

*Bedienungsanleitung  
für  
Power Drive Controller*

*pdcx85*



Kühlkörper optional

Stand: 37/2009 Änderungen vorbehalten

# **INHALTSVERZEICHNIS**

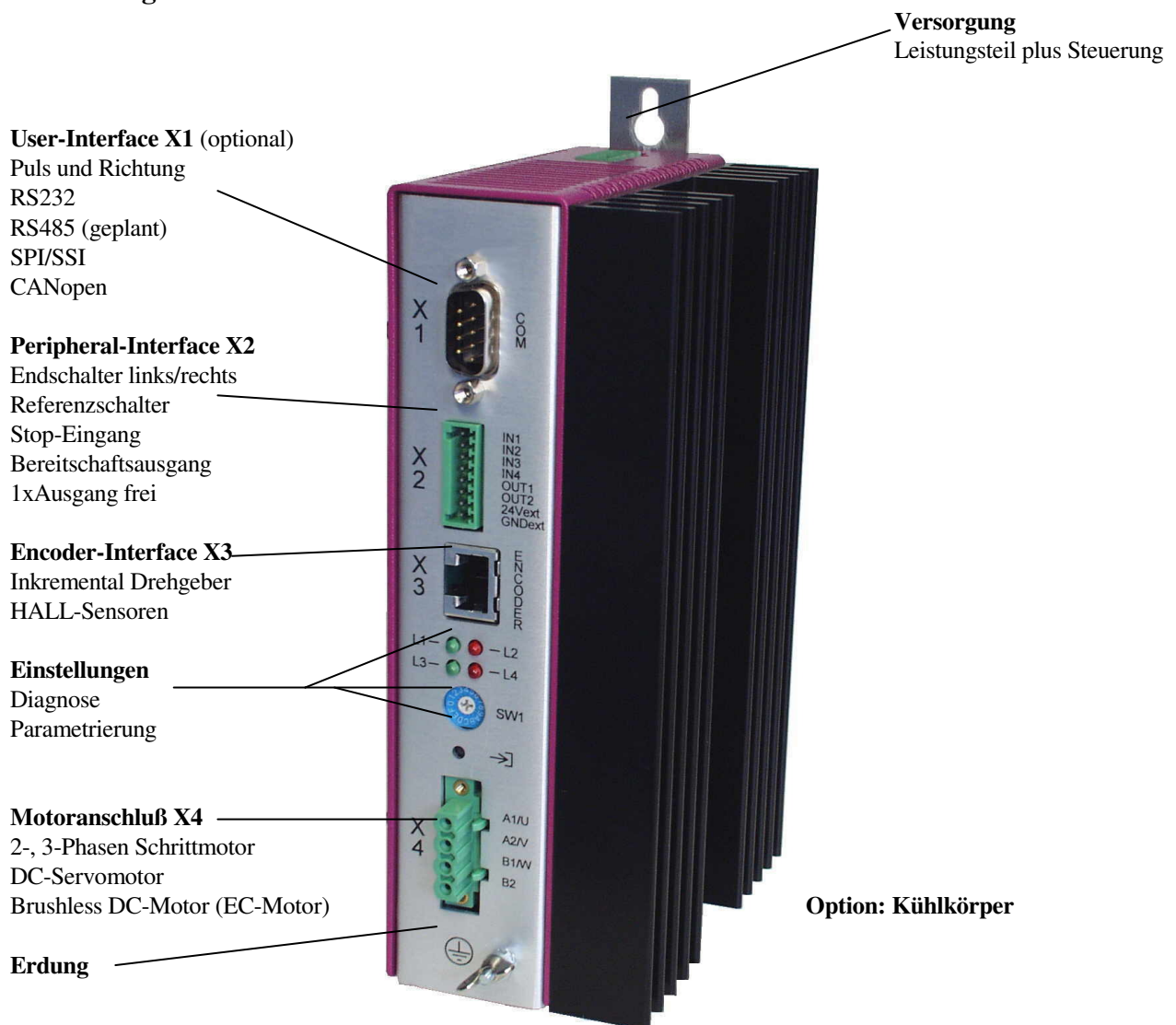
<b>PRODUKTMERKMALE</b>	<b>3</b>
<b>BESTELLSCHLÜSSEL</b>	<b>4</b>
<b>ZUBEHÖR (GETRENNT LIEFERBAR)</b>	<b>5</b>
<b>VERDRAHTUNGSPLAN: (ALLGEMEIN)</b>	<b>6</b>
<b>BEDIENUNG, EINSTELLUNGEN</b>	<b>8</b>
<b>PARAMETERLISTE</b>	<b>9</b>
<b>BETRIEBSART: PULS UND RICHTUNG</b>	<b>12</b>
<b>BETRIEBSART: MULTITURN UND RICHTUNG</b>	<b>13</b>
<b>BETRIEBSART: RS232 SERIELL</b>	<b>15</b>
<b>BETRIEBSART: CANOPEN</b>	<b>19</b>
<b>FEHLERZUSTÄNDE: ALLGEMEIN</b>	<b>26</b>
<b>ABMESSUNGEN</b>	<b>28</b>
<b>TECHNISCHE DATEN</b>	<b>29</b>
<b>ALLGEMEINE INSTALLATIONSANFORDERUNGEN</b>	<b>30</b>
<b>SICHERHEITSHINWEISE / SCHUTZANFORDERUNGEN</b>	<b>30</b>

## Produktmerkmale

- Leistungsteil plus Steuerung
- für 2- oder 3-Phasen Schrittmotoren (Microschritt), DC- oder Bürstenlose-DC (EC) Motoren
- Fahr- und Kontrollfunktionen
- Versorgung: 24...80(130) Volt, bis 5(10)Ampere
- alle Einstellungen an Frontplatte ohne Programmiergerät
- umfangreiche Schutzfunktionen  
Überstrom, Temperatur, Spannung, Fahrüberwachung, ...
- User-Interface (galvanisch getrennt)  
Puls/Richtung, RS232/RS485, SPI, CANopen (optional)
- Peripheral-Interface (galvanisch getrennt)  
2xEndschalter, Referenzschalter, Stopeingang  
Bereitschaftsausgang, 1xAusgang betriebsartabhängig

- Encoder-Interface (galvanisch getrennt)  
Inkremental-Drehgeber, HALL-Sensoren
- LED-Zustandsanzeigen
- alle Anschlüsse mit lösbaren Steckverbinder
- kompaktes Metallgehäuse, Wandmontage, DIN-Schiene

## Anordnung der Interfaces



## Bestellschlüssel

pdcx85.11xxxxx

<b>Power Drive &amp; Controller</b>	_____
<b>Baureihe Motortyp*</b>	-----
0	DC Motor
1	BLDC (EC) Motor
2	2 Phasen Schrittmotor
3	3 Phasen Schrittmotor
<b>Baureihe Nominalspannung**</b>	_____
8	24...80Vdc
<b>Baureihe Nominalstrom</b>	-----
5	1,25...5Ampere
<b>Signalspannung</b>	_____
1	24V Signaleingangsspannung
<b>Gehäusevariante</b>	-----
1	IP20 (mit Haube)
<b>Kühlung</b>	_____
0	kein Kühlkörper
1	mit Kühlkörper
<b>Montage</b>	-----
0	Wandmontage
1	Halteklammer DIN-Schiene
<b>User Interface</b>	_____
0	ohne
1	RS232
2	RS485 (in Vorbereitung)
3	---
4	CANopen SUI (Simple User Interface)
5	PULSE/DIRECTION
<b>Motorspannungsbereich</b>	-----
0	Nominalmotorspannung 24...80Vdc
1	Höhere Motorspannung 60...130Vdc
<b>Motorstrombereich</b>	_____
0	Nominalmotorstrom 1,25...5Ampere
1	Höherer Motorstrom 2,5...10Ampere

