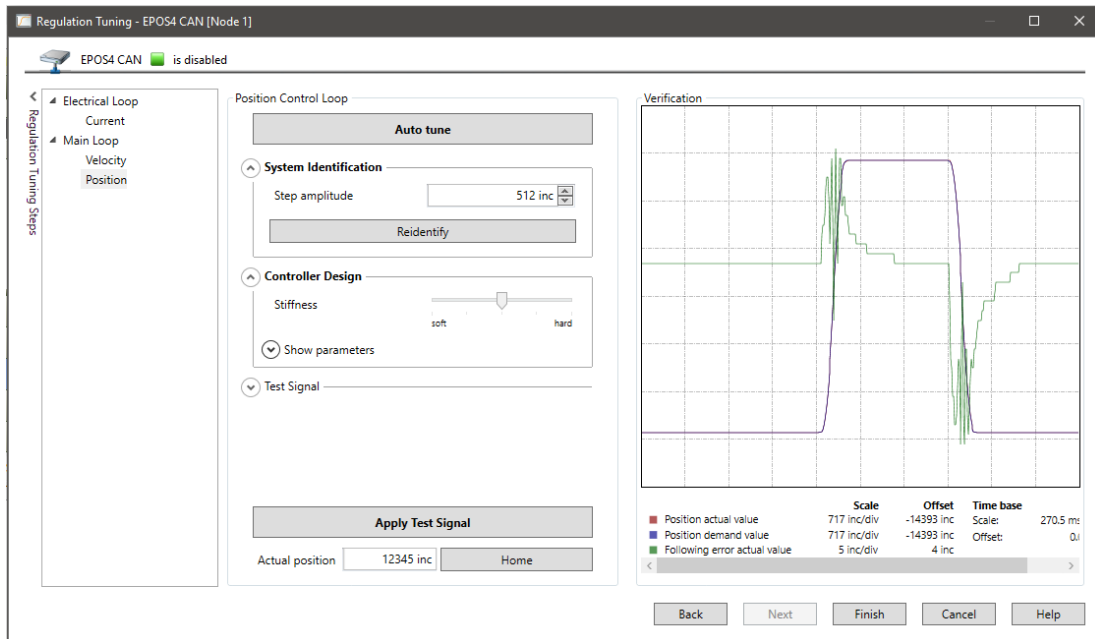


Abbildung 31: Regulation Tuning Assistant: Position
Das Ergebnis des automatischen Tuning der Position. Das Verification Diagramm zeigt die Reaktion auf eine Positionsbewegung vor und zurück. Das grüne Signal ist der verstärkte Positionsfehler, d. h. die Abweichung zwischen Istwert und Sollwert.



Tuning Parameter und manuelles Tuning

Der *Regulation Tuning* Assistent erlaubt die Einflussnahme in den Tuning Prozess. Wir zeigen hier einige der Funktionen bezüglich der Positionssteuerung.

Für die *Systemidentifikation* können wir festlegen, wie gross die Bewegungsamplitude des Motors sein soll. Standardwert ist ein Achtel einer Motorumdrehung oder 45° . In Kombination mit Getrieben und anderen mechanischen Anordnungen können höhere Amplituden notwendig sein, um eine wesentliche Bewegung aller mechanischen Teile zu erzeugen. Für die richtige Abstimmung müssen wir alle relevanten Massenträgheiten und Reibungen identifizieren, was nur möglich ist, wenn die gesamte Mechanik in Bewegung gesetzt wird. Sobald die Identifikation durchgeführt wurde, sind das Systemverhalten und die Auswirkungen von Reibung und Massenträgheiten bekannt. Daher ist es nicht notwendig, die Identifikation zu wiederholen, wenn Sie nur die Parametereinstellung im Schritt "*Controller Design*" ändern.

Die Art und Weise, wie die tatsächlichen Tuning-Parameter im *Controller Design* berechnet werden, hängt vom gewünschten Systemverhalten und Ihrer Anwendung ab. Vielleicht möchten Sie ein sehr steifes und schnell reagierendes System und es ist Ihnen egal, ob die Zielposition oder die Drehzahl überschwingt. Oder Sie stellen eine weiche Regelsteifigkeit mit langsamerer Reaktion und weniger Überschwingen ein.

Probieren Sie verschiedene Steifigkeitseinstellungen aus, vergleichen Sie die Parameterwerte, drücken Sie *Apply test signal* und beobachten Sie das Ergebnis in den Verifikationsdiagrammen. Um eine bessere Auflösung in den Diagrammen zu erreichen, stellen Sie die Einschwingzeit ein. In unserem Fall sind 200ms oder sogar weniger perfekt (siehe Abbildung unten).

Im Bereich *Test signal* können Parameter für eine typische Testbewegung eingestellt werden. In der Regel sollten die Parameter der Testbewegung den realen Bewegungen in den Anwendungen, typischerweise der kritischsten, ähnlich sein.

Das Aktivieren des Kontrollkästchens *Dimension manually* ermöglicht den Schreibzugriff auf die Regelparameter, d.h. die manuelle Abstimmung. Die manuelle Abstimmung wird normalerweise verwendet, um die Ergebnisse der automatischen Abstimmung zu optimieren. Dabei wird einer der Parameter geändert und das Ergebnis nach Anlegen des Testsignals grafisch überprüft.

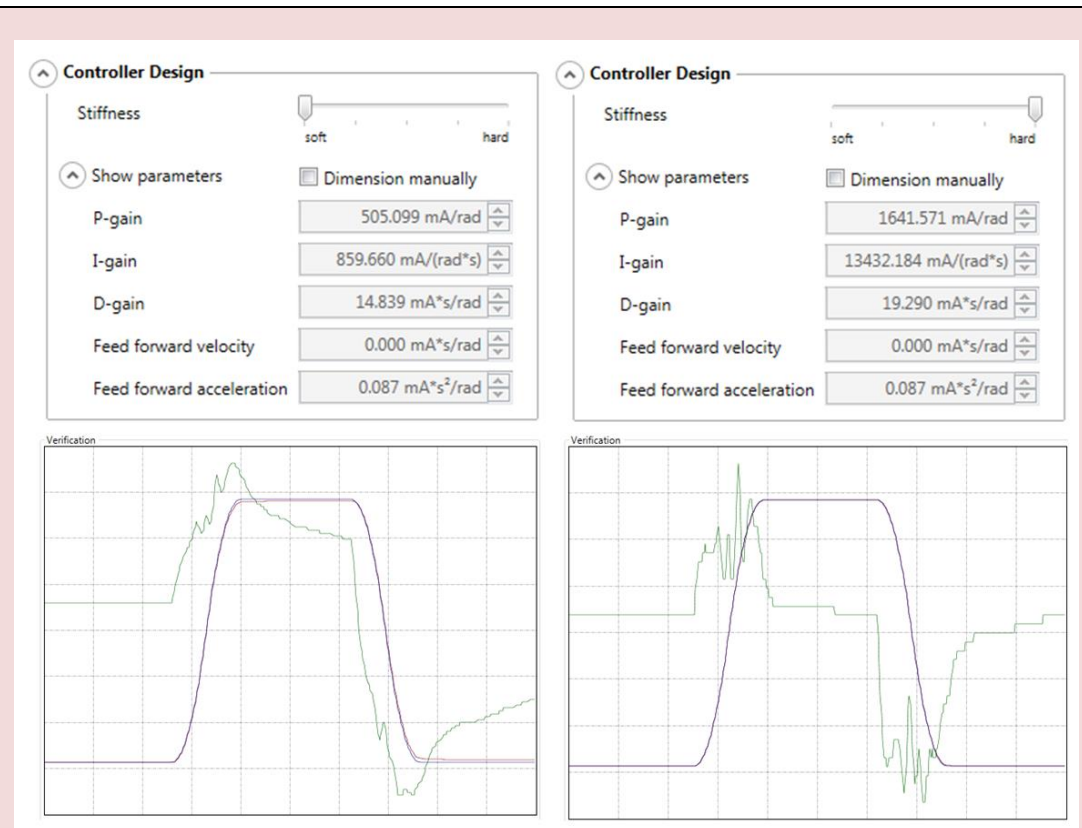


Abbildung 32: Regulation Tuning Assistant.
 Vergleich von weicher (links) und harter (rechts) Regelsteifigkeit: PID-Parameter (oben) und Systemantwort (Diagramm unten), die sich aus unterschiedlichen Einstellungen der Regelsteifigkeit ergeben. Die grüne Linie gibt den Positionierfehler an. Die weiche Regelung auf der linken Seite ist nicht in der Lage, die Endposition für beide Bewegungen innerhalb der kurzen Einschwingzeit von 200ms zu erreichen. Mit den steiferen Regelungsparametern auf der rechten Seite kann die Endposition fast erreicht werden. Beachten Sie die grössere Skala des Fehlersignals im linken Diagramm; jeder Schritt in den Signalen entspricht einem Inkrement.



Feedback und Feed Forward

PID Feedback Verstärkung

PID steht für *Proportional*-, *Integral*- und *Differenzial*- Regelparameter. Sie beschreiben, wie das Fehlersignal e (siehe Abbildung 33) verstärkt wird, um eine entsprechende Korrektur zu erzeugen. Ziel ist es, diese Regelabweichung, d.h. die Abweichung zwischen dem Soll- und dem Istwert, zu verringern. Kleine Werte der Regelparameter führen zu einem trägen Regelverhalten. Hohe Werte führen zu einer steiferen Regelung mit der Gefahr des Überschwingens und bei einer zu hohen Verstärkung kann das System zu schwingen beginnen. Dies lässt sich an unserem System leicht beobachten, indem man die Auto Tuning Parameter manuell um einen Faktor von etwa 5 erhöht.

Bei Drehzahlreglern wird üblicherweise ein geschlossener Regelkreis mit einem einfachen PI-Algorithmus implementiert. Bei Positioniersystemen ist zusätzlich ein Ableitungsterm notwendig. Die drei Terme können sich gegenseitig beeinflussen und das Verständnis dieser Wechselwirkung ist für die Feinabstimmung eines Positioniersystems von besonderer Bedeutung. Für eine optimale Systemleistung müssen die Koeffizienten K_P , K_I und K_D in Abhängigkeit von der vorgegebenen Bewegung und Lastträgheit eingestellt werden (Literatur: Feinmess).

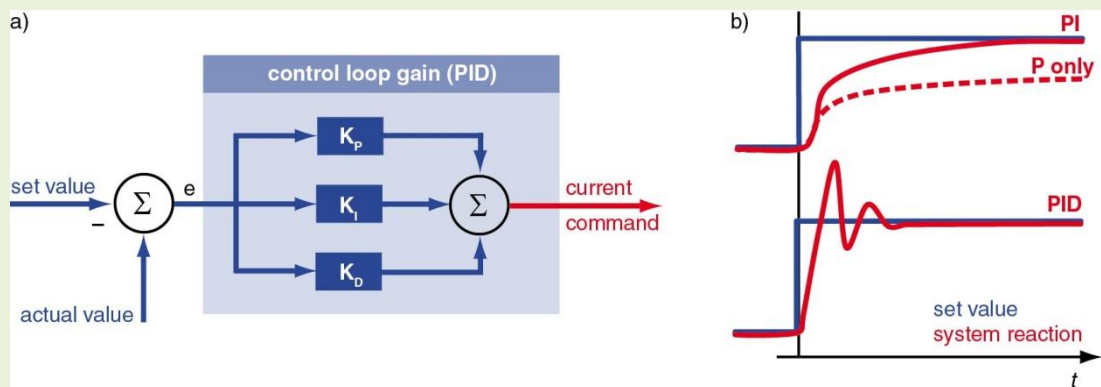


Abbildung 33: PID Regler

a) Schematische Darstellung eines Positions- oder Drehzahlreglers mit dem PID-Glied als Verstärker für das Fehlersignal e .

b) Systemreaktion auf eine Sollwertänderung mit einem P-, einem PI- und einem PID-Glied realisiert.

Proportional Regler (P): Das Fehlersignal e (Differenz zwischen Ist- und Sollwert) wird mit einem benutzerdefinierten Faktor K_P multipliziert und dann als neuer Stromsollwert übertragen. Ein erhöhter K_P -Wert beschleunigt die Fehlerkorrektur. Ist K_P jedoch zu gross, kommt es zu starken Überschwingern. Bei noch höheren Werten von K_P kann das System zu schwingen beginnen, was wiederum zu Instabilität führen kann, wenn die Dämpfung im System nicht ausreichend ist. K_P kann die Regelabweichung e nicht vollständig eliminieren, da der proportionale Korrekturwert ($K_P \cdot e$) mit abnehmender Regelabweichung e kleiner wird. Daraus ergibt sich ein Restfehler, der besonders bei Systemen wichtig ist, die einen hohen Strom benötigen, um die Bewegung aufrechtzuerhalten (z.B. wegen hoher Reibung).

Integral Regler (I): Hier wird der Fehler über eine bestimmte Zeit aufsummiert, mit einem benutzerdefinierten Faktor K_I multipliziert und dann zum neuen Stromsollwert addiert. Dadurch wird der Restfehler der reinen P-Verstärkung eliminiert, da eine länger andauernde Abweichung eine zunehmende Korrektur provoziert.

Diese Methode hat jedoch einen inhärenten Nachteil, insbesondere in Fällen, in denen der Fehler zwischen positiven und negativen Werten oszilliert. Da sich vergangene Fehler verzögert auswirken, kann dies zu einer positiven Rückkopplung und Destabilisierung des gesamten Regelkreises führen. Ohne entsprechende Dämpfung können hohe K_I -Werte zu starken Systemschwingungen führen.

Differenzial Regler (D): Der D-Regler berücksichtigt die Änderung der Regelabweichung, die mit einem benutzerdefinierten Faktor K_D multipliziert und zum aktuellen Befehl addiert wird. Plötzliche Fehler, wie z.B. Sollwertsprünge, können mit dieser Methode sehr schnell korrigiert werden.

Richtig eingestellt, kann diese Art der Regelung die Regelstabilität verbessern. Sie kann als eine Art elektronische Dämpfung angesehen werden. Eine Erhöhung des K_D -Wertes führt zu einer größeren Systemstabilität, aber ein zu hoher K_D -Wert führt zu Schwingungen.

Feed-Forward

Bei den PID-Algorithmen wird nur dann korrigierend eingegriffen, wenn es eine Abweichung zwischen Soll- und Istwert gibt. Für Positioniersysteme bedeutet dies, dass es während der Bewegung immer einen Positionsfehler gibt - eigentlich geben muss. Diese wird als Schleppfehler bezeichnet. Ziel der Vorsteuerung (Feed-Forward) ist es, diesen *Schleppfehler* zu minimieren, indem die Sollwertänderungen im Voraus berücksichtigt werden. Zur Kompensation der Reibung und der Massenträgheitsbeschleunigung wird in einem Steuerregler Energie zur Verfügung gestellt.

Generell stehen bei Feed-Forward zwei Parameter zur Verfügung. Sie müssen für die jeweilige Anwendung und Bewegungsaufgabe ermittelt werden:

Speed Feed-forward gain: Diese Komponente wird mit der geforderten Drehzahl multipliziert und kompensiert die drehzahlproportionale Reibung.

Acceleration Feed-forward correction: Diese Komponente bezieht sich auf die Massenträgheit des Systems und liefert ausreichend Strom, um diese Trägheit zu beschleunigen.

Durch Einbezug des *Feed-forward* wird der durchschnittliche Schleppfehler beim Beschleunigen und Abbremsen reduziert. Durch die Kombination von *Feed-forward* und PID-Regler muss der PID-Regler nur noch den nach dem *Feed-forward* verbleibenden Restfehler korrigieren, wodurch das Systemverhalten verbessert wird und ein sehr steifes Regelverhalten möglich ist.

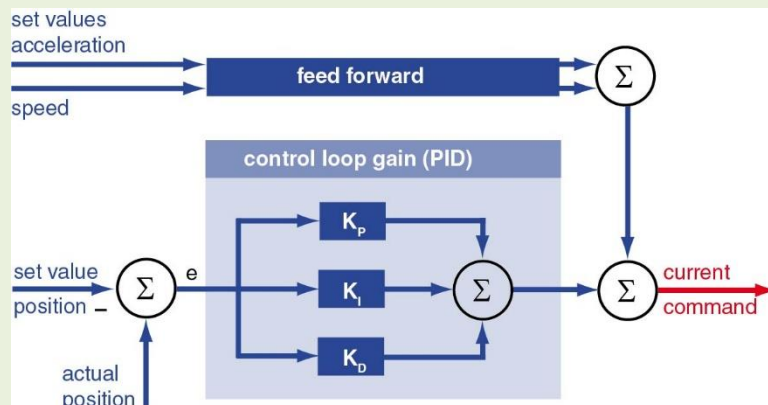


Abbildung 34: Schematische Darstellung des Feed-forward
Die Feed-forward Werte werden aus den gewünschten Sollwertänderungen, d. h. Beschleunigung und Drehzahl, berechnet. Beide Werte werden zu dem PID-Stromsollwert addiert.

Teil 4: EPOS4 Positionierungsregler

In Teil 3 haben wir das System konfiguriert, den Positionsregler abgestimmt und alle Einstellungen auf dem Gerät gespeichert. Der EPOS4-Motion-Controller ist nun bereit für die weitere Erkundung.

Kapitel 14 beleuchtet den EPOS4-Motion-Controller mit den Werkzeugen zur Untersuchung der verschiedenen Betriebsarten.

Kapitel 15 behandelt die mit dem Motion Controller verbundenen Ein- und Ausgänge.

In unserem Setup übernimmt das *EPOS Studio* die Rolle des Masters. Vom Studio aus senden wir einzelne Bewegungsbefehle (und andere) und können direkt am Motor und im Studio verfolgen, wie sie ausgeführt werden. Für den Moment ist es nicht nötig, irgendwelche Programmierungen vorzunehmen. Spielen Sie einfach!



Master, Slaves und online Einzelbefehle

Die *EPOS4*-Motion-Controller werden als Slaves in einem Netzwerk eingesetzt. Slaves erhalten ihre Befehle von einem übergeordneten System, genannt *Master*.