## Zubehör (getrennt lieferbar)

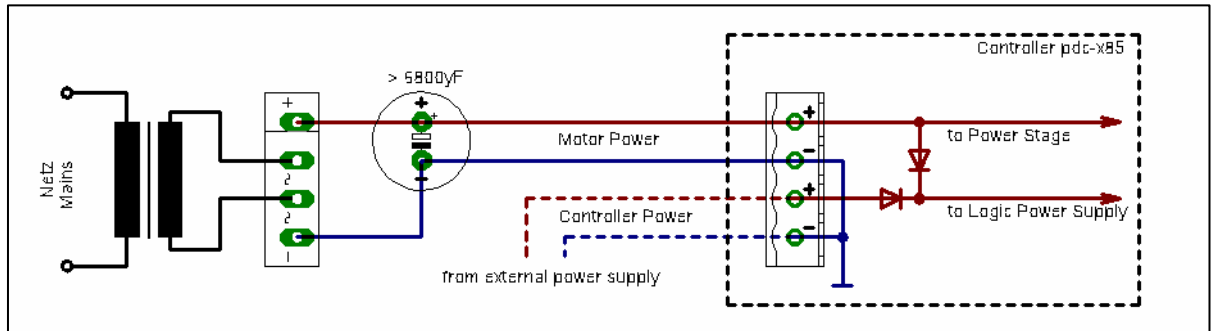
STS.02	Steckersatz für Interface: Versorgungsspannung, X1-User, X2-Peripheral und X4-Motor
STS.03	SUB-D 9 poliger Abschlusswiderstand Buchse/Stecker mit 120 Ohm
HS.00	Kühlkörper (ab 6A Motorphasenstrom empfohlen)
DINH.00	Halteklammer für Gerätemontage auf DIN Schiene („Hutschienenmontage“)
KD.00	Nullmodemkabel 2 Meter Buchse/Buchse SUB D 9polig (! nur in Verbindung mit RS232 User Interface)
KM <i>m</i>	Motorkabel 2x2 0.75mm <sup>2</sup> paarweise verseilt mit Schirmgeflecht <i>m</i> = Länge in Meter 01, 02, 05, 10 (Beispiel 2 Meter Motorkabel = KM02)
KD.03	RJ45 Encoder Cable. 2,5 Meter Lang .
KD.04	RJ45 Encoder Cable. 5 Meter Lang .
KE <i>m</i>	Encoderkabel mit RJ45 Stecker für Interface X3. Zweites Ende offen. <i>m</i> = Länge in Meter 01, 02, 05, 10 (Beispiel 5 Meter Encoderkabel = KE05)
DOKU	DIN-A5 gebundene Dokumentation. Beinhaltet pdc und IOEXT.

\* DC, EC-und 2- (3-)Phasen Schrittmotoren auf Anfrage. Wir beraten gerne bei der Motorauswahl und Antriebsauslegung.

\*\* PFC Netzteil ps400 (400 Watt 80Vdc oder 130Vdc) auf Anfrage

# Verdrahtungsplan: (allgemein)

## Stromversorgung:



In der Regel wird die pdc-x85 über die Motorspannung komplett versorgt. Soll der Motor aus Sicherheitsgründen stromlos geschaltet werden können ohne dass die Steuerung dabei ausfällt so muss der Steuerungsteil getrennt versorgt werden. Dies bietet prinzipiell die Möglichkeit eines „Wiederaufsetzen“ nach einer Sicherheitsabschaltung des Motors.

Es muss sichergestellt sein, dass das Netzteil einen ausreichenden Ladekondensator von mindestens 6800yF aufweist, damit beim Bremsvorgang durch die Rückspeisung der kinetischen Energie die Betriebsspannung nicht über die maximal erlaubte Versorgungsspannung ansteigt.

Zu hohe Motorspannungen können zur Zerstörung der Endstufe führen.

Eine bereits integrierte aktive Ballastschaltung verhindert im grenznahen Bereich Überspannungen in gewissen Grenzen.

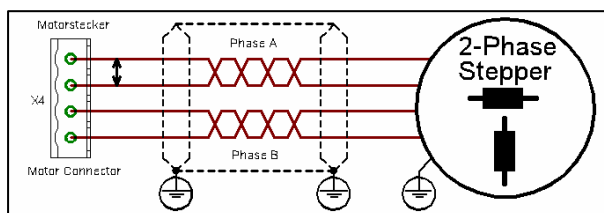
Jedoch ist bei hohem Motorstrom und großer zu treibender Trägheit ein entsprechendes Netzteil vorzusehen, das die maximal zulässige Betriebsspannung nicht übersteigen lässt.

Die Motorversorgung darf keinesfalls schlagartig auf die Endstufe geschaltet werden, da unter Umständen der Ladevorgang der internen Elkos die Sicherung ansprechen lässt. Die Funktion ist garantiert, wenn innerhalb einer viertel Netzperiode (5ms) die volle Betriebsspannung erreicht wird.

Niemals unter Spannung anklemmen, da sonst durch das plötzliche laden der Elkos die internen Sicherungselemente ansprechen können

**! Auf Polung achten**

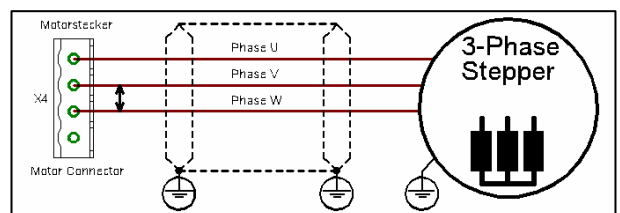
### Motoranschluss: 2-Phasen Schrittmotor X4



Für Richtungswechsel einfach Phase A vertauschen

Achtlitzige Motoren haben pro Wicklung A und B jeweils ein Wicklungspaar. Diese können parallel oder in Serie geschaltet werden. Die Parallelschaltung ist bei höheren Drehzahlen vorteilhaft. Dagegen resultiert die Serienschaltung in ein höheres Drehmoment mit geringem Motorstrom.

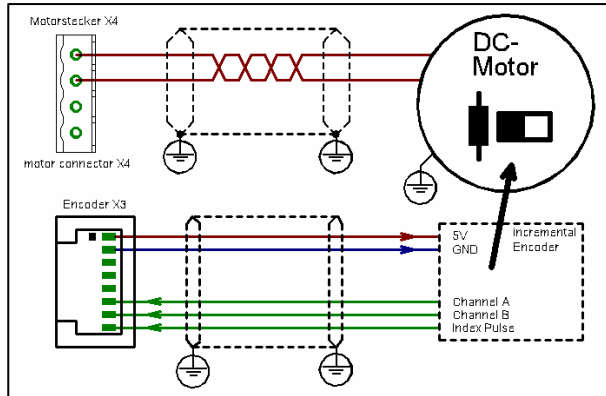
### Motoranschluss: 3-Phasen Schrittmotor X4



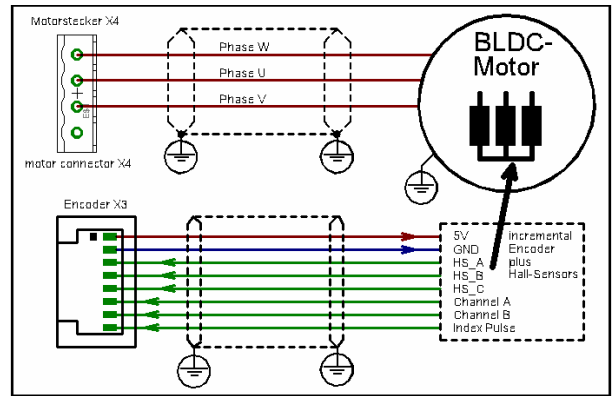
Für Richtungswechsel einfach Phase V-W vertauschen

**Achtung: Niemals Motorstecker unter Spannung lösen oder stecken**

### Motoranschluss: DC-Motor X4

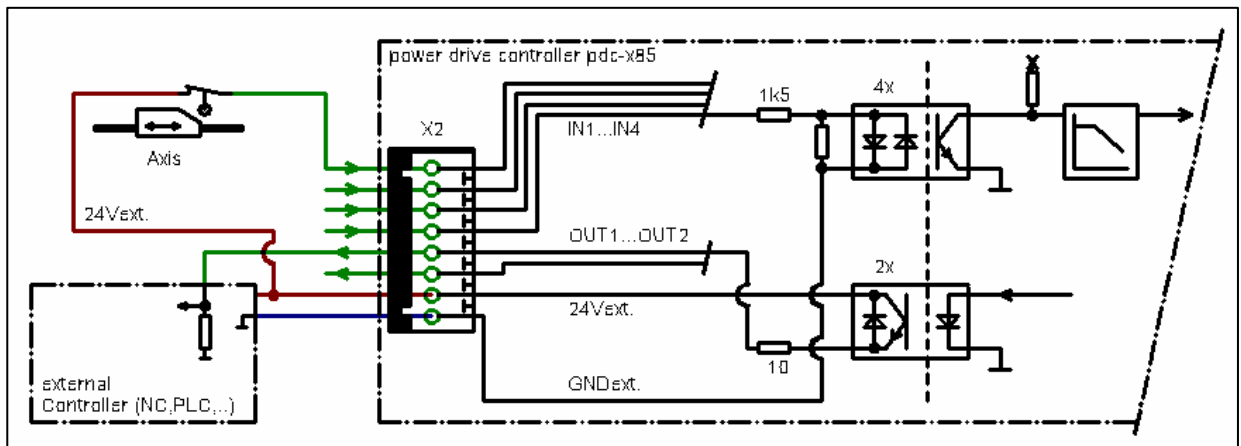


### Motoranschluss: Brushless DC-Motor X4



**Achtung: Niemals Motorstecker unter Spannung lösen oder stecken**

### Anschluss: Peripheral Interface X2



Am Peripheral Interface X2 werden in der Regel die Achssignale wie Endschalter, Referenzschalter und Stop-Eingang angeschlossen. Die Eingänge sind zum Bezugspotential „GNDext“ wechsellspannungsfähig. Das hat den Vorteil, dass man zwischen High aktiven und Low aktiven Eingängen wählen kann.