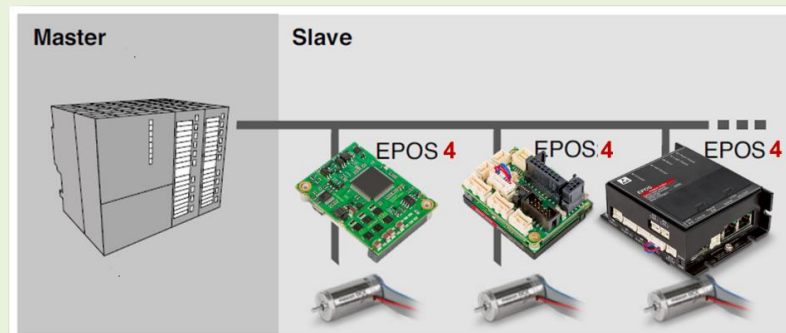


Abbildung 35: Master-Slave Architektur mit EPOS4 Slaves

Im Master läuft die Software, die alle Prozesse und Slaves im Netzwerk steuert. Ein Master kann eine SPS, ein Computer oder ein Mikroprozessor sein.

Die in diesem Teil 4 verwendete Master-Software ist das *EPOS Studio*.

Wichtig zu beachten ist, dass der Slave *EPOS4* keine Befehlssequenzen speichern kann; er wird online durch einzelne Befehle gesteuert, die vom Master gesendet und sofort nacheinander ausgeführt werden.

8 Erkundung des Motion Controllers

Die Werkzeuge zur Erkundung des Motion Controllers finden Sie in der Registerkarte *Tools* der *EPOS4*.

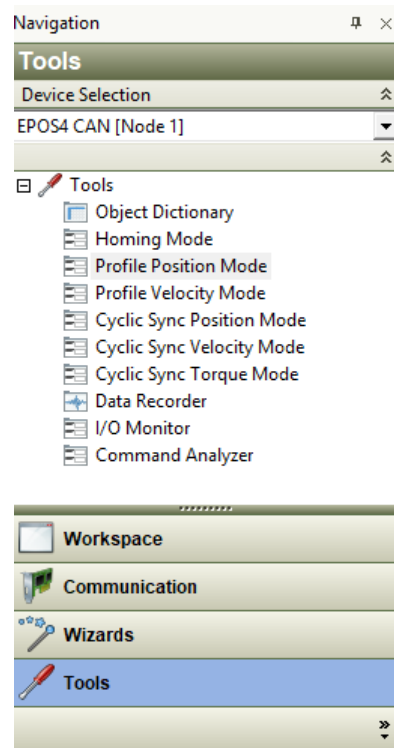


Abbildung 36: So öffnen Sie den *Profile Position Mode*
Wählen Sie die Registerkarte *Tools*, dann das Gerät *EPOS4 [Node 1]* und anschliessend einen Doppelklick auf *Profile Position Mode*.

8.1 Profile Position Modus

Zielsetzung: Ausführen einer Punkt-zu-Punkt-Bewegung und Überwachung der Bewegung durch den eingebauten Datenrekorder (*Data Recorder*).

Der Standard-Positionierungsmodus für die meisten Anwendungen ist der *Profile Position Mode*. Die Positionierung erfolgt, indem einem Drehzahlprofil gefolgt wird, das automatisch durch den eingebauten Pfad- oder Trajektoriengenerator erstellt wird. Der Pfadgenerator hat eine Taktrate von 0,4 ms, angepasst an die 2,5 kHz Abtastrate des Positionsregelkreises. Alle 0,4 Millisekunden wird ein neuer Positionsbedarfswert berechnet und als neuer Sollwert in den Positionsregelkreis eingespeist. Die generierte Trajektorie berücksichtigt Bewegungsparameter wie Beschleunigung, Drehzahl, Zielposition, Profiltyp und andere.

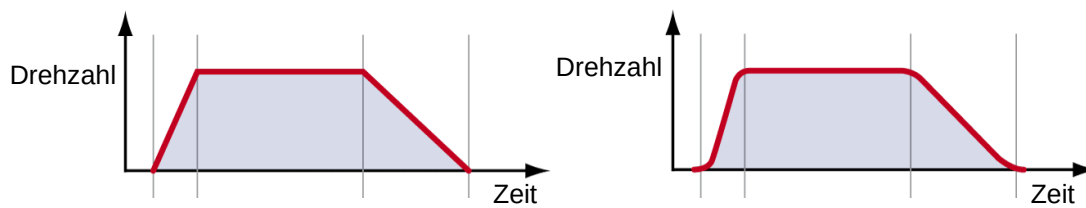


Abbildung 37: Drehzahlprofile (Drehzahl vs. Zeit) für eine Positionsfahrt
Links: Ein trapezförmiges Profil mit schneller Beschleunigung und langsamer Abbremsrate. Rechts (noch nicht verfügbar): Das gleiche Profil, aber mit sanfteren Beschleunigungsänderungen (Sinusprofil).

Punkt zu Punkt Bewegungen

Hier finden Sie eine erste Schritt-für-Schritt-Anleitung, wie Sie eine Bewegung befehlen.

- Schritt 1 Wählen Sie das Werkzeug *Profile Position Mode* und aktivieren Sie den Modus.
- Schritt 2 Setzen Sie die Zielposition *Target position* auf 20'000 inc
- Schritt 3 Setzen Sie die Profil-Drehzahl *Profile velocity* auf 500 rpm (min^{-1}).
- Schritt 4 Setzen Sie die Profilbeschleunigung *Profile acceleration* auf 10000 rpm/s.
- Schritt 5 Setzen Sie die Profilverzögerung *Profile deceleration* auf 5000 rpm/s.
- Schritt 6 Wählen Sie relatives Ziel *Relative target*.
- Schritt 7 Aktivieren (*Enable*) Sie die EPOS4.
- Schritt 8 Starten Sie die Bewegung, indem Sie auf die Schaltfläche *Move to target* klicken und beobachten Sie den Motor beim Drehen. Verfolgen Sie die Bewegung auch in den Parametern des Bereichs *Outputs* oben rechts im Dialogfenster. Wiederholen Sie den Vorgang nach Belieben.

Weitere Übungen

Versuchen Sie andere Bewegungen mit anderen Parametern (Drehzahl, Zielposition).

Sicherlich können Sie sich den Unterschied zwischen den Einstellungen *Absolute target* und *Relatives Ziel Relative target* vorstellen. Was ist die Endposition, wenn Sie auf *Move to zero* klicken?

Versuchen Sie, die Scheibe auf der Motorwelle mit der Hand aus der Zielposition herauszufahren. Was spüren Sie?

Beachten Sie auch die *Status* Anzeigen unten rechts im Dialogfenster.



Grüne und rote LED: Anzeige des internen EPOS-Zustands

Die grüne und rote LED an der EPOS4 zeigen den Zustand des internen EPOS4-Motion-Controllers an.

1. Grüne LED blinkt: Endstufe deaktiviert. Der Motor wird nicht mit Strom versorgt.
2. Grüne LED leuchtet konstant: Endstufe freigegeben. Der Motor wird gespeist und gesteuert.
3. Rote LED leuchtet: Ein Fehler ist aufgetreten.

Der Zustand dieser LEDs ist auch oberhalb der Dialogfenster im EPOS Studio zu sehen.



Die Regelkreise im Motion Control

Bei Motion Control ist die geregelte Grösse im Allgemeinen eine Kraft oder ein Drehmoment, eine Drehzahl (Geschwindigkeit) oder eine Position. Entsprechend wird der Regler als Strom-, Drehzahl- oder Positionsregler abgebildet. Oft kann zwischen verschiedenen Regelungsarten oder Betriebsarten umgeschaltet werden.

Regelung von Kraft/Drehmoment

Dies ist die grundlegende Aufgabe eines Antriebs in einem Regelsystem. Sie ist die Basis, auf der die übergeordneten Drehzahl- und Positionierregelungen aufbauen.

Kräfte und Momente werden benötigt, um Massen in Bewegung zu setzen, ihre Bewegung abzubremesen und zu stoppen sowie Reibungskräfte zu überwinden. Es kann auch Komponenten der Schwerkraft geben, gegen die gearbeitet werden muss, entweder um Lasten zu heben und zu halten oder um eine Abwärtsbewegung abzubremesen. In einigen Fällen der Drehmomentsteuerung geht es nicht darum, etwas zu bewegen, sondern einfach darum, mit einer definierten Kraft gegen ein Hindernis zu ziehen oder zu schieben. Das Motordrehmoment ist eine lineare Funktion des Motorstroms. Die Grundlage zur Erzeugung von geregelten Kräften und Drehmomenten ist also die Regelung und Steuerung des Motorstroms. Es ist der grundlegendste Regelkreis

Regelung der Geschwindigkeit/Drehzahl

Der Drehzahlregler versucht, die tatsächlich gemessene Drehzahl an eine geforderte Drehzahl anzupassen. Um einen Körper mit einer vorgegebenen Drehzahl zu drehen oder zu bewegen, muss der Körper zunächst beschleunigt werden. Anschliessend müssen die Kräfte und Momente, die auf die Drehzahl wirken, kompensiert werden. Um dieses Ziel zu erreichen, muss der Motor das notwendige Drehmoment erzeugen - er braucht also Strom. Der Drehzahlregler gibt also einen entsprechenden Befehl an den unterlagerten Stromregelkreis (ähnlich dem Positionsregler in *Abbildung 38*).

Positionsregelung

Um ein Objekt in eine vorgegebene Endposition zu bewegen, muss das Objekt zunächst in Bewegung gesetzt (beschleunigt), bewegt und anschliessend abgebremst werden. Zum Schluss muss die vorgegebene Position gegen alle Störkräfte gehalten werden. Alle diese Vorgänge erfordern ein Motordrehmoment, das dem Positionsregelungssystem zur Verfügung gestellt werden muss. Letztlich ist die Positionsregelung auf den unterlagerten Stromregelkreis angewiesen, um das notwendige Drehmoment zu erzeugen.

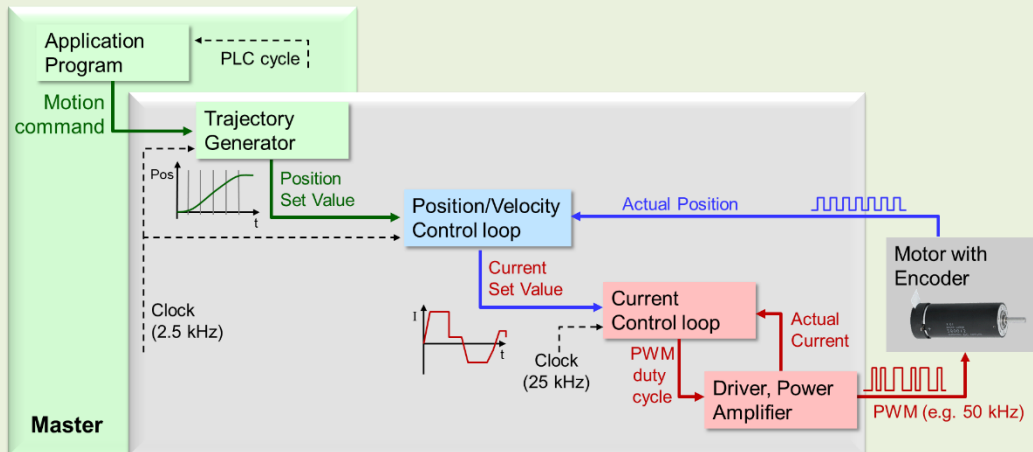


Abbildung 38: Schema eines Positionsregelungssystems im geschlossenen Regelkreis mit einem unterlagerten Stromregler. Das Applikationsprogramm im Master-System sendet Bewegungsbefehle an den Motion Controller. Die Befehle werden vom Pfad- bzw. Trajektoriengenerator verarbeitet, der in vorgegebenen Zeitintervallen (0,4 ms bei EPOS4-Systemen) Zwischenpositionen auf dem Pfad zum Erreichen der Endposition errechnet. Diese Sollwerte werden in den Positionsregelkreis eingespeist, der durch den Vergleich mit der Ist-Position die Soll-Werte für den Stromregler ermittelt. Der Positions- und Drehzahlregelkreis in der EPOS4 arbeitet mit einer Abtastfrequenz von 2,5 kHz; der Regelkreis erhält alle 0,4 Millisekunden neue Positionsinformationen. Dies ist für fast alle Anwendungen schnell genug, denn die mechanischen Reaktionszeiten liegen mindestens in der Grössenordnung von einigen Millisekunden, wie sich aus den mechanischen Zeitkonstanten der Motoren ableiten lässt. Der Stromregler steuert über die Endstufe bzw. den Treiber den Motorstrom, der zur mechanischen Reaktion des Antriebs führt. In der EPOS4 hat der Stromregelkreis eine Zykluszeit von 0,04 ms oder 25 kHz. Dies ist 10-mal schneller als der Positionsregelkreis, was bedeutet, dass der Strom bei Bedarf vom Positions- oder Drehzahlregler fast sofort eingestellt wird.

Konfiguration des Rekorders

Die Datenaufzeichnung ist ein nützliches Werkzeug für die Analyse des Bewegungsverhaltens. Sie arbeitet ähnlich wie ein Oszilloskop.

Wir verwenden den Rekorder, um die Bewegungsparameter im Detail zu verfolgen. Hier ist eine grundlegende Konfiguration Schritt für Schritt dargestellt.

- Schritt 1 Wählen und öffnen Sie das Werkzeug *Data Recorder*.
- Schritt 2 Wählen Sie im Bereich *Data Channels* für den obersten Kanal den Eintrag *Position demand value* aus dem Pulldown-Menü.
- Schritt 3 Wiederholen Sie den Vorgang mit dem zweiten Kanal und wählen Sie den *Position actual value* Positions-Istwert als anzuzeigenden Parameter.
- Schritt 4 Setzen Sie die Datenabtastzeit auf 2,0 ms.
- Schritt 5 Wählen Sie im Bereich *Triggers* die Option *Single Trigger* und stellen Sie den Type auf *Motion states*, den Trigger auf *Drive follows command* und die Verzögerung Delay auf 100 ms. Die Datenanzeige beginnt 100 ms vor der Ausführung des nächsten Bewegungsbefehls.
- Schritt 6 Drücken Sie die Schaltfläche *Start* um den Datenrekorder zu aktivieren.

Aufzeichnungen

Wechseln Sie auf die Registerkarte *Profile Position Mode* und aktivieren Sie den Modus, falls erforderlich. Führen Sie eine relative Bewegung aus, z.B. mit folgenden Parametern:

1. Zielposition 8192 inc (das sind 2 Motorumdrehungen)
2. Profil-Drehzahl 1000 rpm
3. Profilbeschleunigung 2000 rpm/s
4. Profilverzögerung 1000 rpm/s

Prüfen Sie die aufgezeichneten Daten im Register *Data Recording* und vergleichen Sie den Positionssollwert *Position demand value* mit der Ist-Position.

Weitere Übungen

Probieren Sie verschiedene Einstellungen des Datenrekorders und probieren Sie verschiedene Bewegungsprofilparameter aus.

Nutzen Sie die Hinweise im folgenden Kasten "Best Practice".

Best Practice

Datenrekorder

Der Datenrekorder ist ein leistungsfähiges Werkzeug zur Untersuchung aller Arten von Signalen: Bewegungsbezogene Parameter wie Drehzahl, Position und Strom, aber auch Encoder- oder Hallsensor-Feedback oder E/A- und Kommunikationssignale.

Trigger: Es gibt mehrere Möglichkeiten, die Aufzeichnung von Daten zu starten. Wählen Sie je nach Ihrem Signal die am besten geeignete aus. Die kontinuierliche Erfassung kann z.B. für Drehzahlmessungen verwendet werden, während die Einzel-Trigger-Erfassung, bei der der Trigger auf den Beginn der Bewegung gesetzt wird, besser zur Erfassung von Positionsbewegungen passt (siehe oben). Falls es Sie interessiert, wie die Endposition erreicht wird, können Sie als Trigger *Target reached* wählen.



Best Practice

Cursor: Aktivieren Sie den Cursor, indem Sie mit der rechten Maustaste auf das Diagramm klicken und *Cursor* wählen. Im unteren Bereich werden die Cursorwerte für jeden Kanal angezeigt. Verwenden Sie die Kontrollkästchen im Bereich *Data Channels*, um auszuwählen, welche Kurven angezeigt werden sollen.

Skalierung: Wenn Sie das Kontrollkästchen *Auto scale* deaktivieren, können Sie für jeden Kanal individuelle Skalen einstellen. Aktivieren Sie zunächst den entsprechenden Kanal im oberen rechten Feld. Der Vergleich von Signalen gleicher Art (z.B. *position demand* und *actual value*) ist einfacher, wenn sie im gleichen Massstab dargestellt werden. Die automatische Skalierung hat jedoch den grossen Vorteil, dass das Signal immer sichtbar ist und nicht aus dem definierten Anzeigebereich herausfällt. Erfassen Sie daher die Daten mit *Auto scale* und analysieren Sie sie anschliessend mit manueller Skalierung, Zoomen und der Cursor-Funktion.

Zoom in das Diagramm, indem Sie den interessanten Bereich mit der Maus markieren, z.B. in das Ende der Positionsfahrt, um die Genauigkeit der Positionierung und die Einschwingzeit zu überprüfen. Zoomen Sie mit einem doppelten Mausklick heraus.

Speichern und Exportieren der aufgezeichneten Daten in eine .csv-Datei, z.B. geeignet für den Import in Microsoft Excel! (Tipp: Rechte Maustaste im Diagramm verwenden).



Schleppfehlerfenster (Following error window)

Das *Following error window* gibt die maximal zulässige Differenz zwischen Soll- und Ist-Position zu einem beliebigen Zeitpunkt der Auswertung an. Es dient als Sicherheits- und Bewegungsüberwachungsfunktion.

Wenn die Schleppabweichung zu gross wird, ist das ein Zeichen dafür, dass etwas schief läuft: Entweder kann der Antrieb die geforderte Drehzahl nicht erreichen oder er ist sogar blockiert. Lassen Sie jedoch einen gewissen Schleppfehler zu, da die Regelung nur funktioniert, wenn eine Differenz zwischen Soll- und Ist-Wert besteht. Verringern Sie daher das Schleppfehlerfenster nicht zu sehr.



Positioniergenauigkeit

Die Endpositionen werden bei einem gut abgestimmten System (Tuning) mit einer Genauigkeit von 1 Encoder-Inkrement (inc) erreicht. Sobald die Position um 1 inc vom Ziel abweicht, führt der Positionsregelkreis Korrekturmaßnahmen durch.