High aktiv: -> GNDext= low (Normalfall)  
 Low aktiv: -> GNDext= high

Es stehen zwei Ausgänge zur Verfügung, wobei Ausgang „OUT1“ standardmäßig immer mit Bereitschaft der Steuerung belegt ist. Die Ausgänge sind generell High-aktiv. Also bei Steuerung ok ist Bereitschaftsausgang high. Die Versorgung der Ausgänge muss extern durch ein Netzteil oder zur Auswertung durch die Fremdsteuerung bereit gestellt werden.

Alle Signale sind galvanisch getrennt.

## Bedienung, Einstellungen

### Allgemein:

Für den Betrieb einer Steuerung sind Parameter oder Einstellwerte erforderlich, die das Betriebsverhalten der Steuerung bestimmen. Diese können über ein Bedienermenü direkt an der Frontplatte ohne Hilfsgeräte wie z.B. PC usw. eingegeben werden. Diese Werte werden dann im Maschinenparametersatz abgelegt und sind nach Eingabe sofort verfügbar.

Sollen die neuen Parameter beim nächsten Einschalten verwendet werden, müssen sie dauerhaft gespeichert werden. (siehe Parametrierliste Parameter-Code 9E12)

### Parametereingabe über die Frontplatte

Jeder Parameter wird über mindestens 3 Eingabewerte bestimmt. Diese sind Index 1 und Index 2 sowie Parameterwert 1 und Parameterwert 2

- Index 1: bestimmt die Parametergruppe
- Index 2: Subindex für weitere Parameter innerhalb der Parametergruppe (Index 1)
- Parameter 1: Werteingabe als einzige Ziffer oder mehrstellige #Zahl
- Parameter 2: Werteingabe als Ziffer

Diese Methode resultiert in eine sehr hohe Flexibilität mit minimalen Bedienelementen und ist trotzdem im Handling überschaubar. Damit können alle Grundeinstellungen ohne externe Hilfsmittel (PC) eingegeben werden. Theoretisch sind also 256 Parameter einstellbar.

Für die gesamte Bedieneingabe stehen folgende Elemente an der Frontplatte zur Verfügung:

- |                    |              |        |
|--------------------|--------------|--------|
| - 4x LED           | Bezeichnung: | L1...4 |
| - 1x HEX-Schalter  | SW1          |        |
| - 1x Eingabetaster | ->]          |        |

Die Parametereingabe wird überwacht. Bei Fehleingaben wird der Eingabefehler an L1 blinkend angezeigt. Die Eingabe wird verworfen. Der Fehlerzustand muss vor der Neueingabe mit dem Taster zurück gesetzt werden.

### Parametereingabe:

#### Schritt 1:

- o In das Eingabemenü gelangt man durch drücken des Eingabetasters ->] länger als 2 Sekunden
- o darauf hin blinkt L1 zur Erkennung, dass jetzt am SW1 der Index 1 eingestellt werden muss (Index 1)= Parametergruppe
- o Index 1 übernehmen durch ->] => dann blinkt L2

#### Schritt 2:

- o L2 blinkt als Erkennung dass jetzt am SW1 der Index 2 eingestellt werden muss (Index 2)= Subindex von Index 1 für weitere Parameter innerhalb der Parametergruppe (Index 1)
- o Index 2 übernehmen durch ->] => dann blinkt L3

#### Schritt 3:

- o L3 blinkt als Erkennung dass jetzt am SW1 der Parameterwert 1 eingestellt werden muss (Parameterwert 1) repräsentiert jetzt einen Wert, der als einzelne Ziffer oder mehrstellige #Zahl eingegeben wird.
- o bei einzelner Ziffer: Parameterwert 1 übernehmen durch ->] => dann blinkt L4. Weiter mit Schritt 4, auch wenn Parameter 2 nicht benötigt wird.

- o bei einer mehrstelligen #Zahl

Jede Ziffer wird übernommen mit ->]. Als Bestätigung der Übernahme leuchtet L4 kurz auf. Bei einer negativen Zahl muss als erstes Zeichen ‚F‘ eingegeben werden. Die Eingabe der Zahl wird mit den Zeichen ‚A...E‘ beendet. Schritt 4 entfällt. Mehrstellige Zahleneingaben sind mit # markiert.

Eingabebeispiel: -4711

F4711C ,F‘= minus, ,C‘= return

#### Schritt 4:

- o L4 blinkt als Erkennung dass jetzt am SW1 der Parameterwert 2 eingestellt werden muss (Parameterwert 2) repräsentiert ebenfalls einen Wert
- o Parameterwert 2 übernehmen durch ->] => dann ist die Eingabe abgeschlossen, die Menüebene wird verlassen, neue Eingabe siehe Schritt 1

Werden keine oder nur ein Parameterwert benötigt, ist Schritt 3 und 4 trotzdem zu machen (einfach durchklicken, Parameterwerte werden verworfen) Schritt 4 entfällt, wenn Parameter 1 eine mehrstellige #Zahl ist.

Die Defaultwerte sind hinterlegt.

Nicht hinterlegte Werte sind auf jeden Fall einzustellen.

## Parameterliste

Index 1	Index 2	Wert 1	Wert 2	Beschreibung
0	---	---	---	nicht benutzt
1	0	0...2	---	Schritte pro Umdrehung nur bei Schrittmotor <b>! als 1. Parameter einzugeben</b> 1000, 2000, 4000
	1	#0...F	---	Phasenstrom beim Schrittmotor [mA] 1000 #500...10000 Spitzenstrom beim DC-Motor
	2	#0...F	---	Strombegrenzung beim DC-Motor [mA] 6000 #500...10000
	3	0...5	---	Stromabsenkung auf [%] (nur Schrittmotor) 100, 90, 80 70, 60, 50
	4	0...6	---	Drehgeberschritte pro Umdrehung 0, 50, 100, 200, 250, 500, 1000 Schrittmotor: 0= keine Drehüberwachung BLDC-Motor 0= Drehzahlmode nur mit Hall-Sensoren >0 sonst muss externer Geber (mindestens 500 Striche) angemeldet sein
	5	1...4	0...1	Signalinvertierung Wert1 -> IN1...IN4 Wert2 -> 0= nicht, 1= invertiert
	6	#0...F	---	Getriebefaktor: Zähler #1...1000
	7	#0...F	---	Getriebefaktor: Nenner #1...1000
	8	#0...F	---	Maßeinheit: XFER[mm] pro Motorumdrehung #0...100 bei 0 gilt Einheit in Schritten
9	0...3	---	Messeinheit: UNIT 1/1, 1/10, 1/100, 1/1000	
2	0	0...F	---	Lageregler: Kp [%] 0, 10, 20, 30
	1	0...F	---	Ki [%] 40, 50, 60, 70
	2	0...F	---	Kd [%] 80, 90, 100, 110
	3	0...F	---	Auto Tuning (noch nicht) 125, 150, 175, 200
	4	---	---	RESET -> alle auf 100%
	5	0...F	---	<b>Stromregler:</b> Kp [%]
	6	0...F	---	Ki [%]
	7	0...F	---	Auto Tuning (noch nicht)
8	---	---	RESET -> alle auf 100%	
3	0	0...8	---	Baudrate: RS232, RS485 1200, 2400, 4800, 9600, 19200, 38400, 38400, 38400 CANopen 10k, 20k, 50k 100k, 125k 250k, 500k, 800k, 1M
	1	0 1 2	---	Format: 8Bit, no parity, 1 stop 8Bit, even parity, 1 stop 8Bit, odd parity, 1 stop
	2	#0...F	---	Geräteadresse 0 RS232 127 CANopen #0...127
4	0	#0...F	---	Beschleunigung: [%] von Maximalwert 800Hz/ms Step-Motor 20 #1...1000 1kHz/ms DC-Motor
	1		---	Verzögern: [%] (nur bei Drehzahlprofilen) >1000% stoppt sofort 20 #1...1000
	2		---	Startgeschwindigkeit [%] von 200/40kHz bei DC/SM #0...100
	3		---	Endgeschwindigkeit [%] von 200/40kHz bei DC/SM 25 #1...1000
	4		---	Geschwindigkeit: Endschalte suchen [%] von 200/40kHz bei DC/SM 50 #1...1000
	5		---	Geschwindigkeit: Endschalte freifahren [%] von 200/40kHz bei DC/SM 10 #1...100
	6		---	Beschleunigung in Hz/ms für Schrittmotor #1...800 DC-Motor #1...1000
7		---	Endgeschwindigkeit in [Hz] für Schrittmotor #4... 40000 DC-Motor #1000...200000	
5	---	---	---	nicht benutzt
6	0...15	#0...F	---	Geschwindigkeits-Variablen 0...7 [%] von 200/40kHz bei DC/SM #25 1...100
7	0...15	#0...F	---	Positions-Variablen 0...7 #0 +/- 10.000.000
8	0...7	#0...F	---	User-Variablen 0...7 #0 +/- 32.765
9	E	1	2	Maschinendaten speichern