Bemerkungen: Die Systemreaktion auf Abweichungen von den Sollpositionen ist abhängig von der Drehmoment- und Drehzahlfähigkeit des Systems, von den Reibungs- und Massenträgheiten und von den eingestellten Feed-forward Parametern. Die Werte der Regler Parameter wurden während des Tuning-Prozesses auf der Grundlage der Systemeigenschaften festgelegt. Man erkennt wieder, dass die Abstimmung mit dem kompletten System durchgeführt werden sollte.

Ein weiterer Aspekt der Positionsgenauigkeit betrifft die Art und Weise, wie das Erreichen der Zielposition angenommen wird. Es gibt verschiedene Parameter, die das *Standstill window* beschreiben, wie Sie in *Abbildung 39*.

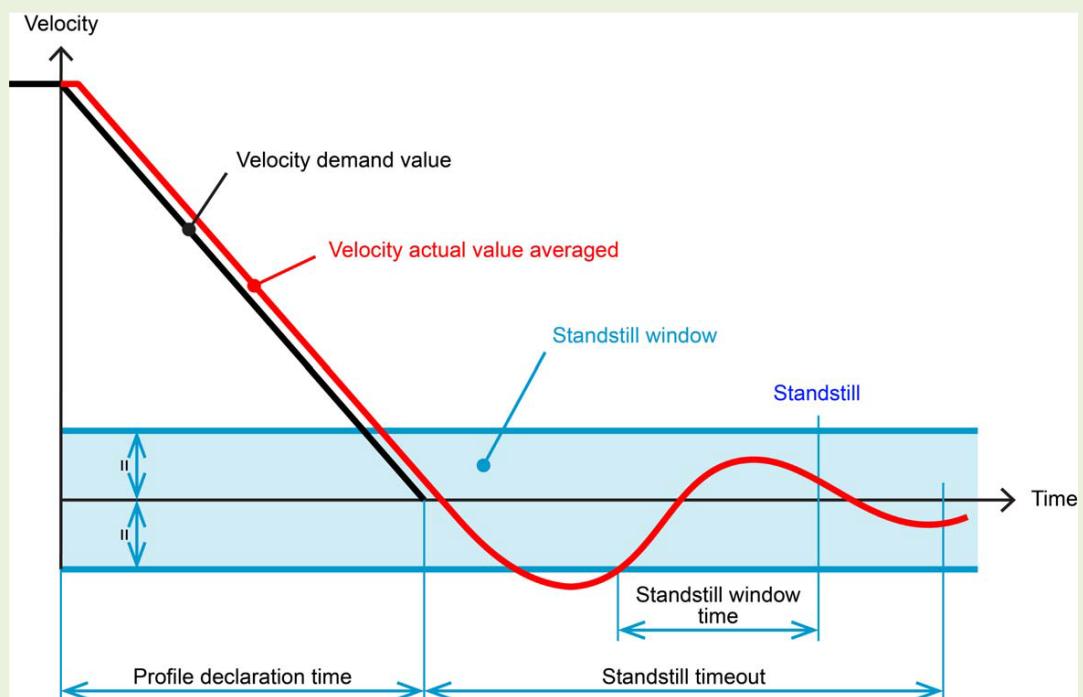


Abbildung 39: Standstill window

Die Parameter, die das Erreichen der Endposition einer Bewegung regeln (aus dem Dokument *EPOS 4 Firmware Specification*). Nur wenn die Drehzahl für eine bestimmte Zeitdauer (*Standstill window*) ausreichend klein (d.h. innerhalb vom *Standstill window*) bleibt, wird angenommen, dass die Zielposition erreicht ist.

8.2 Homing

Zielsetzung: Ausführen einer Referenzfahrt (Homing) auf einen mechanischen Anschlag und auf ein digitales Signal.
Kennen der verschiedenen Parameter des Homing.

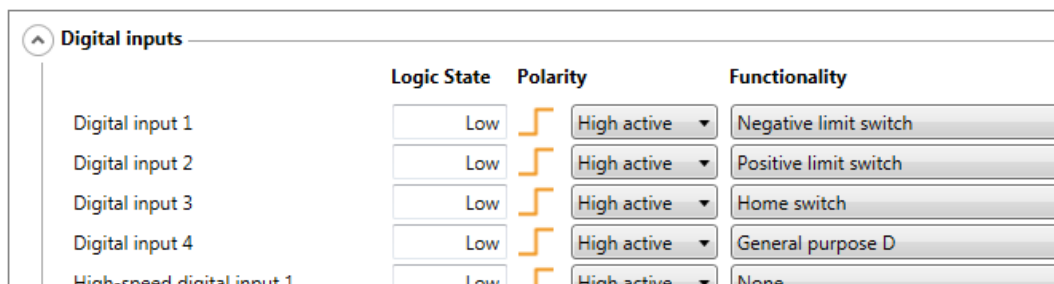
Homing (Referenzfahrt) auf mechanischen Anschlag

Zunächst führen wir eine Referenzfahrt mit der Stromschwellenmethode *Current Threshold* durch. Wir simulieren einen blockierenden mechanischen Anschlag durch einen steigenden Strompegel. Folgen Sie diesen Schritten:

- Schritt 1 Wählen Sie das Werkzeug *Homing Mode* und aktivieren Sie den Modus.
- Schritt 2 Wählen Sie aus dem Dropdown die Homing Methode *Current Threshold Negative Speed*.
- Schritt 3 Setzen Sie den Parameter *Speed for switch search* auf 60 rpm.
- Schritt 4 Setzen Sie den Parameter *Current threshold for homing mode* auf 1000 mA.
- Schritt 5 Aktivieren Sie mit *Enable* die EPOS4.
- Schritt 6 Starten Sie das *Homing* mit der Taste *Start*.
- Schritt 7 Imitieren Sie das Anfahren eines mechanischen Hindernisses, indem Sie die Motorscheibe mit dem Finger stoppen. Sobald der Motorstrom über den Schwellenwert von 1000 mA ansteigt, geht die EPOS4 davon aus, dass das Ende des Bewegungsbereichs erreicht ist und definiert dies als Home- oder Referenzposition.

Homing (Referenzfahrt) auf Endschalter

Bevor wir mit der Referenzfahrt an einem Endschalter beginnen, müssen wir sicherstellen, dass wir ein Referenzfahrtsignal an einem der digitalen Eingänge zur Verfügung haben. Wählen Sie dazu das Werkzeug *I/O-Monitor*. Stellen Sie die digitalen Eingänge 1 bis 4 gemäß *Abbildung 40* ein, was deren Standardeinstellung ist. Sie können die Eingangssignale einstellen, indem Sie die entsprechenden Schalter auf der angeschlossenen Leiterplatte aktivieren.



	Logic State	Polarity	Functionality
Digital input 1	Low	High active	Negative limit switch
Digital input 2	Low	High active	Positive limit switch
Digital input 3	Low	High active	Home switch
Digital input 4	Low	High active	General purpose D
High-speed digital input 1	Low	High active	None

Abbildung 40: Konfiguration der digitalen Eingänge für die Homing Übung

Führen Sie für diese Homing-Methode einfach die folgenden Schritte aus:

- Schritt 1 Wählen Sie aus der Dropdown-Liste die Homing-Methode *Negative Limit Switch and Index*.
- Schritt 2 Setzen Sie den Parameter *Speed for switch search* auf 100 rpm.
- Schritt 3 Setzen Sie den Parameter *Speed for zero search* auf 10 rpm.
- Schritt 4 Aktivieren (*Enable*) Sie die EPOS4.

- Schritt 5 Starten Sie die Referenzfahrt mit der Taste *Start*.
- Schritt 6 Simulieren Sie den anzufahrenden Endschalter, indem Sie den digitalen Eingang 1, d.h. den Schalter 1 auf der Platine aktivieren.

Dadurch nimmt die EPOS4 an, dass das Ende des Bewegungsbereichs erreicht wird, wenn der Schalter aktiviert wird. Sie addiert dann zum Indeximpuls des Encoders eine Rückbewegung und definiert diese als Home- oder Referenzposition.

Andere Homing Methoden und Parameter

Wiederholen Sie die Referenzfahrtübung mit verschiedenen Parametern und versuchen Sie, Antworten auf die folgenden Fragen zu finden (siehe die grünen *Motion-Control*-Kästchen auf der nächsten Seite.):

1. An welchen mechanischen Positionen bleibt die Scheibe auf der Motorwelle stehen?
2. Was ändert sich, wenn Sie eine Referenzfahrt mit ... *Positive Speed* wählen?
3. An welchen mechanischen Positionen bleibt die Scheibe an der Motorwelle stehen, wenn Sie eine Homing-Methode mit ... *and Index* wählen?
4. Wenn Sie ein Homing mit Index ausführen: Was ist der Unterschied zwischen den Parametern *Speed for switch search* und *Speed for zero search*?
5. Wie funktioniert die Homing-Methode mit *Actual Position*?
Und was ist die Funktionalität der Schaltfläche *Define Position*?
6. Welchen Einfluss hat die Einstellung vom *Home offset move distance*?
7. Welche Bedeutung hat der Parameter *Home position*?

Beachten Sie auch die Statusanzeige unten rechts.



Was ist Homing und warum wird es benötigt?

Die Ausgangssignale von Inkremental Encodern können nur relative Positionsinformationen oder Positions differenzen anzeigen. Für eine absolute Positionierung muss das System zunächst auf eine Null- oder Referenzposition initialisiert werden. Dieser Vorgang wird als Referenzfahrt oder Homing bezeichnet, die durch das Anfahren einer vordefinierten Position realisiert wird, auf die sich alle anderen Positionen beziehen. Solche Referenzpunkte werden üblicherweise durch einen induktiven Referenz- oder Endschalter oder durch einen mechanischen Anschlag realisiert. Das Erreichen eines mechanischen Anschlags führt zu einem Stromanstieg. Wenn der Strom einen vordefinierten Schwellenwert überschreitet, wird angenommen, dass die Home-Position erreicht ist.

Homing mit zusätzlicher Bewegung zum Indexkanal

Um die Wiederholbarkeit und Genauigkeit der Referenzposition zu erhöhen, kann eine zusätzliche Fahrt zum ersten Zustandsübergang des Indexkanals (Kanal I oder Z des Encoders) durchgeführt werden. Der Indexkanal I liefert einen Impuls pro Umdrehung, was zu einer absoluten Position innerhalb einer Motorumdrehung führt. Durch diese Art der Referenzfahrt spielt es keine Rolle mehr, ob der Referenzschalter etwas früher oder später betätigt wird; die Referenzposition ist nun sehr genau durch den Zustandsübergang des Indexkanals definiert.

Seien Sie jedoch in realen Systemen vorsichtig. Die Verwendung des Index als präzise Grundstellung kann nur funktionieren, wenn der Index-Impuls nicht in den Referenzschalter-Positionsfehler fällt. Deshalb versuchen Maschinenbauer, den Einsatz dieser Funktion zu vermeiden. Wenn Sie den Motor oder den Encoder austauschen müssen, wird der Index nicht an der gleichen Position stehen und Sie müssen das System neu kalibrieren und die programmierten Positionen ändern.

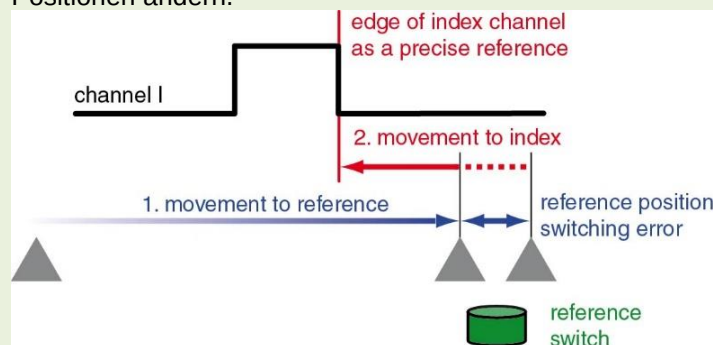


Abbildung 41: Prinzip des Homing mit dem Indexkanal des Encoders

Nach dem Anfahren des Referenzschalters (bzw. des mechanischen Anschlags) erfolgt eine Rückfahrt auf die Signalfanke des Indexkanals. Dadurch ergibt sich immer die gleiche Referenzposition, unabhängig von möglichen Positionsfehlern beim Betätigen des Referenzschalters (z.B. durch Verschmutzung).

8.3 Cyclic Synchronous Position (CSP) Modus

Zielsetzung: Erkunden Sie die Unterschiede zwischen *Profile Position Mode* und *Cyclic Sync Position Mode*.

Der Modus *Cyclic Sync Position* (CSP) ermöglicht eine Positionierung ohne Profil, d.h. der eingebaute Trajektoriengenerator der EPOS4 greift nicht ein. Stattdessen wird das Positionsprofil im Master-System erzeugt. Die Grundidee des *CSP-Mode* besteht darin, viele Positionssollwerte in kurzen Zeitintervallen (ms oder darunter) über den Feldbus an die EPOS4 zu senden, typischerweise unter Verwendung der PDO-Kommunikation. Durch die kurze Zeitdifferenz zwischen den Sollwerten ergeben sich nur kleine Positionsänderungen. Die EPOS4 nutzt die vielen Zielpositionen als sofortige Sollwerte im Positionsregelkreis. Der *CSP-Mode* ermöglicht dem Master-System, mehrere Achsen zu synchronisieren. Im *CSP-Mode* ist nur eine absolute Positionierung möglich, es findet keine relative Bewegung statt. Überprüfen Sie daher die Ist-Position, bevor Sie einen CSP-Befehl ausführen. Im EPOS Studio kann der *CSP-Mode* nur sehr eingeschränkt genutzt werden, da es keine Möglichkeit gibt, die Menge von Sollwerten zu erzeugen, die zur Nachahmung eines Bewegungsprofils benötigt werden. Es können lediglich einzelne Sprungantworten ausgewertet werden. Um der Natur des CSP Rechnung zu tragen, sollten nur kleine Positionsschritte getätigt werden.

Auch hier zunächst ein Rezept für die Auswertung des *CSP-Mode*.

- Schritt 1 Setzen Sie die Ist-Position auf 0, z.B. durch Homing. Dies vermeidet Probleme mit einem zu hohen Schleppfehler.
- Schritt 2 Konfigurieren Sie den Datenrekorder:
Wählen Sie *Position Demand Value* und *Position Actual Value* als aufgezeichnete Parameter. Ich empfehle, feste Achsenskalen, z.B. -50 inc bis 150 Inc.
Optional können Sie auch den *Actual Current* aufzeichnen.
Stellen Sie als Sampling Time das Minimum 0,4 ms ein. Das ist die Zykluszeit des Positionsregelkreises.
Wählen Sie im Bereich Trigger die Option *Single Trigger* und stellen Sie den Typ auf *Motion states*, den Trigger auf *Drive follows command* und die Verzögerung auf 10 ms
- Schritt 3 Wechseln Sie zum Reiter *Cyclic Synchronous Position Mode*, aktivieren Sie den Modus und aktivieren Sie den Regler mit *Enable*.
- Schritt 4 Starten Sie eine Bewegung von 100 qc (*Apply Position*) und beobachten Sie die Reaktion des Motors.
- Schritt 5 Untersuchen Sie die Bewegung im aufgezeichneten Diagramm. Beobachten Sie, wie sich der Positions-Sollwert sofort ändert und wie die Ist-Position versucht zu folgen, überschwingt und sich schliesslich an der befohlenen Zielposition einpendelt.

Weitere Übungen

Wiederholen Sie den Vorgang mit grösseren und kleineren Positionsbefehlen. Passen Sie die Skala des Datenrekorders entsprechend an.

Ändern Sie die Parameter für die Abstimmung der Positionsregelung (hart vs. weich) im Assistenten für die Regelungsabstimmung. Welchen Einfluss haben sie auf die Systemreaktion?



Cyclic Synchronous Modus

In den zyklisch synchronen Positions- und Drehzahlmodi (*CSP* und *CSV*) erfolgt die Trajektoriengenerierung im kommandierenden Master. Neue Sollwerte werden mit einer ausreichend hohen Rate vom Master an den Motion Controller (z.B. eine EPOS4) gesendet, typischerweise über PDO-Kommunikation (siehe Kapitel 10.6).

Die Modi *CSP* und *CSV* sind nützliche Betriebsarten für eine Situation, in der die Achse als Slave-Achse agiert, die durch progressive Positions- oder Drehzahl-Sollwerte ohne grosse Sprünge kommandiert wird. Entsprechend wird der eingebaute Trajektorien-generator nicht benötigt. Typischerweise werden die zyklisch synchronen Betriebsarten verwendet, um die Bewegung mehrerer Achsen zu synchronisieren, wobei die kombinierte Pfadplanung für alle Achsen im Master durchgeführt wird. Der Master generiert neue Positionssollwerte für alle Achsen basierend auf der Ist-Position und der Drehzahl jeder Achse. Dies erfordert eine leistungsfähige Kommunikation (typischerweise EtherCAT), um die Istwerte aller Achsen zu lesen und neue Befehle mit einer Abtastrate im Millisekunden-Bereich zu senden. Die CANopen-Kommunikation kommt sehr schnell an die Grenzen, wenn es darum geht, mehrere Achsen mit einigermaßen schnellen Synchronisationszeiten zu synchronisieren. Die zyklisch synchronen Betriebsarten mit EPOS4 erfordern daher sehr oft die wesentlich performantere EtherCAT-Kommunikation.

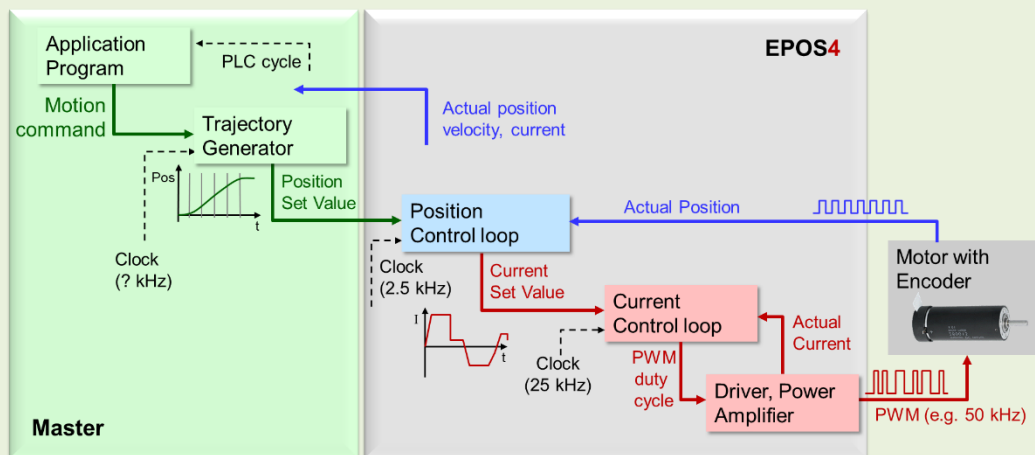


Abbildung 42: Diagramm des CSP Modus

Der Vergleich zur Abbildung 38 zeigt, dass die Trajektoriengenerierung nun im Master-System erfolgt. Das EPOS muss dem Master aktuelle Positionsinformationen zur Verfügung stellen.