## Spezifische Angaben zu Parameterwerte

### Schrittauflösung

Die Schrittauflösung bestimmt im Wesentlichen das Laufverhalten des Motors. **Vor XFER und UNIT setzen !**

### Laufverhalten:

⊕ weniger als 1000, ⊖ 1000, ⊙ mehr als 1000

### Resonanz-/Positionierverhalten bei Schrittmotoren

Das Resonanzverhalten und somit die Laufkultur des Schrittmotors wird mit zunehmender Schrittauflösung positiv beeinflusst. Allerdings haben Ergebnisse aus der Praxis gezeigt, dass Auflösungen über 4000 Schritte pro Umdrehung hinaus keine nennenswerten Verbesserungen mehr bringen weil

- der Schrittmotor die Schrittgenauigkeit nicht mehr einhalten kann bedingt durch Reibung und andere Kräfte und
- das Resonanzverhalten nicht mehr zu verbessern ist

### Stromeinstellung

Die Motorstromeinstellung bei Schrittmotoren erfolgt als Summenstrom. Dieser ist die geometrische Summe  $I_{\text{Motor}} = \sqrt{(I_a^2 + I_b^2)}$  der beiden Phasenströme  $I_a$  und  $I_b$ .

Grundsätzlich sollte nur soviel Strom wie notwendig eingestellt werden, auch wenn der Nennstrom des Motors höher liegt. Erhöhte Motorströme bringen keine antriebstechnische Verbesserungen, sondern resultieren nur in einer zusätzlichen Verlustleistung.

**Liegt allerdings der Schwerpunkt der Applikation bei einem möglichst geringen Schrittwinkelfehler, so ist die Einstellung des Nennstroms zu empfehlen.**

Bei höheren Schrittfrequenzen kann der eingestellte Strom bedingt durch die Motorinduktivität nicht mehr eingepreßt werden. Drehmomentverluste sind die Folge. (siehe Motorkennlinie der Hersteller)

### Automatische Stromabsenkung

Im Betrieb mit Stillstandszeiten lohnt es sich, die automatische Stromabsenkung zu aktivieren. Dabei wird der Motorstrom auf einen einstellbaren Wert abgesenkt. Die Verlustleistung im Motor sowie im Leistungsteil reduziert sich dabei entsprechend.

Stromabsenkung	0%	auf 60%
Verlustleistung	100%	
Motormoment	100%	

**! Stromabsenkung reduziert das Haltemoment. Es muss sichergestellt sein, dass dieses für die Applikation noch ausreichend ist.**

Die Stromabsenkung wird aktiviert, wenn länger als ca. 2s keine Pulse mehr ankommen

Zum Bestromungsbeginn des Pulseinganges wird wieder der Nennstrom eingestellt. Die Dauer bis zum vollen Nennstrom ist abhängig von Motortyp, und der Motorspannung

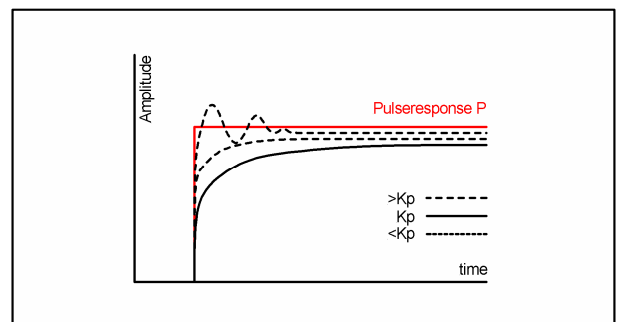
**Ab einem Motorstrom >6A muss die Stromabsenkung auf jeden Fall aktiviert sein**

### Reglereinstellungen:

Veränderungen an den Regelparameter für Lageregler und insbesondere des Stromreglers sind mit großer Sorgfalt durchzuführen.

Die Defaultwerte sind so gewählt, dass im Normalfall ein breites Anwendungsspektrum abgedeckt werden kann. Veränderungen können prozentual innerhalb eines vorgegebenen Bereiches vorgenommen werden.

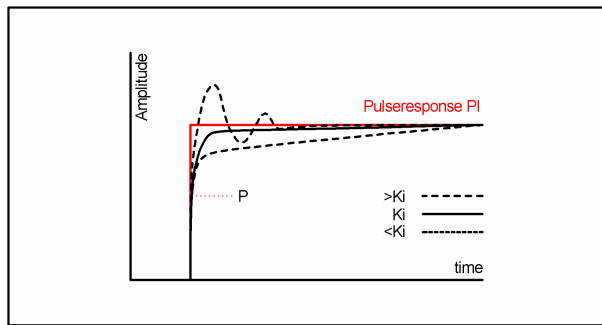
### Verhalten des Verstärkungsfaktors $K_p$ :



Viel  $K_p$  erhöht die Steifigkeit des Systems und resultiert grundsätzlich in einer kleineren Regelabweichung. Wird zuviel  $K_p$  eingestellt, neigt das System zum Klingeln, gemeint ist hierbei eine Schwingneigung höherer Frequenz. Eine optimale  $K_p$ -Einstellung ist etwa so, dass man ein kleiner Überschwinger akzeptiert, der aber schnell ohne wiederholende Oszillationen abklingt.

Wie aus der Grafik zu erkennen ist, hat man beim reinen P-Regler immer eine bleibende Regelabweichung. Diese ist in der Antriebstechnik in der Regel unerwünscht. Deshalb verwendet man immer PI-Regler, dessen Integralanteil die Regelabweichung nach endlicher Zeit null werden lässt. Für den Motorstromregler wird ein PI-Regler verwendet.

### Verhalten des Integralfaktors Ki:

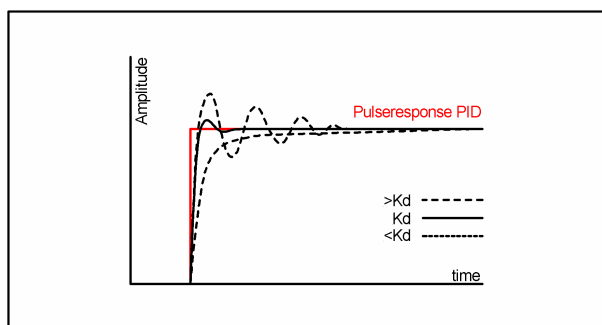


Stellt man einen kleinen Ki-Faktor ein, dauert es länger, bis die Regelabweichung gegen null geht. Dagegen muss man sehr aufpassen, dass der Integralanteil Ki gegenüber dem Proportionalanteil Kp nicht dominiert. Dies resultiert in einem „Winding-Up“ Verhalten, d.h. das System fängt an mit niedriger Frequenz als beim „Klingeln“ dauerhaft zu schwingen. Daher sollte man folgende Einstellregel beachten:

Zunächst maximal mögliches Kp mit wenig oder kein Ki einstellen. Dann soviel Ki wie es die Applikation unbedingt benötigt.

Ein Integralanteil reduziert die Systemdynamik. Um dennoch hohe Dynamik und Systemsteifigkeit zu erhalten, (hohes Kp und Ki) muss ein dämpfendes Element implementiert werden, das die Schwingneigung verhindert. Dazu findet der PID-Regler Anwendung, der als zusätzliches Element den Differentialfaktor Kd enthält. Für den Lageregler wird ein PID-Regler verwendet.

### Verhalten des Differentialfaktors Kd:



Ein zu kleines Kd neigt zum Schwingen, da bei diesem Reglertyp meist hohes Kp und Ki eingestellt werden sollen. Zuviel Kd resultiert in einem ruppigen Laufverhalten und macht sich als „Rauschen“ bemerkbar. Außerdem verhindert die hohe Dämpfung ein schnelles Erreichen der Zielposition.