Following Error Fenster und CSP Mode

Setzen Sie das Schleppfehler-Fenster auf einen Wert, der grösser ist als der grösste zu erwartende Positionsschritt. Andernfalls kann es zu einer sofortigen Fehlermeldung kommen, sobald die EPOS versucht, den Befehl auszuführen, da der neue Sollwert (die Zielposition) zu stark von der Ist-Position (der Startposition) abweicht.



Cyclic Synchronous Position (CSP) Mode

Wie oben erwähnt, erlaubt das EPOS Studio nicht, den *Cyclic Synchronous Position (CSP) Mode* richtig auszunutzen. Es gibt keine Möglichkeit, einen Schwall von Positions-Sollwerten in kurzen (ms) Intervallen zu senden. Dennoch können wir die Reaktion auf einen einzelnen Bewegungsschritt untersuchen, um ein Gefühl dafür zu bekommen.

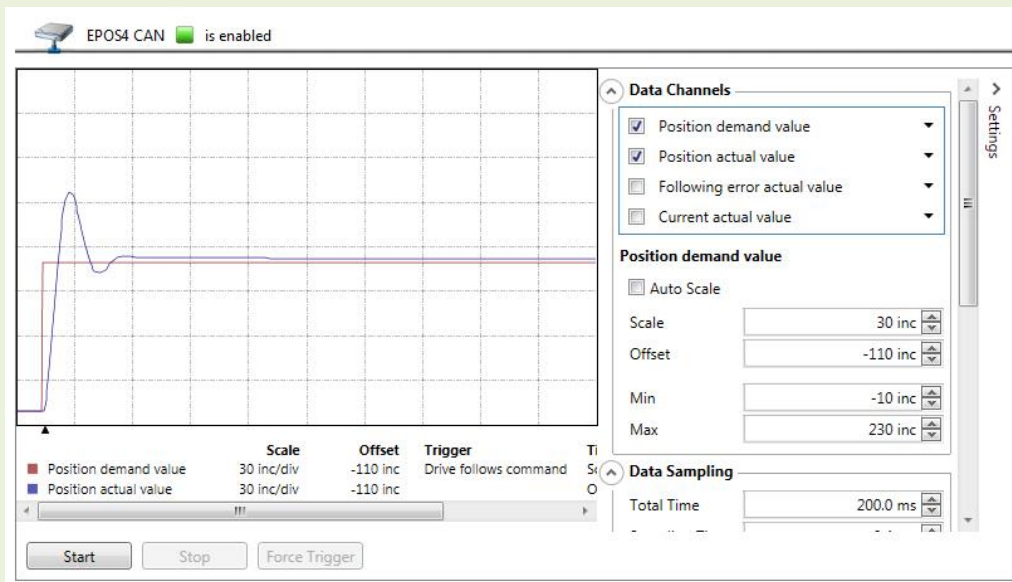


Abbildung 43: Position Recording im CSP Modus

Die rote und die blaue Kurve zeigen den geforderten Positionsschritt von 100 inc bzw. die tatsächliche Systemreaktion. Es gibt ein erhebliches Überschwingen und es dauert etwa 30 ms, bis der Motor zur Ruhe kommt. Das Überschwingen ist auf die recht steife Regelabstimmung und die Motordrehzahl von bis zu ca. 200 min^{-1} in diesem Fall zurückzuführen.

Beachten Sie, dass in realen CSP-Mode-Applikationen das Bewegungsprofil im Master neue Positionsschritte erzeugt, die deutlich kleiner als 100 inc wären. Dies gilt insbesondere gegen Ende einer Bewegung, was zu einer viel sanfteren Bewegung und weniger Überschwingen führt (siehe Abbildung 44).

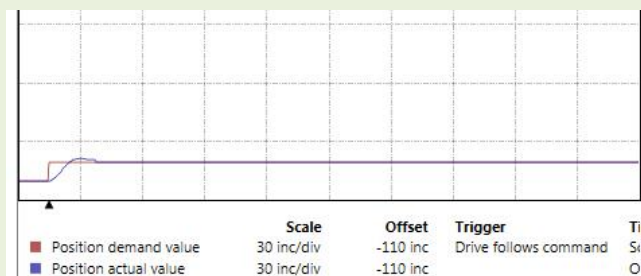


Abbildung 44: Positionsaufzeichnung im CSP-Modus für einen kleinen Schritt von 10 Inc. Alle Parameter gleich wie in Abbildung 43.

8.4 Profile Velocity Modus

Zielsetzung: Einstellen einer Drehzahl in der EPOS4 (Drehzahlregelung).
Überwachung der Drehzahl im eingebauten *Data Recorder*.

Die nächste zu untersuchende Betriebsart ist der *Profile Velocity Mode*. Dieser Modus steuert die Motordrehzahl und nicht die Position. Die neue Zieldrehzahl wird durch eine Drehzahlrampe - oder wenn Sie möchten ein Drehzahlprofil, daher der Name - mit einstellbaren Beschleunigungs- oder Verzögerungswerten erreicht. Der *Profile Velocity Mode* ermöglicht sanfte Drehzahländerungen.

Als Ausgangspunkt für Ihre Erkundung im *EPOS Studio* führen Sie das folgende Drehzahlprofil aus und zeichnen die Bewegung auf.

- Schritt 1 Wählen Sie den Reiter *Data Recorder* und konfigurieren Sie den Rekorder.
Setzen Sie die Datenkanäle wie folgt:
Kanal 1: *Velocity demand value*
Kanal 2: *Velocity actual value*
Kanal 3: *Velocity actual value averaged*
Setzen Sie die minimale *Sampling Time* auf 0.4 ms (entsprechend der Abtastfrequenz der Drehzahlregelung von 2.5 kHz.)
Setzen Sie den Trigger auf *Continuous Acquisition* und drücken Sie auf *Start*.
- Schritt 2 Wählen Sie den Reiter *Profile Velocity Mode* und aktivieren Sie den Modus.
- Schritt 3 Stellen Sie die *Target Velocity* auf 2000 rpm ein. Wenn Sie einen relativ niedrigen Beschleunigungswert von z.B. 200 rpm/s einstellen, können Sie beobachten, wie sich die Scheibe auf der Motorwelle beschleunigt.
- Schritt 4 Aktivieren Sie den Controller (*Enable*) und starten Sie die Bewegung mit der Schaltfläche *Set velocity*.
- Schritt 5 Beobachten Sie die Drehzahlssignale auf dem Datenrekorder. Zur besseren Ansicht und Analyse der Signale stoppen Sie den Datenrekorder und stellen Sie die Skalen der verschiedenen Drehzahlssignale ein. Stellen Sie z.B. die Min- und Max-Werte der Drehzahlsskalen auf 1900 rpm bzw. 2100 rpm ein. Beachten Sie, dass bei 2000 rpm eine Motorumdrehung 30 ms dauert. Lesen Sie im nächsten grünen Motion-Control-Kasten über das *Verständnis von Drehzahlssignalen am Datenrekorder*.

Aufzeichnen der Drehzahlrampe

Die Beschleunigung des Motors kann verfolgt werden, wenn im Datenrekorder die Methode *Single Trigger* ausgewählt ist.

- Schritt 1 Stellen Sie den Typ auf *Parameter values*, den *Trigger* auf *Channel 1*, den *Modus* auf *Rising above value* 10 rpm und *Delay* auf 100 ms. Um den gesamten Drehzahlbereich anzuzeigen, stellen Sie die Skalen aller Kanäle auf -100 rpm bis 2500 rpm. Behalten Sie alle anderen Einstellungen des Datenrekorders bei. Drücken Sie *Start*.
- Schritt 2 Wählen Sie den Reiter *Profile Velocity Mode* und aktivieren Sie diesen.
- Schritt 3 Belassen Sie *Target Velocity* auf 2000 rpm aber erhöhen Sie die Beschleunigung auf 20'000 rpm/s.

- Schritt 4 Aktivieren (*Enable*) Sie den Controller und starten Sie die Bewegung mit dem Knopf *Set velocity*.
- Schritt 5 Beobachten Sie die Drehzahlsignale auf dem Datenrekorder.
Notieren Sie sich das Diagramm für den späteren Vergleich mit dem CSV-Modus (Kapitel 8.5).
Bemerkung: Das durchschnittliche Drehzahlsignal verzögert sich aufgrund der benötigten Mittelungszeit.

Weitere Übungen

Wiederholen Sie den Vorgang mit grösseren und kleineren Drehzahlen. Passen Sie die Skalen des Datenrekorders entsprechend an.

Ändern Sie die Abstimmungsparameter der Drehzahlregelung (hart vs. weich) im Assistenten für das Regler-Tuning im *Regulation Tuning* Assistenten.

Wie ist ihr Einfluss auf die Systemreaktion?



Verständnis von Drehzahlsignalen am Datenrekorder

Wenn Sie die Drehzahlssignale auf dem Datenrekorder betrachten, sehen Sie möglicherweise etwas, das dem folgenden Bild ähnelt. (Achten Sie darauf, dass Sie für alle Signale die gleiche Skala eingestellt haben; z.B. durch feste *Min Max* Werte.)

Offensichtlich ist die horizontale rote Linie der Soll-Drehzahlwert von 2000 rpm. Das blaue Signal ist der Drehzahl-Istwert des Sensors. Es sieht recht verrauscht aus und zeigt bei näherer Betrachtung, dass es in Schritten von etwa 37 rpm springt. Zusätzlich gibt es eine gewisse Periodizität im Signal mit einer Periode von 30 ms. Dieses periodische Verhalten (mit 4 Schwankungen pro 30 ms) ist im grünen Signal, das die gemittelte Drehzahl anzeigt, deutlicher zu erkennen.

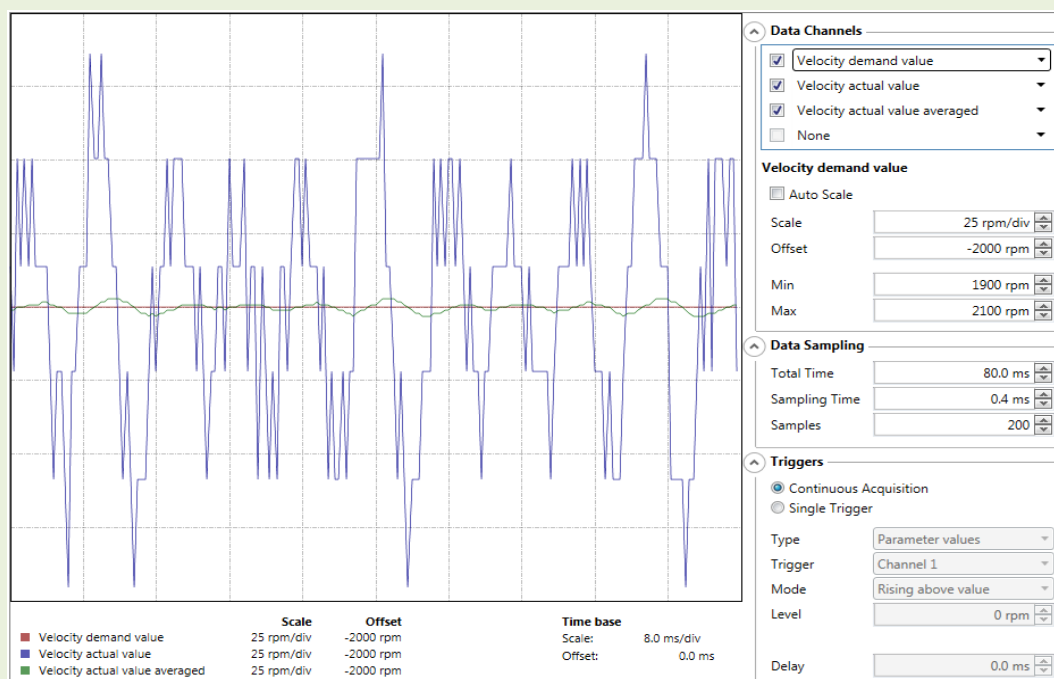


Abbildung 45: Aufgezeichnete Drehzahlssignale eines DCX Motors mit EASY Encoder

Woher kommen die Schritte von 37 rpm im blauen Signal? Sicherlich ist der Motor nicht in der Lage, seine Drehzahl so zu verändern, wie es dieses Signal suggeriert; seine mechanische Zeitkonstante ist zu gross (> 4 ms). Es handelt sich vielmehr um ein Quantisierungsphänomen bei der Drehzahlmessung, das daraus resultiert, dass die Drehzahl aus der Positionsänderung pro Abtastintervall von 0,4 Millisekunden berechnet wird. Die Position kann sich in Schritten von 1 Encoder-Inkrement (inc) ändern. 1 inc pro 0,4 Millisekunden entspricht 150'000 inc pro Minute und dies entspricht 36,6 rpm bei einem Encoder mit 4096 inc pro Umdrehung.

Wenn Sie also die tatsächliche Drehzahl Ihres Motors wissen wollen, ist es besser, den gemittelten Drehzahl-Istwert, den *Velocity actual value averaged* zu betrachten, d.h. das grüne Signal im obigen Diagramm. Das macht mehr Sinn.

Übrigens, die Drehzahlgenauigkeit des gemittelten Signals liegt bei einer Drehzahl von 2000 rpm bei etwa 3 rpm Abweichung. Und das ist schon ziemlich gut!

Woher kommt die Periodizität?

Die Periodizität im Drehzahl-Istwert (blaues Signal) und im gemittelten Drehzahl-Istwert (grünes Signal) korrelieren mit der Umdrehung des Motors. Bei 2000 rpm dauert eine Umdrehung der Welle genau 30 ms, wie an der Signalperiodizität zu erkennen ist.

Man kann sich zwei mögliche Gründe für diese Oszillationen vorstellen: Sie rühren entweder von einer Unregelmässigkeit im Motor her, z.B. von einer erhöhten Reibung an einigen Motorpositionen, oder von Unvollkommenheiten des Messgeräts, d.h. des Encoders.

Hier denke ich, dass es eher der zweite Grund ist. Von EASY Encodern ist bekannt, dass sie diese Art von Welligkeit mit 4 Perioden pro Motorumdrehung haben, die aus dem Interpolationsverfahren im Encoder resultiert. Ein Vergleich mit einem sehr genauen optischen Encoder zeigt deutlich den Unterschied (Abbildung 46)

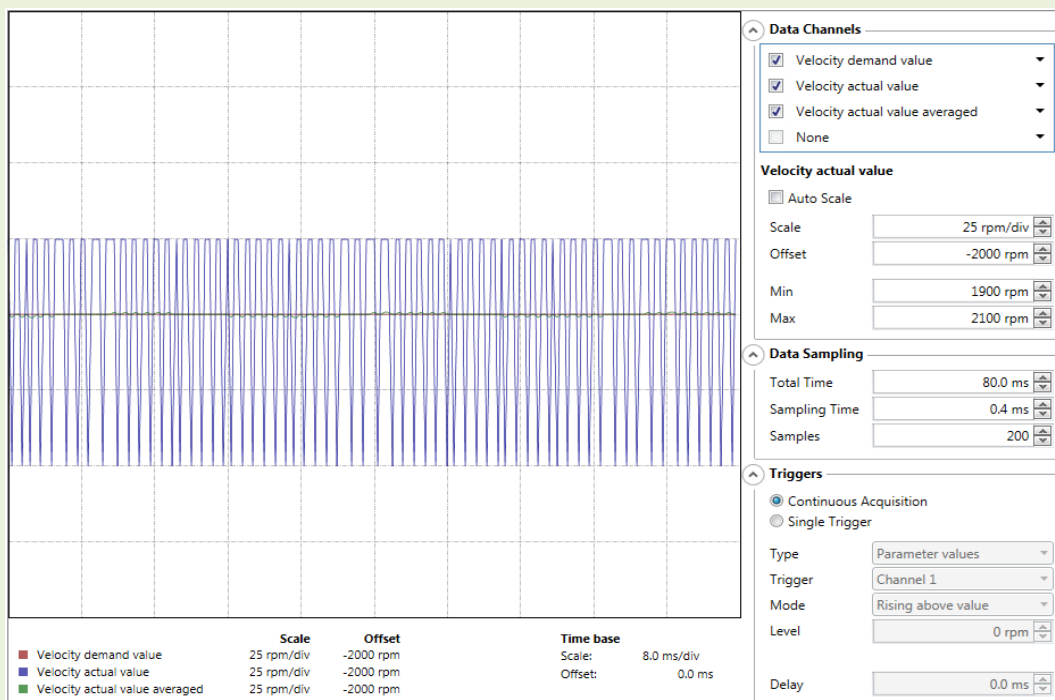


Abbildung 46: Aufgezeichnete Drehzahlssignale an einem Motor mit optischem Encoder (500 cpt)

8.5 Cyclic Synchronous Velocity (CSV) Modus

Zielsetzung: Erkennen Sie die Unterschiede zwischen *Profile Velocity Mode* und *Cyclic Sync Velocity (CSV) Mode*.

Die Betriebsart *Cyclic Synchronous Velocity (CSV) Mode* funktioniert ähnlich wie die Betriebsart *Cyclic Synchronous Position (CSP)* im Abschnitt 8.3. Der CSV-Modus verwendet jedoch die Drehzahlregelung (anstelle der Positionsregelung im CSP-Modus). Auch hier generiert der Master die Bewegungstrajektorie und die Drehzahlrampen und füttert den EPOS4-Slave mit einem Schwall von sich allmählich ändernden Drehzahlsollwerten.