Wegen der Abhängigkeit von Kp, Ki und Kd ist für die Reglereinstellung eine iterative Vorgehensweise notwendig. Daher ist für die Einstellung eines PID-Reglers folgende Prozedur zu empfehlen:

1. moderates Kd einstellen (Defaultwert)
2. moderates Kp einstellen (Defaultwert)
3. moderates Ki einstellen (Defaultwert)
4. Kp um eine Stufe erhöhen
5. Kd um eine Stufe erhöhen
6. Schritt 4 und 5 wiederholen, solange keine Schwingneigung (Kp) und kein „Rauschen“ (Kd) zu erkennen ist. Schwingt der Antrieb über die Zielposition hinaus, dann Kd gegenüber Kp priorisieren, evtl. Kp reduzieren
7. Integralfaktor Ki soviel wie nötig einstellen.

Ergebnisse aus der Praxis haben gezeigt, dass die Defaultwerte in den meisten Fällen praktikabel sind.

### Drehgeberauswertung:

Schrittmotor:

Drehgeberauflösung= 0: Die Drehüberwachung ist ausgeschaltet

Drehgeberauflösung >0: Die Drehüberwachung ist aktiv. Dabei muss die Pulszahl pro Drehgeberumdrehung eingegeben werden.

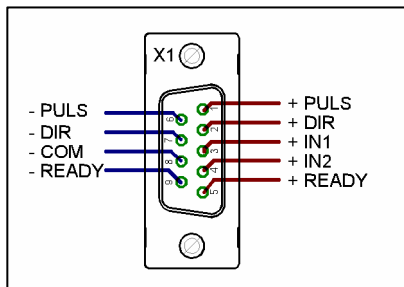
BLDC- oder EC-Motor:

Drehgeberauflösung= 0: Im Drehzahlmode wird die Ist-Drehzahl nur über die Hall-Sensoren erfasst, sonst über externen Drehgeber.

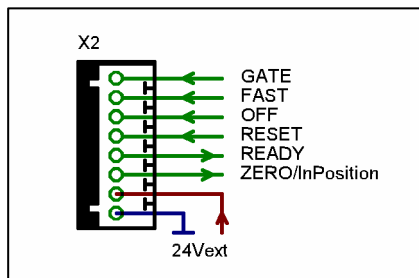
Für alle anderen Betriebsmodi muss immer der externe Drehgeber angemeldet werden. (Parameter 1\_4\_n= >0)

## Betriebsart: Puls und Richtung

### Anschluss: User-Interface



### Anschluss: Peripheral-Interface



## Signalbeschreibung

### PULS:

Mit Beginn des aktiven Signals wird ein Schritt ausgeführt. Das Leistungsteil reagiert nur auf Signalfanken. Bei aktivierter Stromabsenkung und Pulspausen länger als ca. 2s wird der Motorstrom entsprechend abgesenkt.

### DIR: (Direction, Richtung)

Das Richtungssignal bestimmt den Drehsinn des Motors. Das Signal darf nur im Stillstand geändert werden.

### IN1: (nicht verwendet)

### IN2: (nicht verwendet)

### COM: (0 Volt Eingang für IN1, IN2)

### READY: (Bereitschaft)

Dieser Ausgang ist bei ordnungsgemäßer Funktion stromführend. Treten fatale Fehler auf wird der Bereitschaftsausgang stromlos geschaltet. Fatale Fehler können sein Überstrom, Übertemperatur, Schleppfehler zu groß usw.

Dieser Zustand bleibt gespeichert und kann nur mittels des RESET-Eingangs oder durch Drücken des Tasters an der Frontplatte aufgehoben werden.

Das Leistungsteil meldet erst dann Bereitschaft, wenn die Versorgung für ca. 200ms stabil ansteht.

### GATE: (Tor)

Ist der Eingang TOR bestromt, werden alle Pulse von der Endstufe ignoriert. Damit ist es möglich, mehrere Leistungsteile an einer Pulsquelle zu betreiben, wobei sich nur diejenigen Motoren bewegen, die nicht über das TOR-Signal ausgeblendet sind.

### FAST: (Eilgang)

Durch Bestromen des Eingangs wird die Schrittauflösung auf  $\frac{1}{4}$  gesetzt. Dadurch wird die Motorgeschwindigkeit das 4-fache.

Bei FAST ist der Ausgang ZERO nicht mehr garantiert.

### OFF: (Endstufe aus)

Mit Off kann die Endstufe stromlos geschaltet werden. Der Schrittzähler bleibt dabei erhalten.

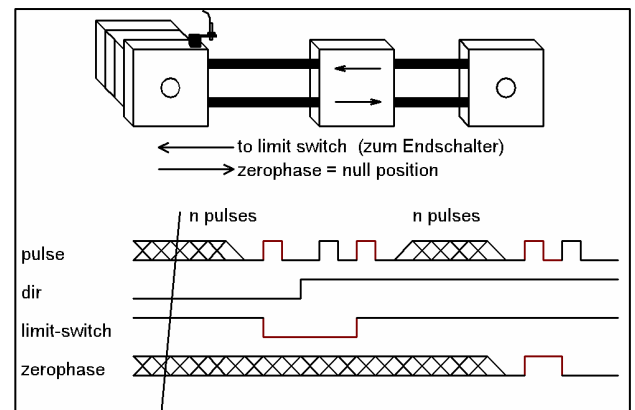
### RESET (Fehler löschen)

Wechsel vom Fehler- in den Betriebszustand. Im Störfall geht das Leistungsteil in den Fehlerzustand über. Ursache hierfür sind zB. zu große Motorströme (Kurzschluss), Übertemperatur usw. Das Leistungsteil schaltet ab, die entsprechende LED zeigt diesen Zustand an, der Bereitschaftsausgang wird stromlos. Mit der Bestromung des RESET-Eingangs wird dieser Zustand aufgehoben. Die Rotorstellung ist undefiniert

### InPosition (nur bei DC-Motoren)

Der Ausgang InPosition wird gesetzt, wenn der Motor die Zielposition erreicht hat.

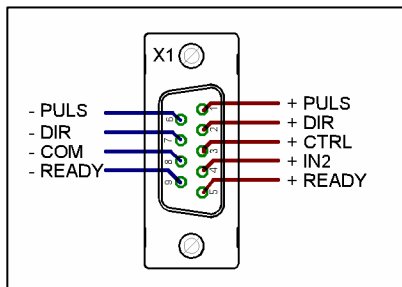
### ZERO: (nur bei Schrittmotor)



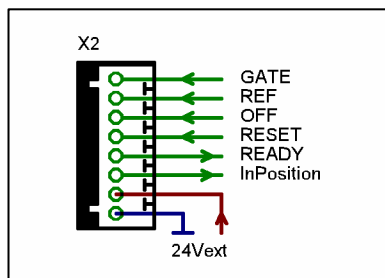
Der Ausgang ZERO kann zum exakten und reproduzierbaren Auffinden des Nullpunktes herangezogen werden. Dazu wird zunächst auf den Endschalter gefahren, dann die Drehrichtung gewechselt und gefahren, bis der ZERO Ausgang schaltet. Dabei muss sichergestellt sein, dass die Schalthysterese des Endschalters außerhalb des ZERO-Signales liegt. (Endschalter justieren)

## Betriebsart: Multiturn und Richtung

### Anschluss: User-Interface



### Anschluss: Peripheral-Interface



### Allgemein

Mit der Betriebsart kann auf ein Startsignal (Puls) hin immer wieder eine bestimmte Strecke oder Umdrehung gefahren werden. Der Fahrweg/Winkel kann über die Beziehung  $1000 \cdot \text{Getriebezähler} / \text{Getriebebenner}$  eingestellt werden. Der Getriebefaktor, Beschleunigung, Endfrequenz usw. lässt sich über die Parametereingabe an der Frontplatte einstellen. (siehe Parameterliste)

Beispiel:

Schrittauflösung: 4000 Schritte pro Umdrehung

Getriebefaktor: Zähler= 2, Nenner= 1

Start(Puls) => Motor dreht  $\frac{1}{2}$  Umdrehung

### Signalbeschreibung

#### PULS (START):

Mit jedem Puls wird der Motor gestartet. Die Fahrfunktion ist dabei abhängig vom Steuereingang CTRL und DIR.

#### DIR: (Direction, Richtung)

Das Richtungssignal bestimmt den Drehsinn des Motors. Das Signal darf nur im Stillstand geändert werden. Die Übernahme erfolgt mit dem PULS-Signal

#### CTRL: Betriebsartwahl

Um eine Achse einzurichten, muss in der Regel zuerst eine Referenzfahrt durchgeführt werden um die Nullposition fest zu legen. Dies erfolgt durch das Bestromen

des CTRL-Eingangs. Mit PULS wird die Referenzfahrt gestartet. Der Motor läuft, bis der Eingang REF am Peripheral-Interface bestromt wird. Diese Position ist dann die Nullposition.