Als Ausgangspunkt für den Vergleich der Drehzahländerungen im *Profile Velocity Mode* und im *Cyclic Synchronous Velocity (CSV)-Modus* verwenden wir dieselben Drehzahländerungs- und Datenrekorder-Einstellungen wie in Kapitel 8.4 zur Überwachung der Drehzahlrampe.

- Schritt 1 Wählen Sie den Reiter *Data Recorder* und konfigurieren Sie den Rekorder.
Setzen Sie die Datenkanäle wie folgt:
Kanal 1: *Velocity demand value*
Kanal 2: *Velocity actual value*
Kanal 3: *Velocity actual value averaged*
Um den gesamten Drehzahlbereich anzuzeigen, stellen Sie die Skalen aller Kanäle auf -100 rpm bis 2500 rpm ein.
Setzen Sie die minimale *Sampling Time* auf 0.4 ms (entsprechend der Abtastfrequenz der Drehzahlregelung von 2.5 kHz.)
Setzen Sie den Trigger auf *Single Trigger* und stellen Sie den *Type* auf *Parameter values*, den Trigger auf Kanal 1, den Modus auf *Rising above value* 10 rpm und die Verzögerung *Delay* auf 100 ms. und drücken Sie auf *Start*.
- Schritt 2 Wählen Sie den Reiter *Cyclic Sync Velocity Mode* und aktivieren Sie den Modus.
- Schritt 3 Stellen Sie die *Target Velocity* auf 2000 rpm ein.
- Schritt 4 Aktivieren Sie den Controller (*Enable*) und starten Sie die Bewegung mit der Schaltfläche *Apply velocity*.
- Schritt 5 Beobachten Sie die Drehzahlsignale auf dem Datenrekorder. Vergleichen Sie die Drehzahlsignale mit der Beschleunigung im *Profile Velocity Mode* (Abschnitt 8.4).

Weitere Übungen

Wiederholen Sie den Vorgang mit grösseren und kleineren Drehzahlschritten. Passen Sie die Einstellungen des Datenrekorders entsprechend an.



Cyclic Synchronous Velocity (CSV) Mode

Cyclic Synchronous Torque (CST) Mode

Mögliche Regler-Architekturen des CSV- bzw. CST-Modus sind in der folgenden Abbildung dargestellt.

Die Architektur des CSV-Modus ist dem CSP sehr ähnlich, nur werden der Positionsregelkreis und die Positionssollwerte durch einen Drehzahlregelkreis und Drehzahl-Sollwerte ersetzt. Im Master ist sehr oft ein übergeordneter Positionsregler aktiv, der die Drehzahl-Sollwerte erzeugt.

In der Betriebsart CST arbeitet der Motion Controller als Stromregler, der die Endstufe für den Motorbetrieb nutzt. Der Positions-Regelkreis ist im Master selbst geschlossen.

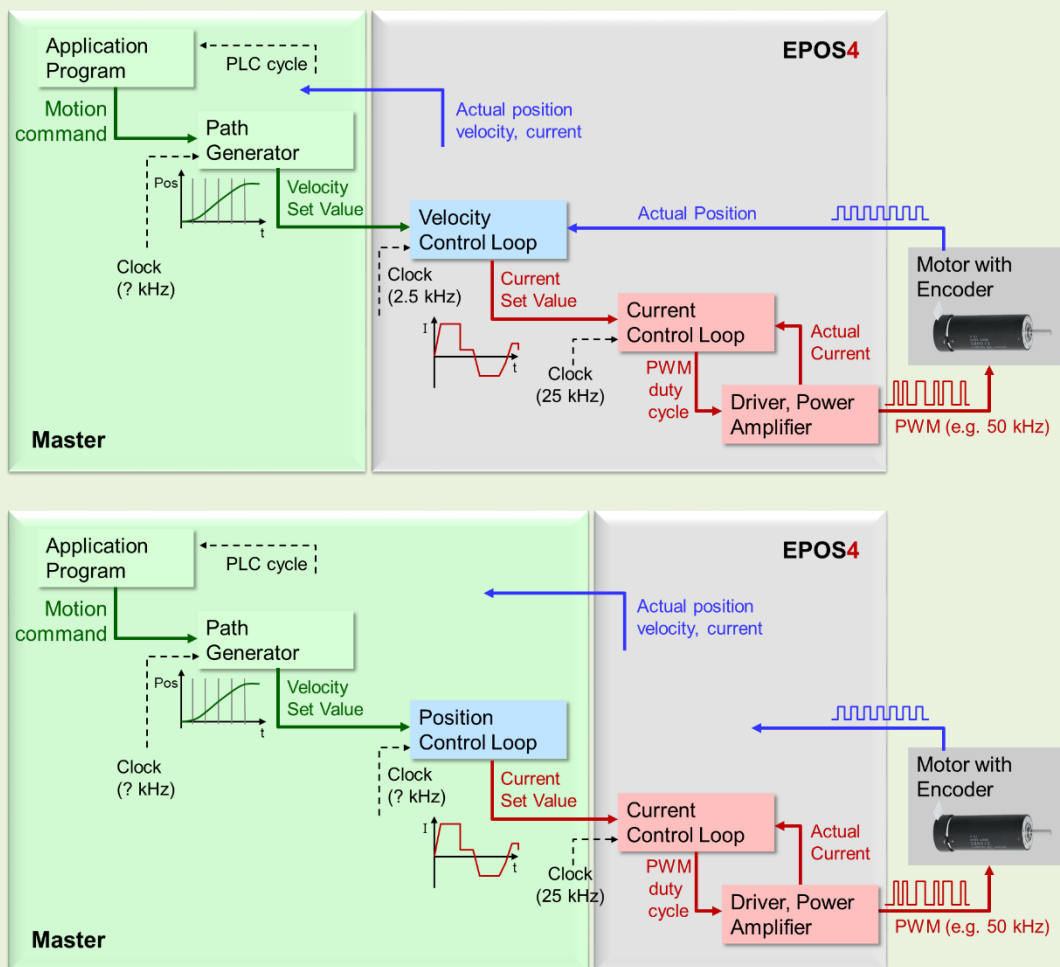


Abbildung 47: Schematischer Aufbau der Betriebsarten CSV (oben) und CST (unten).
Vergleichen Sie auch mit Abbildung 42.

8.6 Cyclic Synchronous Torque (CST) Modus

Zielsetzung: Erfahren Sie, wie sich der Motor nur bei Stromregelung verhält.

Im *Cyclic Synchronous Torque (CST) Mode*, wird der Motorstrom geregelt. Da der Motorstrom proportional zum Drehmoment ist, ist die Regelung des Motorstroms gleichbedeutend mit der Regelung des Drehmoments. Wir sprechen direkt den innersten Regelkreis an - den Stromregelkreis (siehe *Abbildung 38*) – und umgehen den Drehzahl- bzw. Positions-Regelkreis.

Erkunden Sie den CST-Modus, z. B. mit den folgenden Schritten.

- Schritt 1 Wechseln Sie zum *Cyclic Sync Torque Mode* und aktivieren Sie diesen.
- Schritt 2 Aktivieren Sie den Controller mit *Enable*.
- Schritt 3 Stellen Sie den Wert für das Zieldrehmoment *Target torque* auf 10 % des Motor-Nennmoments ein und drücken Sie die Schaltfläche *Apply Torque*. Was passiert nun? Beobachten Sie die Drehzahl und die aktuellen Ausgangswerte im rechten Teil des Bildschirms. Drücken Sie *Apply zero*, um den Motor anzuhalten.
- Schritt 4 Blockieren Sie die Motorwelle mit der Hand und drücken Sie die Schaltfläche *Apply Torque*. Wie ändern sich nun die Ausgangswerte? Warum?
- Schritt 5 Was spüren Sie an der blockierten Welle, wenn Sie das Drehmoment um 50% oder mehr erhöhen? Vergessen Sie nicht, die Schaltfläche *Apply Torque* zu drücken.
- Schritt 6 Legen Sie bei blockierter Welle ein Drehmoment von ca. 200% auf. Spüren Sie, wie nach einigen Sekunden der Thermoschutz aktiviert wird, d.h. der Strom reduziert wird? Verfolgen Sie die Änderung des Stromwerts im Abschnitt «Output» auf der rechten Seite. Wie lange können Sie ein Drehmoment von 300% aufbringen?



Weglaufen oder Durchdrehen im Stromregelungsmodus

Wenn die Motorwelle in der Stromregelung nicht blockiert wird, kann die Motordrehzahl bis zu sehr hohen Werten ansteigen. Warum passiert das?

Stromregelung bedeutet Drehmomentregelung. Der Regler versucht, den Strom (Drehmoment) auf dem eingestellten Wert zu halten.

Drehmoment am Motor bedeutet in der Regel Beschleunigung => Eine höhere Drehzahl führt zu einer höheren induzierten Spannung (Gegen-EMK), die der angelegten Spannung entgegenwirkt => Es wird mehr angelegte Spannung benötigt, um den Strom auf dem eingestellten Wert zu halten => Höhere Spannung bedeutet, dass der Motor weiter beschleunigt => höhere Gegen-EMK => mehr angelegte Spannung benötigt => dadurch höhere Drehzahl => ...

Dadurch läuft die Motordrehzahl bis zur maximalen Drehzahl, die mit der gegebenen Versorgungsspannung möglich ist, weg. Ein solches Weglaufen kann auftreten, wenn nicht durch einen übergeordneten Regelkreis (Drehzahl- oder Positionsregelung), einen mechanischen Anschlag oder eine zusätzliche Sicherheitsdrehzahlbegrenzung (wie die im Abschnitt *Limits* des Inbetriebnahme-Assistenten eingestellte maximal zulässige Drehzahl) gebremst wird.

Mit dem *Cyclic Synchronous Torque (CST)-Modus* haben wir die Erkundung der grundlegenden Motion-Control-Funktionalität abgeschlossen.

9 Verwendung des I/O Monitor Werkzeuges

Zielsetzung Verwendung des *I/O-Monitors* und praktisches Verständnis der I/O-Funktionalität.
(soweit die Funktionalitäten bereits implementiert sind, Juni 2020)

An der *EPOS4* sind mehrere digitale Ein- und Ausgänge sowie zwei analoge Ausgänge vorhanden. Diese I/O's sind in erster Linie für spezielle Aufgaben in der Peripherie der jeweiligen Bewegungsachse vorgesehen. Ein Beispiel für einen Eingang ist ein Endschalter, der bei einem Linearantrieb anzeigt, dass die Mechanik den sicheren Betriebsbereich verlassen hat. Das Einfahren in den Endschalter führt dazu, dass der Motion Controller diesen speziellen Fehler an das Mastersystem meldet. Endschalter können auch als Referenz für eine Referenzfahrt verwendet werden; in diesem Fall, ohne einen Fehler zu provozieren, wie Sie es bei der Ausführung einer Referenzfahrt (Homing) Kapitel 8.2.

Die I/O's des EPOS sind frei konfigurierbar. Sie können sie für die vordefinierten Funktionalitäten (z.B. Home-Switch-Funktionalität) oder für jeden anderen beliebigen Zweck (*General Purpose*) verwenden. Sie könnten einen der digitalen Eingänge verwenden, um einen bestimmten Teil Ihres Programms in der Master-SPS zu starten oder das Signal eines Temperatursensors mit einem analogen Eingang auszulesen oder das Ende einer Subroutine auf einem der digitalen Ausgänge zu signalisieren.

Es gibt zwei Arten von digitalen I/O's: Die Standard-Eingänge (genannt *Digital inputs 1* bis *4* und *Digital outputs 1* bis *2*) werden für Statussignale wie Bereit oder als Endschalter verwendet. Die *High-speed digital inputs* dienen zum Einlesen von schnell variierenden Signalen, z. B. den Signalen von Analog- oder Absolutencodern.

Im EPOS Starter Kit auf Ihrem Tisch können Schalter auf der Platine die Standard-Digitaleingänge aktivieren. Die analogen Eingänge lesen die Spannung von Potentiometern und die digitalen Standardausgänge lassen farbige LEDs aufleuchten. Es sind keine Hochgeschwindigkeits-I/O's angeschlossen. Öffnen Sie das I/O-Monitor-Tool und probieren Sie die folgenden Übungen aus, um zu erfahren, wie die Ein- und Ausgänge funktionieren.

9.1 Eingänge

Anzeigen des analogen Eingangs

Drehen Sie die Potentiometer auf der Platine und beobachten Sie den Messwert auf dem *I/O Monitor*.

Bemerkung: Zwischen der Aktivierung der Eingangsschalter oder Potentiometer auf der Platine und der Reaktion des *I/O Monitors* gibt es eine Verzögerung. Es müssen viele Informationen über die USB-Kommunikation übertragen werden. Daher dauert es eine Weile, bis die Anzeige des I/O Monitors im *EPOS Studio* aufgefrischt wird. Keine Sorge, die *EPOS4* internen Zustände der Eingänge werden sofort geändert.

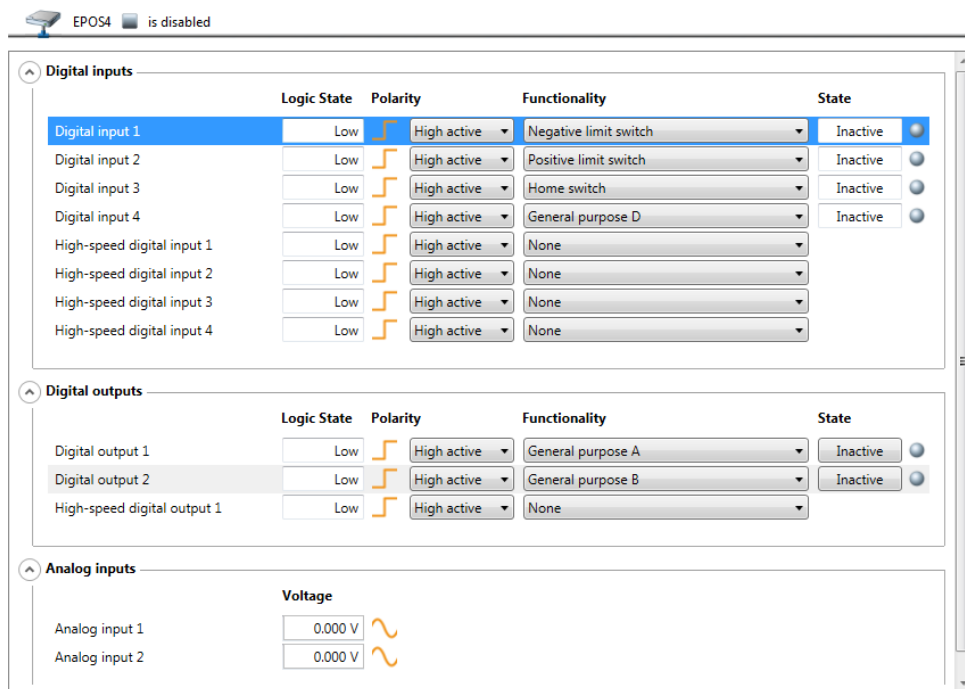


Abbildung 48: Das I/O Monitor Werkzeug der EPOS4

Digitale Eingänge

Standardmässig ist der *Digital input 4* als *General purpose* eingestellt, d.h. er kann zum Einlesen eines beliebigen digitalen Signals verwendet werden, das typischerweise im Masterprogramm verarbeitet werden soll. Versuchen Sie Folgendes

- Schritt 1 Aktivieren Sie den *DigIn4* Schalter auf der Platine und beobachten Sie die Reaktion im *I/O Monitor*. Wie ändern sich die Parameter *Logic State* und *State change*?
- Schritt 2 Setzen Sie die Polarität des digitalen Eingangs auf Low aktiv. Was passiert mit den Parametern *Logic State* und *State*? Welcher dieser beiden Parameter steht für den physikalischen Signalpegel (Spannung) am Eingang, welcher für den sich daraus ergebenden booleschen Signalpegel?

Als Nächstens beobachten wir die *Digital inputs 1* bis *4* welche eine Funktionalität zugewiesen haben.

- Schritt 1 Aktivieren Sie den *DigIn1* oder den *DigIn2* Schalter (limit switches) auf der Platine und beobachten Sie die Reaktion im *I/O Monitor*. Was passiert mit der LED auf der Rückseite der EPOS4-Steuerplatine oder der EPOS-Statusanzeige ganz oben im Fenster "*I/O Monitor*"? Prüfen Sie auch das Statusfenster am unteren Rand des EPOS Studio-Fensters auf Fehler und Warnungen.
- Schritt 2 Setzen Sie den aktivierten Schalter zurück und löschen Sie den Fehler im Statusfenster (rechter Mausklick). Beobachten Sie die Reaktion der LED.
- Schritt 3 Aktivieren Sie den *DigIn3* Schalter auf der Platine (Home switch) und beobachten Sie die Reaktion auf dem *I/O Monitor*. Warum tritt in diesem Fall kein Fehler auf?

- Schritt 4 Ändern Sie die Funktionalität des *Digital input 4* auf *Drive enable* mit *High active* Polarität. Aktivieren Sie den entsprechenden Schalter auf der Platine. Wechseln Sie in das Werkzeug *Profile Position Mode* und versuchen Sie, die EPOS4 zu deaktivieren. Was können Sie beobachten? Deaktivieren Sie die EPOS4 mit dem Schalter und versuchen Sie, ihn im *Profile Position Mode* Werkzeug zu aktivieren.
- Schritt 5 Ändern Sie die Funktionalität des *Digital input 4* auf *Quick stop*. Beobachten Sie am besten das Verhalten der Motordrehzahl im *Profile Velocity Mode* (alternativ im *Profile Position Mode* bei langen Positionierfahrten). Wie unterscheidet sich der Hardware *Quick stop* (*Digital input 4*) vom Software *Quick stop* (Schaltfläche in den Kommunikationsfenstern von EPOS Studio)?



Achsenbezogene digitale Eingangsfunktionalitäten

Limit Switches und Home Switch

Wir haben bereits gesehen, wofür *Limit Switches* verwendet werden können. Sie begrenzen den mechanischen Arbeitsbereich auf beiden Seiten. Sehr häufig werden induktive Näherungsschalter als Endschalter eingesetzt, die bei Annäherung an ein metallisches Teil ihren Ausgangszustand ändern. Bei Betätigung eines Endschalters geht der Antrieb davon aus, dass er aus dem zulässigen Bereich herausgefahren ist. Es wird ein Fehlersignal erzeugt, der Antrieb kommt zum Stillstand und wird gesperrt.

Sie können einen der Endschalter für das *Homing* nach dem Einschalten verwenden. Es wird kein zusätzlicher *Home-Switch* benötigt, und es ist klar, in welcher Richtung ein Endschalter zu suchen ist; am entsprechenden positiven oder negativen Ende des mechanischen Fahrwegs. Bei Verwendung des Endschalters für das *Homing* wird natürlich kein Fehler erzeugt.

Für das *Homing* können Sie auch einen speziell entwickelten *Home Switch* und dessen Eingangsfunktionalität verwenden.

Gerätefreigabe

In den meisten Fällen werden die Freigabe und Sperrung der Endstufe in der Steuerung durch die Befehle des Programms im Master geregelt, der die Anwendung steuert. Die Funktionalität des *Drive enable* Eingangs bietet eine alternative Möglichkeit, die Endstufe durch ein externes Digitalsignal zu steuern. Ein aktiver Zustand an diesem Eingang schaltet die Endstufe ein, ein inaktiver Zustand schaltet sie aus.

Beachten Sie, dass die Definition eines Eingangs als Antriebsfreigabe jeden Freigabe- oder Sperrbefehl, der über die Feldbusverbindung vom Master gesendet wird, ausser Kraft setzt. Diese Funktionalität kann in Situationen nützlich sein, in denen Sie den Motor stromlos schalten wollen, ohne auf das Programm im Master zuzugreifen. Seien Sie jedoch vorsichtig, diese Funktion entspricht nicht den Sicherheitsvorschriften!

Schnellstopp

Der *Quick stop* bringt die Achse mit der Schnellstopp-Bremsrampe zum Stillstand und hält die Endposition. Der Motor ist weiterhin bestromt, d.h. die Endstufe bleibt freigegeben. Software-Schnellstopp und Hardware-Schnellstopp arbeiten parallel; wenn einer der beiden aktiviert wird, stoppt der Motor und hält die Position.



Achsenbezogene digitale Eingangsfunktionalitäten

Touch probe

Der *Touch probe* Eingang zeichnet die Ist-Position auf, wenn dieser digitale Eingang aktiviert ist. Eine typische Situation für diese Funktionalität ist die folgende: Wenn ein Objekt auf einem Förderband einen bestimmten Punkt (z.B. einen Lichtvorhang) passiert, wird dies dem Motion Controller über den Eingang *Touch probe* (Positionsmarker) signalisiert und die entsprechende Position des Förderbandes gespeichert. Das Prozesssteuerungsprogramm kann nun den Offset addieren, damit das Objekt auf dem Band an einer definierten Position stoppt.

Achsenbezogene digitale Ausgangsfunktionalitäten

Ready/Fault

Der *Ready/Fault* Ausgang kann verwendet werden, um den korrekten Betrieb des Geräts zu signalisieren. Denken Sie an die übliche grüne und rote Lampe an Produktionsmaschinen.

Holding Brake und Set Brake (GPIO)

Haltebremsen werden verwendet, um eine Position zu halten, ohne dass der Motor bestromt wird. Dies ist nützlich, um Energie in Anwendungen zu sparen, bei denen der Antrieb lange Zeit in der gleichen Position verharren muss oder bei denen das Halten einer Position ein hohes Drehmoment erfordert (z.B. bei einem Vertikalantrieb). Haltebremsen halten die Position, wenn sie nicht bestromt sind. Daher können Haltebremsen auch als Notstoppvorrichtung bei Stromabschaltung verwendet werden.

Der leistungsfähige ***Holding Brake*** Ausgang der EPOS4 ermöglicht die direkte Ansteuerung von Bremsen ohne zusätzliche Spannungsversorgung unter Berücksichtigung der Reaktionszeit der Bremsenansteuerung.

Der ***Set Brake (GPIO)*** Ausgang dient zur Ansteuerung einer vom Host (Master) betriebenen Bremse gemäss der Spezifikation CiA 402

Position Compare

Der *Position Compare* Ausgang ist noch nicht implementiert (**Juni 2020**). *Position Compare* ermöglicht das Senden von Signalen an einer vordefinierten Position oder Positionsintervallen. Dies kann nützlich sein, um ein anderes Gerät oder einen Prozess durch ein digitales Signal zu aktivieren, wenn Ihre Achse eine bestimmte Position durchfährt.

Achsenbezogene analoge Ein- und Ausgangsfunktionalitäten

Analog set value ist ab der Firmware-Version 0160h implementiert.

1. Sollwert Strom
2. Sollwert Drehzahl

Die folgenden Funktionalitäten des Analogausgangs sind jedoch noch nicht implementiert. (**Juni 2020**)

1. Strommonitor, Positionsmonitor, Drehzahlmonitor
2. Temperaturwächter

9.2 Ausgänge

Als erste Übung wollen wir einen digitalen Ausgang einstellen und lernen, wie wir den physikalischen Ausgang verbergen können.

- Schritt 1 Klicken Sie auf die Spalte Funktionalität in der Tabelle und setzen Sie die digitalen Ausgänge 1 und 2 als *General purpose A* bzw. *General purpose B*. Setzen Sie die *Polarität* auf *High aktiv*.
- Schritt 2 Setzen Sie den *State* eines der digitalen Ausgänge auf *Aktive* und beobachten Sie die Reaktion der LED-Ausgänge auf der Platine. Wie verhält es sich mit der Anzeige des *Logic State*.
- Schritt 3 Ändern Sie *Polarity* auf *Low Active*. Wie wirkt sich dies auf die LED-Ausgabe aus? Welchen Einfluss hat die Änderung des Status auf *Inactive* oder *Active*.

Jetzt weisen wir einem digitalen Ausgang eine bestimmte Funktionalität zu.

- Schritt 1 Setzen Sie die Funktionalität des *Digital output 1* (oder eines anderen) auf *Ready*. Was zeigen der *Logic State* und die LED auf der Platine an? Bemerkung: Der *State* kann nicht mehr manuell geändert werden.
- Schritt 2 Beobachten Sie das Verhalten des *Logic State* und LED, wenn Sie einen Fehler erzeugen, z.B. durch Betätigen eines Endschalters.
- Schritt 3 Stellen Sie die Funktionalität des *Digital output 2* auf *Holding Brake*. Der Ausgang wird gesetzt (LED leuchtet, Bremse gelöst), wenn die Endstufe gesperrt ist. Beachten Sie, dass Sie Parameter des Bremsenbetriebs, d.h. die Anstiegs- und Abfallzeiten der Haltebremse, festlegen können. (Weitere Informationen finden Sie im Dokument *Firmware Reference*).
- Schritt 4 Aktivieren Sie die Endstufe, z.B. im *Profile Position Mode*. Was passiert mit dem Ausgang "Holding Brake"?



Best Practice

Parameter speichern

Im Falle einer Stromabschaltung gehen Ihre Parametereinstellungen im EPOS Studio verloren, wenn sie nicht dauerhaft in der EPOS4 gespeichert werden. Speichern Sie die geänderten I/O-Parameter (und andere) im Werkzeug *Object Dictionary*. Klicken Sie mit der rechten Maustaste auf die Tabelle des *Object Dictionary* und wählen Sie *Save All Parameter*. Alternativ können Sie den *Startup* Assistenten öffnen und die Schaltfläche *Finish* drücken.

Teil 5: CANopen und EtherCAT Kommunikation

Die *EPOS4* ist ein *CANopen* Gerät, genauer gesagt ein *Motion Controller Device* nach der CiA 402 Definition. Der Zweck dieses Kapitels ist, Ihnen die wichtigsten Funktionen und Ideen im Zusammenhang mit *CANopen*-Geräten zu vermitteln. Dadurch erhalten Sie ein besseres Verständnis, wie die Kommunikation organisiert ist. Dies wird Ihnen später bei der Programmierung nützlich sein.

Dies ist jedoch nur eine Einführung und keine vollständige Beschreibung dessen, was *CANopen* ist. Weitere Informationen finden Sie auf der CiA (CAN in Automation) Website (www.can-cia.org).

Bei der optionalen EtherCAT Kommunikation - genauer gesagt verwendet die *EPOS4 CANopen over EtherCAT* - nutzt man den viel schnelleren Informationsfluss von EtherCAT, behält aber die grundlegende Geräte- und Objektstruktur bei, wie sie im folgenden Kapitel 10 beschrieben wird. Daher halten wir die zusätzlichen Informationen zu EtherCAT eher grundlegend.

10 Eine Einführung zu CANopen und CAN

CANopen ist ein Kommunikationsprotokoll, das auf einen Feldbus angewendet wird, der normalerweise *CAN* ist. *CAN* deckt die unteren Kommunikationsebenen ab und konzentriert sich auf technische Aspekte, während *CANopen* die Bedeutung der Daten und Geräte im Netzwerk definiert. Mit anderen, etwas vereinfachten Worten: der *CAN*-Bus beschreibt das Vehikel, wie Daten übertragen werden, und das *CANopen*-Protokoll beschreibt, was für Daten übertragen werden.

CANopen ist ein Standard, der von der unabhängigen Nutzerorganisation *CAN in Automation* gepflegt und betreut wird.

Wir wollen uns hier nicht damit auseinander setzen, wie genau Meldungen übertragen und empfangen werden. Alles was wir wissen müssen, ist, dass *CANopen* und *CAN* einen zuverlässigen und kostengünstigen Rahmen zur Verfügung stellen, um Informationen zwischen Geräten auszutauschen.

Einige Einschränkungen:

Maximale Anzahl Teilnehmer pro Bus	127 Teilnehmer
Maximale Bitrate	1 Mbit/s bis zu 40 m Buslänge Alle Knoten müssen die selbe Bitrate haben!
Typische Länge eines CAN-Telegramms	100 - 130 Mikrosekunden bei maximaler Bitrate

Das wichtigste Konzept, mit dem wir uns beschäftigen müssen, ist das Gerätemodell *Device Model* und das darin enthaltene Objektverzeichnis *Object Dictionary*. Doch zunächst ein paar Worte zu *CAN* und *CANopen*.

10.1 CAN

CAN steht für *Controller Area Network*. Dieser Feldbus wurde ursprünglich für Anwendungen im Automobilbau entwickelt, wird nun aber auch in anderen Bereichen wie der Industrieautomation und in Medizinalgeräten eingesetzt.

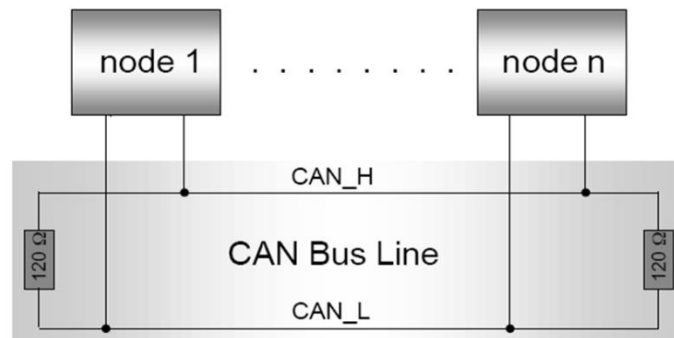


Abbildung 49: Physikalisches Layout des CAN Bus

Auf physikalischer Ebene ist jedes Gerät – Knoten *Node* genannt und mit einer eigenen Adresse *node number* versehen - parallel an die Busleitung angeschlossen. Der Bus überträgt die Signale auf einer differentiellen Leitung, die auf beiden Seiten einen geeigneten Abschluss benötigt, um Signalreflexionen zu vermeiden (120 Ohm Widerstände). Die Signalbits benötigen eine gewisse Zeit, um sich über den gesamten Bus auszubreiten. Daher ist die Übertragungsrate abhängig von der Buslänge. Die maximale Bitrate von 1 Mbit/s kann nur bis zu einer Buslänge von ca. 40 m erreicht werden.

"Der *Transfer Layer* repräsentiert den Kern des CAN-Protokolls." Dort werden Struktur, Übertragung und Empfang der Nachrichten definiert. "Der *Transfer Layer* ist verantwortlich für das Bit-Timing und die Synchronisation, das Verpacken von Nachrichten, die Arbitrierung und Bestätigung, die Fehlerentdeckung, -anzeige und -eingrenzung." (übersetzt aus dem englischen Wikipedia)

Kurz gesagt, bietet CAN einen Rahmen, in dem Mikrocontroller und andere Geräte sicher und zuverlässig miteinander kommunizieren können.

10.2 CANopen

"In den Begriffen des OSI-Modells implementiert CANopen die höher liegenden Kommunikationsschichten; *Network layer* und höher. Der CANopen-Standard besteht aus einem Addressierungsschema, einigen kleinen Kommunikationsprotokollen und einem Anwendungslayer, das durch Geräteprofile festgelegt ist. Die Kommunikationsprotokolle unterstützen das Netzwerkmanagement, die Geräteüberwachung und die Kommunikation zwischen den Knoten, einschliesslich einer einfachen Transportschicht für die Segmentierung und Desegmentierung von Nachrichten. Die unteren Protokoll-Level (Data-Link und physikalische Schichten) werden meist als Controller Area Network (CAN) implementiert, obwohl auch andere Kommunikationarten (wie Ethernet Powerlink, EtherCAT) das CANopen Geräteprofil einsetzen können." (übersetzt aus dem englischen Wikipedia)

Kurz gesagt, benützt CANopen den CAN-Rahmen, um sinnvolle Meldungen zwischen definierten Knoten *Devices* in einem Netzwerk zu senden.

10.3 CANopen Geräteprofil

Das CANopen Geräteprofil ist ein zentrales Element von CANopen. Es beschreibt, wie die Knoten in einem Netzwerk strukturiert sein müssen. Man kann sich ein CANopen Gerät aus drei Untereinheiten bestehend vorstellen: einer *Kommunikationseinheit*, dem Objektverzeichnis (*Object dictionary*) und dem *Anwendungsteil* (Abbildung 50). Zusätzlich braucht es eine Zustandsmaschine, um das Gerät zu starten und zurückzusetzen. Sie muss die Zustände *Initialisierung*, *Pre-operational*, *Operational* und *Angehalten* enthalten. Typischerweise wird der *Pre-operational* Zustand zur Konfiguration benutzt, während *Echtzeitkommunikation auf den Zustand Operational beschränkt ist*.

Die *Kommunikationseinheit* ist für die Kommunikation mit den anderen Knoten im Netzwerk zuständig. Sie enthält alle CAN und CANopen Kommunikationsmerkmale: Netzwerk-Management-Informationen empfangen und senden, als auch Daten ins Objektverzeichnis schreiben oder von ihm lesen.

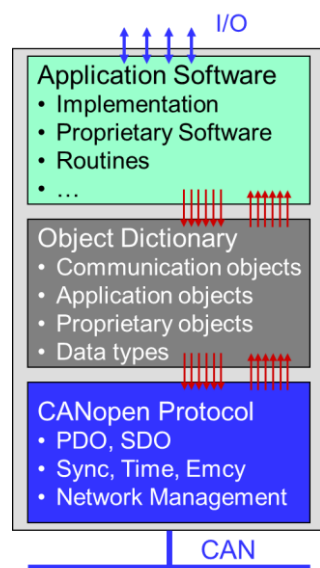
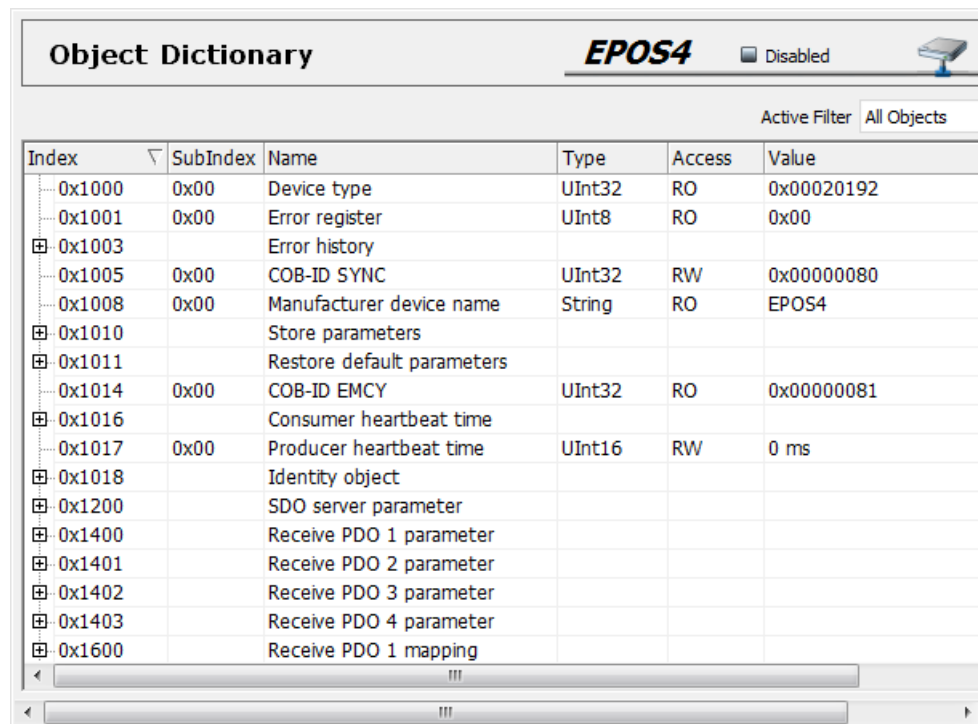


Abbildung 50: CANopen Geräteprofil

Der Anwendungsteil *Application part* führt die gewünschte Funktion des Geräts aus. Dies kann ein Motion Controller wie die *EPOS4*, ein I/O-Gerät, eine speicherprogrammierbare Steuerung (SPS) oder eine beliebige andere Funktionalität sein. Die Anwendung wird über Parameter und Variablen im *Object Dictionary* konfiguriert und gesteuert.

Objektverzeichnis (Object Dictionary)

Das Objektverzeichnis ist das Herz des Geräts. Es verbindet die Welt der CANopen-Kommunikation und die Anwendung. Die Anwendung benützt die Daten des Objektverzeichnisses als Eingangsgrößen, um ihre Aufgabe auszuführen, und sie schreibt die Resultate und Ausgangsgrößen in dasselbe Objektverzeichnis. Andererseits beruht die ganze Kommunikation zwischen Geräten im CANopen-Netzwerk auf dem Datenaustausch zwischen den entsprechenden Objektverzeichnissen. Somit läuft Kommunikation mit einem CANopen-Gerät im Endeffekt darauf hinaus, Parameter zwischen entsprechenden Objektverzeichnissen auszutauschen.