**IN2: (nicht verwendet)**

**COM: (0 Volt Eingang für CTRL, IN2)**

**READY: (Bereitschaft)**

Dieser Ausgang ist bei ordnungsgemäßer Funktion stromführend. Treten fatale Fehler auf wird der Bereitschaftsausgang stromlos geschaltet.

Dieser Zustand bleibt gespeichert und kann nur mittels des RESET-Eingangs oder durch Drücken des Tasters an der Frontplatte aufgehoben werden.

**GATE: (Tor)**

Ist der Eingang TOR bestromt, werden alle Pulse von der Endstufe ignoriert. Damit ist es möglich, mehrere Leistungsteile an einer Pulsquelle zu betreiben, wobei sich nur diejenigen Motoren bewegen, die nicht über das TOR-Signal ausgeblendet sind.

**REF: (Referenzschalter Schließer)**

Anschluß für den Referenzschalter

**OFF: (Endstufe aus)**

Mit Off kann die Endstufe stromlos geschaltet werden. Der Schrittzähler bleibt dabei erhalten.

**RESET (Fehler löschen)**

Wechsel vom Fehler- in den Betriebszustand. Im Störfall geht das Leistungsteil in den Fehlerzustand über. Das Leistungsteil schaltet ab, die entsprechende LED zeigt diesen Zustand an, der Bereitschaftsausgang wird stromlos. Mit der Bestromung des RESET-Eingangs wird dieser Zustand aufgehoben. Die Rotorstellung ist undefiniert

**InPosition**

Der Ausgang InPosition wird gesetzt, wenn der Motor die Zielposition erreicht hat.

Je nach eingestellter Schrittauflösung wird ZERO unter Berücksichtigung des Richtungssignals jedes Vielfache von n Pulsen gesetzt wie nachfolgende Tabelle zeigt.

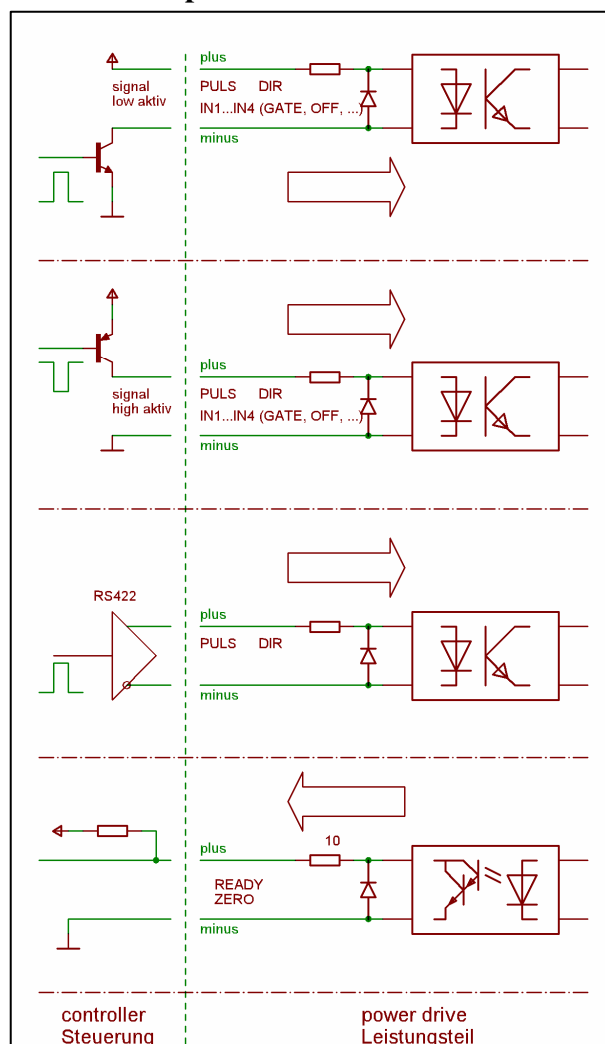
Schritte/Umdr.: ZERO PHASE nach Anzahl Pulse

4000	80
2000	40
1000	20

Die Grundauflösung beträgt generell 4000 Schritte pro Umdrehung

Bei FAST ist der Ausgang ZERO nicht mehr garantiert.

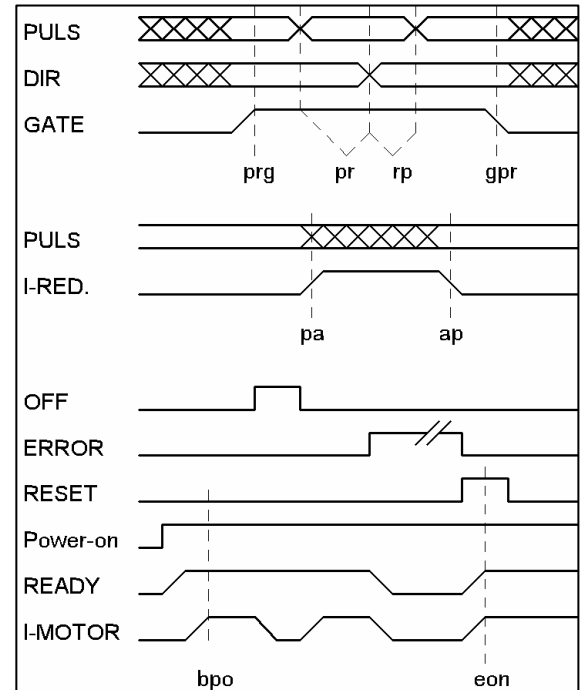
### Ansteuerbeispiele:



Das Signalinterface ist vollkommen mittels Optokoppler galvanisch getrennt. Zur flexiblen Ansteuerung sind bei den Signalen Puls und Richtung jeweils beide Optokopplersignale (plus, minus) herausgeführt.

So ist es einfach möglich, die Endstufe mit high- oder lowaktiven Signalen oder mit Gegentakttreibern (wie RS422) anzusteuern.

### Timing:



! Pulsflanken:	max	2ys
! Pulsdauer:	min	5ys
! Pulspause:	min	5ys

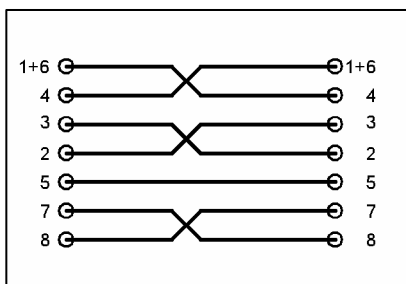
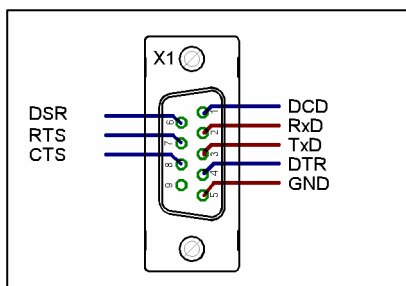
prg:	> 5ys	Gate aktiv nach Puls/Richtung
gpr:	> 1ms	Puls/Richtung aktiv nach Gate
pr:	> 5ys	Puls vor Richtung
rp:	> 5ys	Puls nach Richtung
pa:	< 2s	Stromabsenk. aktiv nach Puls
ap:	< 1ms	Nominalstrom nach Puls
bpo:	< 1s	Bereitschaft nach Power-On
eon:	< 10ms	Bereitschaft nach RESET

### Zustandsanzeige L1...L4:

L1:	Überspannung	Ballastschaltung ist aktiv
L2:	Temperaturwarnung	Lüfter ist an
L3:	bei Schrittmotor:	Stromabsenkung aktiv
	bei DC-Motor:	Motor in Position
L4:	dauernd ein	Run Mode
	langsam blinkend	Stop Mode
Lx	schnell blinkend	Error Mode

## Betriebsart: RS232 Seriell

### Anschluss: User-Interface:



Belegung für 0-Modem Kabel direkt zum PC

Die Signale DCD, DSR und DTR sind intern miteinander verbunden.

Ausgang RTS ist immer aktiv (high, => senden ok)

Mit Eingang CTS kann Datentransfer gestoppt werden.

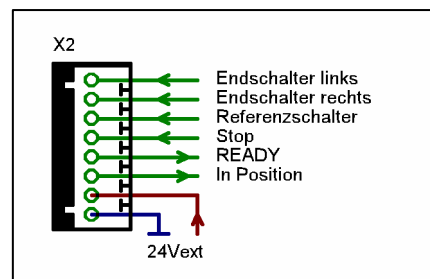
Minimal kabel: RxD, TxD, GND, Schirm

Die empfangenen Zeichen werden zunächst in einem Buffer abgelegt, der genügend groß ist, ein kompletter Kommandostring aufzunehmen. Diese werden sodann entsprechend ihres Eintretens sofort bearbeitet. Die Zeichen LineFeed [LF], Tabulator [HT] und Leerzeichen [SP] werden generell ignoriert.

Mit dem RS232-Interface kann ebenfalls eine einfache BUS-Struktur aufgebaut werden. Dazu wird intern nur der TxD Ausgang aktiv geschaltet, dessen Geräteadresse mit der Kommandoadresse übereinstimmt. Alle Geräte am RS232-BUS sind grundsätzlich auf Empfang, jedoch nur das Gerät führt Kommandos aus, dessen Geräteadresse mit der Kommandoadresse übereinstimmt. Somit kann sehr einfach eine 1 aus n Selektion hergestellt werden.

Weiterhin erlaubt dies grundsätzlich die Möglichkeit, mit nur einem Gateway (relativ teuer) xBUS <-> RS232-BUS mehrere Geräte miteinander zu verbinden und über die 1 aus n Selektion zu betreiben.

### Anschluss: Peripheral-Interface:



## Signalbeschreibung

### Endschalter links/rechts:

Endschalter werden zum Einrichten der Achse benötigt. Dies erfolgt in der Regel beim Einschalten der Maschine durch eine Referenzfahrt, um die Nullposition zu definieren. Endschalter sind in der Regel Öffner. Das bedeutet bei Fahrt auf Endschalter ist der Eingang stromlos. Das Eingangssignal muss dazu invertiert werden. (siehe Parameter 15xx)

### Referenzschalter:

Zur genauesten Nullpunktdefinition kann ein zusätzlicher Referenzschalter verwendet werden. Bei einem Öffner muss das Signal invertiert werden. (siehe Parameter 15xx)

### Stop-Eingang:

Mit dem Stop-Eingang kann die Motorbewegung asynchron von extern abgebrochen werden. Dies wird auch generell als Fehler bewertet.

### READY: (Bereitschaft)

Dieser Ausgang ist bei ordnungsgemäßer Funktion stromführend. Treten fatale Fehler auf wird der Bereitschaftsausgang stromlos geschaltet. Fatale Fehler können sein Überstrom, Übertemperatur, Schleppfehler zu groß usw. Dieser Zustand bleibt gespeichert und kann nur mittels des RESET-Eingangs oder durch Drücken des Tasters an der Frontplatte aufgehoben werden. Das Leistungsteil meldet erst dann Bereitschaft, wenn die Versorgung für ca. 200ms stabil ansteht.

### In Position:

Punkt zu Punkt Mode:

Der Ausgang In Position wird gesetzt, wenn die Achse das Ziel erreicht hat.

Drehzahlprofil Mode:

Der Ausgang In Position wird gesetzt, wenn der Motor die Soll Drehzahl erreicht hat.

### Syntax RS232-Interface:

- der Kommandostring ist im ASCII-Format
- Klein- und Großbuchstaben sind erlaubt
- alle Parameterwerte sind im dezimalen Format
- alle Anweisungen werden mit CR übernommen

Beispiele: ← steht für CR (hex 0D)

richtig	MREL4000← mrel4000← MREL-1234← CLRE←
---------	---

### Betriebsverhalten: RS232-Serial-Mode:

Beim Einschalten kann es sein, dass die übergeordnete Steuerung erst später bereit ist und deshalb Fehler in der Schnittstelle erkannt werden.. Dieser Fehler ist zu löschen mit der Sequenz CR, CLRE und CR.

Fehlerhafte Kommandostrings werden verworfen, der Fehler bleibt jedoch gespeichert und kann nur über das Kommando „CLRE“ oder über den Frontplattentaster gelöscht werden.

### Referenzfahrt:

Pro Richtung werden vier Referenzmodi zur Verfügung gestellt. Fahrt auf Endschalter, zusätzlich auf Referenzschalter oder Indexpuls oder nur auf Referenzschalter. (nur bei Rundtischen)

Zunächst wird in Richtung des Endschaltes gefahren, sofern der Endschalter noch nicht aktiv ist. Beim Erreichen des Endschaltes wird gestoppt und mit umgekehrter Richtung langsam vom Endschalter runter gefahren bis der Endschalter deaktiv wird. (Reversieren) Soll zusätzlich noch auf einen Referenzschalter oder Indexpuls gefahren werden, wird eine entsprechende Suchfahrt angehängt. Die Position am Ende der Referenzfahrt wird zu null erklärt. Danach sollte man unbedingt einen kleinen Offset zum Endschalter einstellen, damit im Betrieb nicht versehentlich der Endschalter reagiert.

### Beispiel:

Kommando:	Kommentar:
MREF0	starte Referenzfahrt links
STAT	Status abfragen x-mal
MREL40	Offset vom Endschalter
PSET0	setze Position null

Die Such- und Reversiergeschwindigkeit der Referenzfahrt wird mit den eingestellten Maschinenparametern „44“ und „45“ durchgeführt. (siehe Parameterliste Seite 9)

Während der Referenzfahrt werden die Endschalter überwacht ob sie nach einer bestimmten Fahrstrecke schalten oder ob die Zuweisung stimmt.

Folgende Fehler werden erfasst:

- falscher Endschalter wurde aktiv
- kein Endschalter erkannt beim Reversieren innerhalb 4000 Schritten
- kein Referenzschalter erkannt beim Reversieren innerhalb 40000 Schritten
- kein Indexpuls erkannt beim Reversieren innerhalb 5000 Schritten.

### Ablauffehler:

Ablauffehler können unter folgenden Umständen auftreten:

- neue Fahr- oder Rampenbefehle während der Fahrt
- Fahrt auf Endschalter
- externer Stop-Eingang wird aktiv
- Schleppfehler oder Unterspannung

Nach dem Fehler zurücksetzen kann die Fahrt mit "MCONT" fortgesetzt werden

### Zustandsanzeige L1...L4:

- |                       |                            |
|-----------------------|----------------------------|
| L1: Überspannung      | Ballastschaltung ist aktiv |
| L2: Temperaturwarnung | Lüfter ist an              |
| L3: bei Schrittmotor: | Stromabsenkung aktiv       |
| bei DC-Motor:         | Motor in Position          |
| L4: dauernd ein       | Run Mode                   |
| langsam blinkend      | Stop Mode                  |
| Lx schnell blinkend   | Error Mode                 |