Index	SubIndex	Name	Type	Access	Value
0x1000	0x00	Device type	UInt32	RO	0x00020192
0x1001	0x00	Error register	UInt8	RO	0x00
0x1003		Error history			
0x1005	0x00	COB-ID SYNC	UInt32	RW	0x00000080
0x1008	0x00	Manufacturer device name	String	RO	EPOS4
0x1010		Store parameters			
0x1011		Restore default parameters			
0x1014	0x00	COB-ID EMCY	UInt32	RO	0x00000081
0x1016		Consumer heartbeat time			
0x1017	0x00	Producer heartbeat time	UInt16	RW	0 ms
0x1018		Identity object			
0x1200		SDO server parameter			
0x1400		Receive PDO 1 parameter			
0x1401		Receive PDO 2 parameter			
0x1402		Receive PDO 3 parameter			
0x1403		Receive PDO 4 parameter			
0x1600		Receive PDO 1 mapping			

Abbildung 51: Zugriff auf das Object Dictionary der EPOS4 in der Registerkarte EPOS Studio Tools.

Das *Object Dictionary* besteht aus einer Tabelle von Parametern, die alle Eigenschaften des Geräts beschreiben: seine Kommunikationskanäle, seine Konfiguration und Einstellungen, aber auch die Anwendungskonfiguration, Eingangs- und Ausgangsdaten.

Ein Eintrag im Objektverzeichnis ist definiert durch

1. **Address** (16-bit Index und 8-bit Subindex)
2. **Name** ein String, der den Eintrag beschreibt
3. **Type** der Datentyp des Parameters
4. **Access rights** Zugriffsrechte: lesen und schreiben (RW), nur lesen (RO), nur schreiben (WO) oder konstant (*const*)
5. **Value** Wert des Parameters

Die grundlegenden Datentypen für Werte im Objektverzeichnis wie Booleans, Integers und Gleitkommazahlen sind im CANopen-Standard definiert. Daneben sind aber auch zusammengesetzte Datentypen wie Arrays, Records und Strings möglich.



Einschränkungen bei Variablentypen von EPOS-Systemen

Numerische Variablen können nur Ganzzahlen *Integer* unterschiedlicher Grösse sein.

Firmware Specification

Alle Objekte der EPOS4 sind in der *Firmware Specification* beschrieben. Dort finden Sie auch Informationen über den Wertebereich und die Bedeutung der Parameterwerte (Einheiten).

Standard CANopen Geräteprofile

Objektverzeichnisse können nicht nach Belieben definiert werden. Erstens gibt es bestimmte Regeln zu beachten, welche Daten wo im Objektverzeichnis zu finden sind. Zweitens gibt es Standard-Geräteprofile für eine Vielzahl von Anwendungen. Die EPOS4 folgt dem CiA 402 Profil für Antriebe und Motion Control.

Profile number	Device class
CiA 401	Generic I/O Modules
CiA 402	Drives and Motion Control
CiA 404	Measuring devices and Closed Loop Controllers
CiA 405	IEC 61131-3 Programmable Devices
CiA 406	Rotating and Linear Encoders
CiA 408	Hydraulic Drives and Proportional Valves
CiA 410	Inclinometers
CiA 412	Medical Devices
CiA 413	Truck Gateways
CiA 414	Yarn Feeding Units (Weaving Machines)
CiA 415	Road Construction Machinery
CiA 416	Building Door Control
CiA 417	Lift Control Systems
CiA 418	Battery Modules

Abbildung 52: Einige Standard CANopen Geräteprofile (gemäss CiA Webseite).

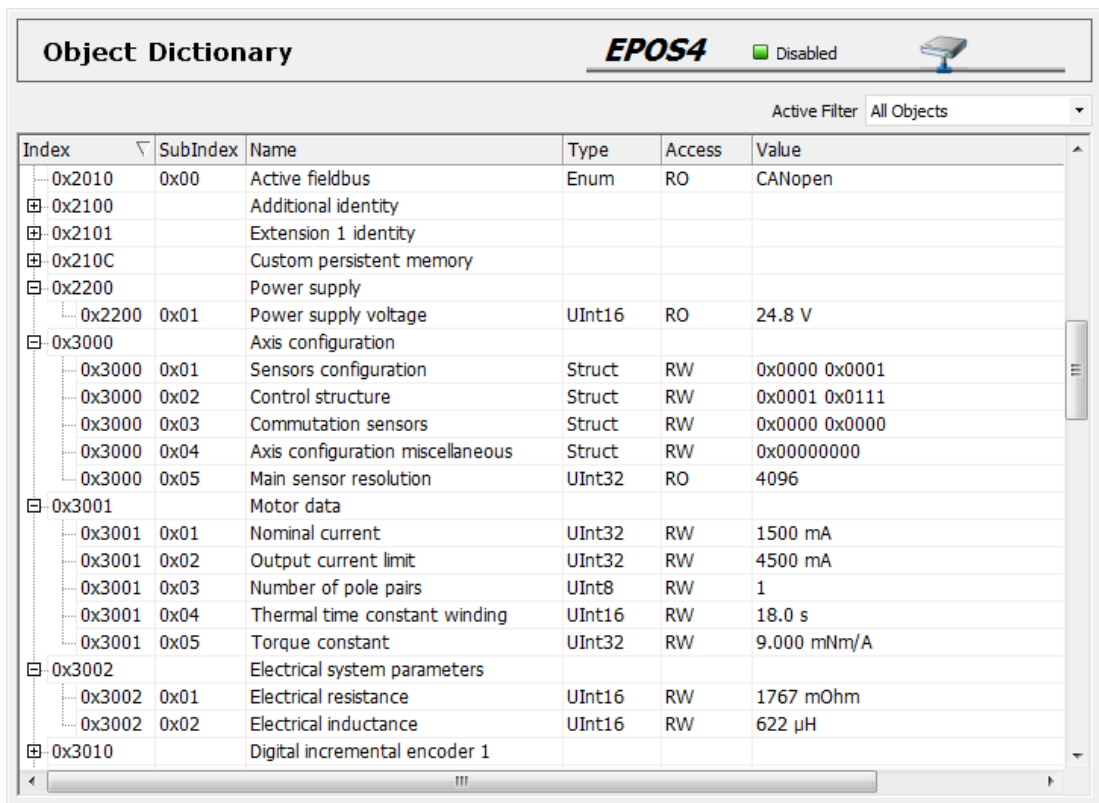
Es ist wichtig zu beachten, dass der Standard in dem Sinne offen ist, dass es einen gewissen Freiheitsgrad gibt, welche der Objekte und Funktionalitäten implementiert sein müssen. Damit ein Gerät dem Standard entspricht, muss es bestimmte Objekte enthalten, aber nicht alle. Ausserdem steht es dem Hersteller frei, zusätzliche Objekte und Funktionalität hinzuzufügen.

10.4 EPOS4 Object Dictionary Werkzeug

Zielsetzung Bearbeiten von Systemparametern im Objektverzeichnis.
Erzeugen eines Objektfilters.

Nehmen wir als Beispiel das Objektverzeichnis der *EPOS4*. Die Motion Control Anwendung verwendet Daten aus dem Objektverzeichnis (z.B. Zielposition) als Eingaben, um eine Bewegung auszuführen und schreibt die Ergebnisse (z.B. Zielposition erreicht) in dasselbe Objektverzeichnis.

Das Objektverzeichnis der *EPOS4* ist als Werkzeug im *EPOS Studio* zugänglich. Öffnen Sie einfach das *Object Dictionary* und aktivieren Sie den Objektfilter *All Objects* in der oberen rechten Ecke.



Index	SubIndex	Name	Type	Access	Value
0x2010	0x00	Active fieldbus	Enum	RO	CANopen
0x2100		Additional identity			
0x2101		Extension 1 identity			
0x210C		Custom persistent memory			
0x2200		Power supply			
0x2200	0x01	Power supply voltage	UInt16	RO	24.8 V
0x3000		Axis configuration			
0x3000	0x01	Sensors configuration	Struct	RW	0x0000 0x0001
0x3000	0x02	Control structure	Struct	RW	0x0001 0x0111
0x3000	0x03	Commutation sensors	Struct	RW	0x0000 0x0000
0x3000	0x04	Axis configuration miscellaneous	Struct	RW	0x00000000
0x3000	0x05	Main sensor resolution	UInt32	RO	4096
0x3001		Motor data			
0x3001	0x01	Nominal current	UInt32	RW	1500 mA
0x3001	0x02	Output current limit	UInt32	RW	4500 mA
0x3001	0x03	Number of pole pairs	UInt8	RW	1
0x3001	0x04	Thermal time constant winding	UInt16	RW	18.0 s
0x3001	0x05	Torque constant	UInt32	RW	9.000 mNm/A
0x3002		Electrical system parameters			
0x3002	0x01	Electrical resistance	UInt16	RW	1767 mOhm
0x3002	0x02	Electrical inductance	UInt16	RW	622 µH
0x3010		Digital incremental encoder 1			

Abbildung 53: Object Dictionary des Motion Controllers EPOS4
Abgebildet sind einige Einträge bezüglich der maxon-Achsenkonfiguration.

Beim Blättern durch die Liste finden wir die folgenden Gruppen von Einträgen (Index und Sub-Index werden als Hexadezimalzahlen angegeben, daher das Präfix 0x):

1. Index 0x1000 bis 0x1FFF Kommunikationsprofil (im CANopen-Standard)
2. Index 0x2000 bis 0x5FFF maxon EPOS spezifisch (nicht im CANopen-Standard)
3. Index 0x6000 bis 0x9FFF Standard-Geräteprofil (im CANopen-Standard)

Die maxon spezifischen Einträge enthalten unter anderem die Objekte für nicht-CANopen Kommunikation und die Objekte für die Konfiguration der digitalen und analogen Ein- und Ausgänge. Im Standard-Geräteprofil finden Sie alle Informationen, die Motion Control betreffen.

Mit dem Werkzeug *Object Dictionary* haben Sie Zugriff auf die vollständige Liste der Objekte. Sie können den Wert eines jeden Eintrags ändern, sofern er beschreibbar ist. Doppelklicken Sie das Objekt oder verwenden Sie das Kontextmenü (rechter Mausklick).

Best Practice

Refreshing rate des Objektverzeichnisses im EPOS Studio

Die USB-Verbindung ist nicht sehr schnell und nicht in Echtzeit, d.h. sie hat nicht die höchste Priorität auf Ihrem Computer und Sie können nicht vorhersagen, wie lange es dauert, bis die Werte des Objektverzeichnisses aktualisiert werden. Sie können die Reaktionszeit in gewissem Masse beschleunigen, indem Sie die *Refresh Rate* in der Menüleiste minimieren.

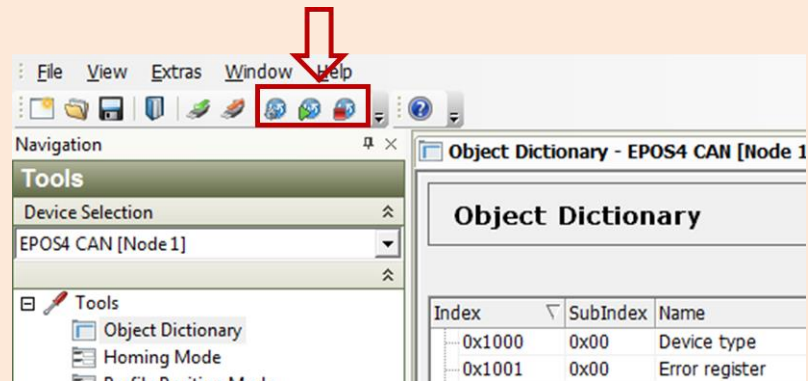


Abbildung 54: Wo Sie die Refresh Rate ändern können



Best Practice

Definieren Sie Ihren eigenen Objektfilter

Den Überblick über das ganze Objektverzeichnis mit seinen vielen Einträgen zu behalten, kann ziemlich herausfordernd sein. Oft interessieren ja nur wenige Objekte. Zu diesem Zweck können Sie einen eigenen Objektfilter erstellen, ähnlich wie den vordefinierten *System Parameter Filter*.

Hier das Rezept um die PID-Regelparameter zu betrachten:

- Schritt 1: Klicken Sie mit der rechten Maustaste auf das Dialogfenster und wählen Sie *Define Object Filter*.
- Schritt 2: Wählen Sie *New*, um einen neuen Filter zu erzeugen.
- Schritt 3: Geben Sie dem Objektfilter einen Namen, z.B. *MyRegGains*.
- Schritt 4: Weisen Sie dem Objektfilter *MyRegGains* mit *Add* die folgenden Objekte zu:
 - 1. *Current control parameter set*
 - 2. *Velocity control parameter set*
 - 3. *Position control parameter set*
- Schritt 5: Speichern des Objektfilters als Datei (optional) und Beenden mit *Exit*.



Parameter speichern

Änderungen im Werkzeug *Object Dictionary* werden erst nach einem Befehl *Save All Parameters* (rechte Maustaste, um das Kontextmenü des Objektverzeichnisses aufzurufen) dauerhaft in der EPOS gespeichert.

Grundsätzlich gibt es drei Sätze von Objektverzeichnissen:

1. Die *Actual parameters* im Arbeitsspeicher (RAM), welche nach einem Stromausfall verloren gehen.
2. Die *Permanent parameters, gespeichert* im Flash-Memory EEPROM, welche auch nach Neustart noch aktiv sind.
3. Die *Default parameters* (d.h. die Werkseinstellung), die sich immer als letzte Sicherung im EEPROM befinden.

10.5 Parameter Import und Export

Das Objektverzeichnis *Object Dictionary* enthält alle Parameter, die ein CANopen Gerät beschreiben. Es ist ein Fingerabdruck des Geräts. Die Parameterliste kann in einem speziellen elektronischen Datenblatt namens *Device Configuration File* gespeichert werden (Parameter Export). So ist es möglich, verschiedene Geräte auf die gleiche Weise zu konfigurieren, indem man einfach das entsprechende elektronische Datenblatt auf die EPOS4 herunterlädt (Parameter Import).

Dazu dient der *Parameter-Export/Import* Assistent im EPOS Studio. Er ermöglicht auch das Wiederherstellen von Standardparametern *Restore Default Parameters*.

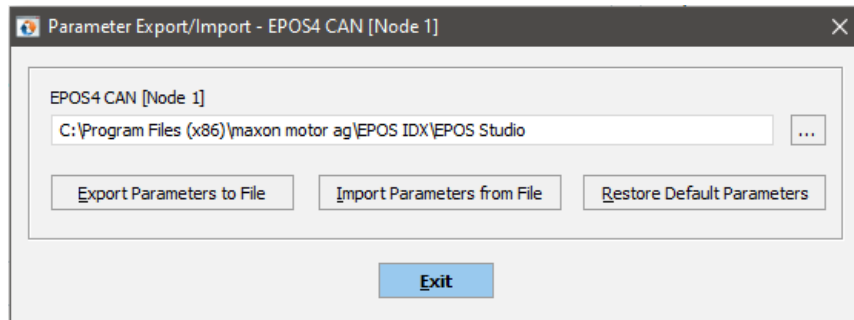


Abbildung 55: Parameter Export/Import Assistent

10.6 CAN Kommunikation

Die Kommunikation in den unteren CAN-Ebenen basiert auf Broadcast-Kommunikation: Jeder Knoten kann Nachrichten senden, die von jedem anderen Knoten abgeholt werden können. Um Datenkollisionen zu vermeiden, hat jedes Telegramm eine definierte Priorität. Starten zwei Nachrichten gleichzeitig, gewinnt diejenige mit dem niedrigeren Prioritätswert und wird zuerst gesendet.

In CANopen ist die Kommunikation spezifischer in dem Sinne, dass es vordefinierte Kommunikationskanäle (jeweils mit einer bestimmten Prioritätsstufe) gibt, die beschreiben, welcher Knoten der Sender und welcher Knoten der Empfänger der Nachricht ist (Peer-to-Peer-Kommunikation).

Bitte beachten Sie auch das Dokument *EPOS 4 Communication Guide*, Kapitel 3

Netzwerkmanagement und spezielle Nachrichten

Die niedrigsten Prioritätswerte (= höchste Priorität) werden Nachrichten zugewiesen, die sicherstellen, dass das Netzwerk korrekt arbeitet und die Kommunikation zuverlässig ist. In dieser Kategorie finden wir Fehlerkontroll- und Notfallnachrichten, Synchronisations- und Zeitstempelnachrichten sowie Netzwerkmanagementnachrichten wie das Booten und Ändern des Betriebszustandes eines Knotens. Die genaue Prioritätsreihenfolge innerhalb dieser Kategorien ist für den Zweck dieses Lehrbuchs nicht von Bedeutung.

Process Data Objekte (PDO)

Die nächste Priorität wird den *Process Data Objects* (PDO) eingeräumt. PDO sind für den Echtzeit-Datenaustausch von kleinen Objekten. Typischerweise handelt es sich dabei um Daten, die sich im Prozess ständig ändern und wie in Echtzeitanwendungen häufig aktualisiert werden müssen.

Um die Übertragung zu beschleunigen, werden die Telegramme klein gehalten, indem unnötige Overhead-Informationen vermieden werden. Um die Buslast zu reduzieren, wird die korrekte Übertragung eines PDO-Telegramms nicht bestätigt.

Der Inhalt jeder PDO Nachricht wird im Voraus definiert, ebenso wie der Ort, von dem aus sie gesendet wird (welches Objekt welchen Knotens) und der Zielort. Diese Konfiguration wird als Mapping bezeichnet, das auch festlegt, wie oft eine PDO-Nachricht gesendet wird: periodisch oder ausgelöst durch ein Ereignis. Das PDO-Mapping ist im Objektverzeichnis der beteiligten Geräte zu finden.

Die CAN PDO Kommunikation wird typischerweise verwendet, um die Prozesseingänge der SPS zu aktualisieren, und es kann mehrere PDO-Kommunikationskanäle zwischen zwei Geräten geben. Beachten Sie, dass die Zykluszeit der SPS-Task und die Aktualisierungsrate, der per PDO gesendeten Daten zwei verschiedene Dinge sind. In bestimmten Anwendungen müssen sie möglicherweise synchronisiert werden. Bemerkung: Alle Befehle, gesendet vom *EPOS Studio*, basieren im Wesentlichen auf SDO-Kommunikation.

Service Data Objekte (SDO)

Die niedrigste Priorität - die höchsten Prioritätswerte - erhalten die *Service Data Objects* (SDO). SDO erlauben die Übertragung einer beliebigen Datengröße. Grosse Nachrichten (> 4 Byte) werden automatisch segmentiert, d.h. in mehrere Telegramme aufgeteilt.

Wie bei der PDO-Kommunikation ist die SDO-Übertragung zwischen zwei festgelegten Knoten definiert. Es werden zwei Kommunikationskanäle benötigt, einer für jede Richtung. Die Telegramme enthalten die Information, welcher Eintrag im Objektverzeichnis geschrieben oder gelesen werden soll. Die Hälfte des Nutzdateninhalts wird bereits durch diese Overhead-Information verbraucht.

SDO ist eine langsame Kommunikation und ist in erster Linie für die Einrichtung des Gerätes gedacht. Da die Bewegung einer einzelnen Achse typischerweise einige hundert Millisekunden dauert, ist die Geschwindigkeit der SDO-Kommunikation für viele Motion-Control Anwendungen ausreichend. Anders kann die Situation aussehen, wenn mehrere Achsen eng koordiniert oder synchronisiert werden müssen.



PDO und SDO in EPOS Systemen

PDO Kanäle in EPOS Systemen sind nur definiert zwischen *Slave* und dem *Master*. Es gibt keine vordefinierten Kommunikationskanäle zwischen *Slaves*.

1. Es gibt einen SDO-Kanal in jeder Richtung zwischen der *EPOS4* und ihrem *Master*.
2. Es gibt 4 PDO-Kanäle zum Senden und 4 PDO-Kanäle zum Empfangen von Prozessdaten auf der *EPOS4*, wie im Objektverzeichnis zu sehen ist.

11 Bemerkungen zu EtherCAT

(Informationen basierend auf <https://en.wikipedia.org/wiki/EtherCAT>)

Auf physikalischer Ebene nutzt EtherCAT die Standard Ethernet Technologie.

Der Hauptunterschied zur CANopen Kommunikation ist die viel höhere Performance von EtherCAT, die bis zu 100 Mbit/s beträgt. Das ist 100-mal schneller als die maximale Bitrate von 1 Mbit/s bei CANopen. So können z.B. die Daten für und von 100 Servo-Achsen mit bis zu 10 kHz aktualisiert werden. Typische Netzwerk-Update-Raten liegen bei 1-30 kHz.

Der EtherCAT-Master kann auf alle Daten, inklusive Name und Datentypen eines EtherCAT-Slaves (z.B. EPOS4) zugreifen. Typischerweise adressiert der EtherCAT-Master mit nur einem Frame das gesamte Netzwerk. Die EtherCAT-Slave-Geräte lesen die an sie adressierten Daten, während das Telegramm das Gerät durchläuft, und verarbeiten die Daten "on the fly". Analog dazu werden Eingangsdaten eingefügt, während das Telegramm durchläuft.

Die EtherCAT-Option der EPOS4 ersetzt lediglich die CAN-Kommunikation durch EtherCAT. Der Kern des EPOS4 ist nach wie vor ein CANopen Gerät mit dem gleichen Objektverzeichnis. Die Lösung heisst *CAN application protocol over EtherCAT* (CoE). Auf Hardware-Ebene sind die EtherCAT Versionen der EPOS4 mit einer speziellen EtherCAT Anschlussplatine oder mit einer EtherCAT-Erweiterungskarte ausgestattet

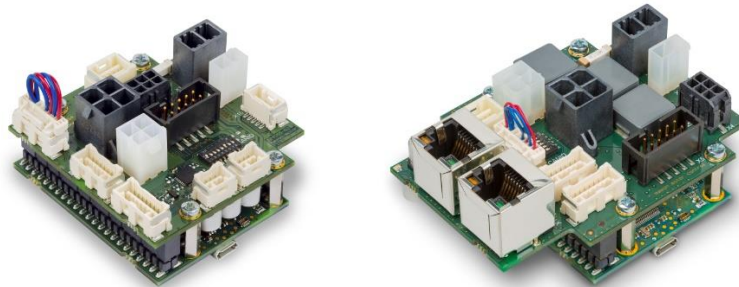


Abbildung 56: EPOS4 50/8 mit CANopen (links) und EtherCAT (rechts) Anschlussplatine

Das schnelle Echtzeit-Kommunikations-CoE bringt Vorteile für anspruchsvolle Anwendungen wie Hochgeschwindigkeits-Synchronisierung mehrerer Achsen, Robotik, Pick & Place. Diese Anwendungen erfordern häufig die Bahnplanung und -steuerung mehrerer Achsen im Master und verwenden die zyklischen Synchronbetriebsarten (CSP, CSV, CST: siehe Kapitel 8.3 ff).

Teil 6: Anhänge, Referenzen und Index

12 Anhänge

12.1 Motor und Encoder Datenblätter

Die maxon-Motor-Encoder Kombination mit der Artikelnummer B7723C9C8D10 besteht aus einem Motor DCX 22 S mit ENX-Encoder.

Motor Datenblatt

Werte bei Nennspannung		
Nennspannung	12	V
Leerlaufdrehzahl	12400	1/min
Leerlaufstrom	71.7	mA
Nennndrehzahl	10700	1/min
Nennmoment (max. Dauerdrehmoment)	14.6	mNm
Nennstrom (max. Dauerbelastungsstrom)	1.65	A
Anhaltemoment	108	mNm
Anlaufstrom	11.8	A
Max. Wirkungsgrad	85.2	%
Kenndaten		
Max. Abgabeleistung	22.7	W
Anschlusswiderstand	1.02	Ohm
Anschlussinduktivität	0.0576	mH
Drehmomentkonstante	9.18	mNm/A
Drehzahlkonstante	1040	1/min/V
Kennliniensteigung	116	1/min/mNm
Mechanische Anlaufzeitkonstante	6.12	ms
Rotorträgheitsmoment	5.05	gcm ²
Thermische Daten		
Therm. Widerstand Gehäuse-Luft	16	K/W
Therm. Widerstand Wicklung-Gehäuse	7	K/W
Therm. Zeitkonstante der Wicklung	18.1	s
Therm. Zeitkonstante des Motors	528	s
Umgebungstemperatur	-40..100	°C
Max. Wicklungstemperatur	125	°C
Mechanische Daten		
Grenzdrehzahl	18000	1/min
Min. Axialspiel	0	mm
Max. Axialspiel	0.1	mm
Radialspiel	0.02	
Max. axiale Belastung (dynamisch)	2.5	N
Max. axiale Aufpresskraft (statisch)	30	N
Max. radiale Belastung 5 mm ab Flansch	16	N
Weitere Spezifikationen		
Polpaarzahl	1	
Anzahl Kollektorsegmente	9	

Abbildung 57: Auszug aus dem Datenblatt für den DCX 22 S (GB,KL,12V)

Encoder Datenblatt

Typ		
Impulszahl pro Umdrehung	1024	
Anzahl Kanäle	3	
Line Driver	RS422	
Max. Aussendurchmesser	16	mm
Max. Gehäuselänge	8.5	mm
Max. elektrische Drehzahl	90000	1/min
Max. Drehzahl	30000	1/min
Technische Daten		
Versorgungsspannung Typisch	5	V
Versorgungsspannung Toleranz +/-	10	%
Ausgangssignal Treiber	Differential	
Strom pro Kanal	-20...20	mA
Min. Zustandslänge	20	°el
Max. Zustandslänge	160	°el
Signalanstiegszeit/Signalabfallzeit	20/20	ns
Min. Zustandsdauer	125	ns
Drehrichtung	A before B CW	
Indexposition	A low & B low	
Index synchron zu AB	ja	
Indexpulsbreite	90	e°
Typische Stromaufnahme im Stillstand	23	mA
Max. Trägheitsmoment der Impulsscheibe	0.05	gcm ²
Gewicht	0	g
Betriebstemperaturbereich	-40..100	°C
Anzahl Sterilisationszyklen	0	

Abbildung 58: Auszüge aus dem ENX 16 EASY Encoder Datenblatt

13 Referenzen, Glossar

13.1 Abbildungsverzeichnis

Abbildung 1: Die Komponenten des ESCON/EPOS4 Starter Kit.	6
Abbildung 2: Sprachauswahl.....	8
Abbildung 3: Anschliessen des ESCON Starter Kit.	8
Abbildung 4: Aktiver Controller Auswahlmenü.....	9
Abbildung 5: Firmware Versionsvergleich.....	10
Abbildung 6: Erweiterte Optionen.....	17
Abbildung 7: Drehzahlsignale auf ESCON-Datenaufzeichnung.....	19
Abbildung 8: Nach Virtuellen Kontrollern suchen.....	27
Abbildung 9: Standard-Parameter wiederherstellen.....	28
Abbildung 10: Die Komponenten des ESCON/EPOS4 Starter Kit.....	29
Abbildung 11: Anschliessen des EPOS4 Starter Kit.....	31
Abbildung 12: Schematische Übersicht der EPOS4 mit externen Anschlüssen.....	32
Abbildung 13: Die Ablagestruktur der maxon EPOS Software Installation.....	33
Abbildung 14: Die Dokumentationsstruktur der maxon EPOS4.....	34
Abbildung 15: Wo Sie das Symbol für ein neues Projekt finden.....	35
Abbildung 16: New Project Wizard, Schritt 1.....	36
Abbildung 17: New Project Assistent, Schritt 2.....	36
Abbildung 18: Das EPOS Studio mit geöffnetem Workspace Register.....	37
Abbildung 19: Fehler und Warnungen im Statusfenster.....	37
Abbildung 20: Das Communication Register im EPOS Studio.....	38
Abbildung 21: Firmware Update Wizard: Einstellung der Optionen.....	39
Abbildung 22: Firmware Update Assistent: Nach erfolgreichem Firmware Update.....	39
Abbildung 23: Restore All Default Parameters.....	40
Abbildung 24: Doppelklick auf Startup im Wizards Register des EPOS Studio.....	41
Abbildung 25: Startup Wizard, Sicherheitshinweise.....	42
Abbildung 26: Startup Wizard, Schritt 1, Motorparameter.....	42
Abbildung 27: Startup Wizard, Schritt 3, Sensor Parameter.....	43
Abbildung 28: Startup Wizard, Schritt 6, Limitierungen.....	44
Abbildung 29: Die Signale eines Inkremental-Encoders.....	47
Abbildung 30: Regulation Tuning Assistent: Current (Strom).....	50
Abbildung 31: Regulation Tuning Assistent: Position.....	52
Abbildung 32: Regulation Tuning Assistent.....	54
Abbildung 33: PID Regler.....	55
Abbildung 34: Schematische Darstellung des Feed-forward.....	57
Abbildung 35: Master-Slave Architektur mit EPOS4 Slaves.....	58
Abbildung 36: So öffnen Sie den Profile Position Mode.....	59
Abbildung 37: Drehzahlprofile (Drehzahl vs. Zeit) für eine Positionsfahrt.....	60
Abbildung 38: Schema eines Positionsregelungssystems.....	62
Abbildung 39: Standstill window.....	65
Abbildung 40: Konfiguration der digitalen Eingänge für die Homing Übung.....	66
Abbildung 41: Prinzip des Homing mit dem Indexkanal des Encoders.....	68

Abbildung 42: Diagramm des CSP Modus.....	70
Abbildung 43: Position Recording im CSP Modus	71
Abbildung 44: Positionsaufzeichnung im CSP-Modus	71
Abbildung 45: Aufgezeichnete Drehzahlssignale eines DCX	74
Abbildung 46: Aufgezeichnete Drehzahlssignale.....	75
Abbildung 47: Schematischer Aufbau der Betriebsarten	77
Abbildung 48: Das I/O Monitor Werkzeug der EPOS4.....	80
Abbildung 49: Physikalisches Layout des CAN Bus	85
Abbildung 50: CANopen Geräteprofil.....	86
Abbildung 51: Zugriff auf das Object Dictionary der EPOS4.	87
Abbildung 52: Einige Standard CANopen Geräteprofile	88
Abbildung 53: Object Dictionary des Motion Controllers EPOS4.....	89
Abbildung 54: Wo Sie die Refresh Rate ändern können	90
Abbildung 55: Parameter Export/Import Assistent	92
Abbildung 56: EPOS4 50/8 mit CANopen (links) und EtherCAT (rechts) Anschlussplatine ..	94
Abbildung 57: Auszug aus dem Datenblatt für den DCX 22 S (GB,KL,12V)	95
Abbildung 58: Auszüge aus dem ENX 16 EASY Encoder Datenblatt	96

13.2 Liste der Kästchen



ESCON Info

1. Aktuellstes ESCON Studio 7
2. ESCON Studio Sprachauswahl 8
3. Motor- und Sensorparameter der vorliegenden Einheit 12
4. ESCON Auto Tuning 13
5. ESCON Expert und manuelles Tuning 14
6. Werkzeug: Parameter 17
7. DC Tacho Feedback 24



EPOS Info

8. Das neueste EPOS Studio 30
9. Handbücher und Software Dokumentation 33
10. Projekt im EPOS Studio 35
11. Fehler und Warnungen 37
12. Motor- und Sensor-Parameter der vorliegenden Einheit 45
13. Tuning Parameter und manuelles Tuning 53
14. Grüne und rote LED 60
15. Einschränkungen bei Variablentypen von EPOS-Systemen 88
16. Parameter speichern 91
17. PDO und SDO in EPOS Systemen 93



Hintergrund zu Motion Control

18. Positions- und Drehzahlauswertung mit Inkr. Encoder 23
19. Mehr Hintergrund zu Regelkreisen 26
20. Kommutierung mit bürstenbehafteten und bürstenlosen Motoren 46
21. Positions- und Drehzahlauswertung mit Inkr. Encoder 47
22. Encoder-Impulszahl und Regeldynamik 48
23. Feedback und Feed Forward 55
24. Master, Slaves und online Einzelbefehle 58
25. Die Regelkreise im Motion Control 61
26. Schleppfehlerfenster 64
27. Positioniergenauigkeit 65
28. Was ist Homing und warum wird es benötigt? 68
29. Cyclic Synchronous Modus 70
30. Cyclic Synchronous Position (CSP) Mode 71
31. Verständnis von Drehzahlsignalen am Datenrekorder 74
32. Weglaufen oder Durchdrehen im Stromregelungsmodus 78
33. Limit Switches und Home Switch 81
34. Gerätefreigabe 81
35. Touch probe 82
36. Ready/Fault 82
37. Position Compare 82
38. Holding Brake 82

Best Practice

Best Practice

39. ESCON Arbeitsverzeichnis 7
40. ESCON Datenaufzeichnung Anzeige Optionen 20
41. EPOS Arbeitsverzeichnis 30
42. Den USB-Treiber finden 31
43. Diagramm Zoom 51
44. Parameter speichern 83
45. Definieren Sie Ihren eigenen Objektfiler 91

13.3 Literatur

- Feinmess www.feinmess.de/mt_all/glossar.htm
- John K.-H. John, M. Tiegelkamp "SPS-Programmierung mit IEC 61131-3", 3rd Edition, Springer Verlag 2000. ISBN 3-540-66445-9
- Wikipedia www.wikipedia.org
- CAN in Automation www.can-cia.org

13.4 Index

A

Assistent
 Firmware Aktualisierung 9
 Firmware Update 39
 Installationsassistent 30
 Neues Projekt Assistent 35
 Startup Assistent 41
Ausgänge
 Digitale Ausgänge 84

B

Bremse 83
 Haltebremse 83

C

CAN 86
 CANopen 86
 CANopen Gerät 87
 CANopen Geräteprofil 89
 Kommunikation 93
 Process Data Object PDO 94
 Service Data Object SDO 94

D

Datenrekorder
 EPOS4 63
 ESCON 17
Diagnose
 ESCON Diagnose 27
Dokumentation 33
Drehzahlregelung
 Profile Velocity Mode 73
 Velocity Mode 77
Drehzahlsteller
 Open Loop 25

E

Eingänge
 Analoge Eingänge 80
Eingänge/Ausgänge
 ESCON 22
 I/O Monitor 80
Encoder
 Inkremental Encoder 47

Endstufe 23
EPOS Studio
 Assistenten 38
 Register Communication 38
 Register Workspace 37
 Tools 38
ESCON Studio
 Installation 7
 Sprachauswahl 7
EtherCAT 95

F

Fault 83
Feed-Forward 57
Firmware Specification 89
Firmware Update 9, 39
Freigabe
 Disabled 23
 Enabled 23
 ESCON Freigabeoptionen 22
 Gerätefreigabe 100

G

Genauigkeit
 Positioniergenauigkeit 65
Gerätefreigabe 82

H

Handbuch 33, 99
Home Switch 82
Homing 66
 Current threshold 66
 With index channel 68

I

Index Kanal 47
Inkmente 12, 45
Installation
 EPOS Studio 30

K

Kommunikation 32
Kommutierung 46, 100
 Block 46

Sinusoidal 46
Komparator 23
Kontroller Überwachung 16

L

LED
grüne LED 60
rote LED 60
Limit Switch 82, 100

M

Manual
Firmware Specification 33
Programming Reference 33
Master 32, 58, 100
Monitor
I/O Monitor 80
Motor
Bürstenbehaftete (DC) 46
Bürstenlose (BLDC, EC) 46

O

Objektfilter 92
Objektverzeichnis 88
EPOS4 [intern] 90
Index 88

P

Parameter
Parameter speichern 92
Polarity of IOs 84
Position Compare 83
Position Marker 83
Positionierung
Position Mode 69
Profile Position Mode 59

Process Data Object 94

R

Ready 83, 100
Regler-Tuning Assistent 12
Rücksetzen auf Standardparameter 40

S

Schleppfehler
Schleppfehlerfenster 64
Schnellstopp 82
Service Data Object 94
Slave 32, 58
Sollwert
ESCON Sollwert Optionen 23
Startup Assistent 10
Stromregler
Current Mode 79
ESCON Stromregler-Modus 25

T

Tuning 12, 50
Autotuning 13
Expert Tuning 13, 14
Manuelles Tuning 14
Manuelles Tuning 53
Parameter 99

U

USB Treiber 31

V

Verkabelung 8, 31
Virtueller Kontroller 27