## Kommandoliste

Bezeichner	Argument Bereich	Einheit	Beschreibung
MABS	-99.999.999...99.999.999 ?	steps oder AW-Einheit	absolute Fahrt Positionsrückmeldung mabs=xxxx CR
MREL	-99.999.999...99.999.999 ?	steps oder AW-Einheit	relative Fahrt (absolute Position bleibt unverändert) Positionsrückmeldung mrel=xxxx CR
MREF	0	---	Referenzfahrt links auf Endschalter
	1	---	Referenzfahrt links auf Endschalter plus Referenzschalter
	2	---	Referenzfahrt links auf Endschalter plus Geber-Indexpuls
	3	---	Referenzfahrt links auf Referenzschalter (bei Rundtische)
	4	---	Referenzfahrt rechts auf Endschalter
	5	---	Referenzfahrt rechts auf Endschalter plus Referenzschalter
	6	---	Referenzfahrt rechts auf Endschalter plus Geber-Indexpuls
7	---	Referenzfahrt rechts auf Referenzschalter (bei Rundtische)	
MCON	---	---	abgebrochene Fahrt fortsetzen
MVEL	-40.000...40.000	steps/sec	Drehzahlprofil bei Schrittmotor
	-200.000...200.000	steps/sec	Drehzahlprofil bei DC/BLDC-Motor mit Drehgeber
	-3.000.....3.000	U/min	Drehzahlprofil bei BLDC-Motor ohne Drehgeber (über Hall-Sensoren)
ACC	1...800 =ACC*STEPS/XFER/1.000	steps/msec <sup>2</sup> AW-Einheit	Beschleunigung Schrittmotor
	1...1.000 =ACC*4.000/XFER	steps/msec <sup>2</sup> AW-Einheit	Beschleunigung DC-Motor
VSTART	1.....16.000 =VSTART*STEPS/XFER	steps/sec AW-Einheit	Startgeschwindigkeit bei Schrittmotor
	1000....20.000 =VSTART*4000/XFER	steps/sec AW-Einheit	Startgeschwindigkeit bei DC-Motor
VEND	1...40.000 =VEND*STEPS/XFER	steps/sec AW-Einheit	Endgeschwindigkeit bei Schrittmotor
	1000....200.000 =VEND*4.000/XFER	steps/sec AW-Einheit	Endgeschwindigkeit bei DC-Motor
STEPS	1.000, 2.000, 4.000	steps/Umdr.	Schrittauflösung pro Umdrehung <span style="float: right;">nur bei Schrittmotor</span> <b>vor XFER und UNIT setzen !</b>
PLIM	1...99.999.999	steps	maximaler absoluter Positionierbereich
PSET	-99.999.999...99.999.999	steps	aktuelle Position wird zu null definiert
IPHASE	0...10.000	mA	Schrittmotor: Amplitude des Phasenstromes 0 -> OFF DC-Motor: Spitzenstrom
ILIM	1000...10.000	mA	Motorstrombegrenzung wirksam nach 2s, <span style="float: right;">(nur bei DC-Motor)</span>
IRED	10...100	%	Stromabsenkung wirksam nach 2s <span style="float: right;">(nur bei Schrittmotor)</span> bei 100% keine Absenkung
STOP	---	---	Fahrt abrechnen <span style="float: right;">(kann mit MCON fortgesetzt werden)</span>
STAT	---	---	Status anfordern -> Rückmeldung -> kein Fehler: ACK StatusByte CR - StatusByte.Bit0: Position nicht erreicht, <span style="float: right;">wenn 1</span> - StatusByte.Bit1: Referenzfahrt aktiv, <span style="float: right;">wenn 1</span> - StatusByte.Bit2: Drehzahl-Mode: Drehzahl erreicht, <span style="float: right;">wenn 1</span> - StatusByte.Bit6: Stromabsenkung aktiv, <span style="float: right;">wenn 1</span> - StatusByte.Bit7: <span style="float: right;">immer 1</span>  Fehler: NAK FehlerByte CR (Fehler-Nr.: siehe Tabelle)
ENCP	---	---	Drehgeber-Rückmeldung encp=xxxx CR
CLRE	---	---	Fehler löschen
ADR	0...127	---	Geräteadresse -> nur Gerät mit Übereinstimmung mit Maschinen- daten-Adresse ist aktiv Adresse 0 ist default und immer aktiv
GNUM	0...1.000	---	Getriebefaktor: Zähler <b>! nur aktiv, wenn XFER= 0</b>
GDEN	1...1.000	---	Getriebefaktor: Nenner <b>! nur aktiv, wenn XFER= 0</b>
XFER	0...100	---	Maßeinheit [mm] pro Rotor-Umdrehung bei 0: -> es gelten STEPS/Umdrehung
UNIT	1,10,100,1.000	---	Messeinheit: Eingabe reziprok, => 10 entspricht 1/10

## Spezifische Angaben zur Kommandoliste

### Verwendung von XFER und UNIT:

XFER und UNIT sind Formfaktoren, um die Fahrwege nicht in Schritten, sondern in SI-Einheiten eingeben zu können. Sie beziehen sich auf die mit STEPS eingestellte Schrittauflösung von STEPS Schritten pro Umdrehung.

**STEPS muss vor XFER und UNIT definiert sein**

### Beispiel:

Es liegt eine Zahnriemenachse vor mit 70mm Hub pro Motorumdrehung, => XFER= 70

Es soll in Einheiten zu 1/10 mm positioniert werden. Dazu wird UNIT mit dem reziproken Wert der Einheit geladen, also UNIT= 10 (bei 1/100 -> 100)

Nun soll um 30mm verfahren werden. Da als Einheit 1/10mm gewählt wurde, ist folgendes Kommando zu übergeben: MREL300

### Bedingung:

XFER \* UNIT muss kleiner gleich STEPS sein

Ist XFER null, wird keine Formatierung vorgenommen. Dann wird die übergebene Position in Schritten interpretiert und mit dem Getriebefaktor. GNUM / GDEN verrechnet

Für einfache Inbetriebnahme und Test kann ein Konfigurationsfile „pdc\_rs232\_com4.ht“ für die Windows ® Applikation Hyper-Terminal® von der Website [www.baur-motion-control.de](http://www.baur-motion-control.de) heruntergeladen werden. Beim Anklicken dieses Files wird Hyper-Terminal® mit den erforderlichen Parametern zur RS232 Kommunikation mit der PDC Steuerung gestartet.

Achtung: COM4 ist im File „pdc\_rs232\_com4.ht“ als PC Schnittstelle voreingestellt und muss unter Umständen an die jeweiligen PC Hardware (COM Ports) angeglichen werden. Dies kann auf Einfache Weise im Hyper-Terminal® Menüpunkt DATEI / EIGENSCHAFTEN eingestellt werden.

### Verwendung von ADR (Kommandoadresse):

Obwohl eigentlich bei dem RS232 Interface nicht üblich, können mehrere Geräte am BUS betrieben werden. Dazu muss jedes Gerät mit einer von anderen verschiedenen Geräteadresse versehen sein. (siehe Maschinendaten Parametereinstellung) Mit dem Kommando ADR wird eine Adresse übergeben. Jedes Gerät vergleicht nun die empfangene Adresse mit der in den Maschinendaten. Das Gerät mit Übereinstimmung ist aktiv, alle anderen nicht.

Beispiel:

Es sind 5 Geräte miteinander verbunden, die jeweils die entsprechende Adressen 1...5 zugewiesen bekommen haben.

Kommando	Kommentar
ADR1	Gerät 1 ist ab sofort aktiv
MREL40000	Fahrauftrag an Gerät 1
ADR2	Gerät 2 ist ab sofort aktiv
MREL200	Fahrauftrag an Gerät 2
ADR5	Gerät 5 ist ab sofort aktiv
MREL3000	Fahrauftrag an Gerät 5

ADR1	wieder auf Gerät 1 umschalten
STAT	und Status abfragen

..... usw:

### Verwendung von MVEL:

Beim BLDC-Motor ist der Drehzahlmode abhängig vom Drehgeberparameter 1.4.n

Drehgeberauflösung n= 0: Die Drehzahl wird nur über die Hall-Sensoren erfasst. Die Drehzahl wird in U/min angegeben. Drehzahlen erst ab 300 U/min sinnvoll.

Drehgeberauflösung n> 0: Die Drehzahl wird über den externen Drehgeber erfasst. Die Drehzahl wird in Schritte pro Sekunde angegeben.

### Verwendung von MREL? und MABS?:

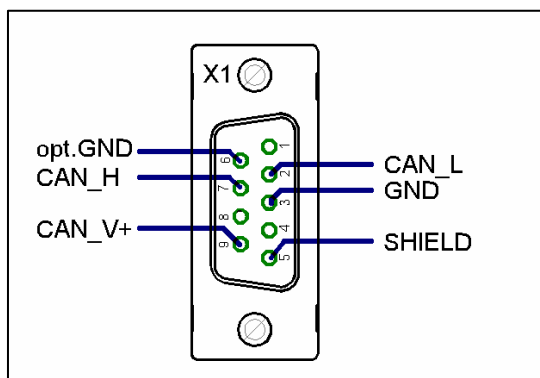
**Relative Fahrt ändert die absolute Position nicht.  
Vor der relativen Fahrt wird Drehgeberzähler null.**

Grundsätzlich gilt, dass die Position immer in dem Mode abgefragt werden soll, indem zuletzt gefahren wurde, also MABS1234 und dann MABS? Allgemein gilt folgender Zusammenhang:

letzte Fahrt	Abfrage	Rückmeldung
relativ	MREL?	relative Position
relativ	MABS?	letzte absolute Position
absolut	MREL?	aktuelle absolute Position
absolut	MABS?	aktuelle absolute Position

## Betriebsart: CANopen

### Anschluss: User-Interface:



Die Signale GND und opt.GND sind intern verbunden.

**Leitungsenden müssen mit einem Widerstand von 120 Ohm abgeschlossen werden (über Zubehör erhältlich)**

**Die pdc-x85 ist ein CANopen Slave Gerät  
Objektverzeichnis und EDS im Internet verfügbar**

Mit dem CANopen kann die pdc-x85 sehr einfach durch einen CANopen Master betrieben werden. Besonderes Augenmerk wurde auf die Verwendung mit nur wenigen Objekten gelegt. Dies wurde erreicht, indem die pdc-x85 durch eine Befehlsstruktur mit Kommando und dazu gehörigen Parametern gesteuert wird, analog wie bei einem RS232 Interface üblich. Dadurch wird der Parametrierungsaufwand beträchtlich reduziert und somit auch die Inbetriebnahmezeiten. Weiterhin bleibt das Objektverzeichnis bei eventuellen Befehlsweiterungen unverändert und muss deshalb nicht immer neu angepasst werden.

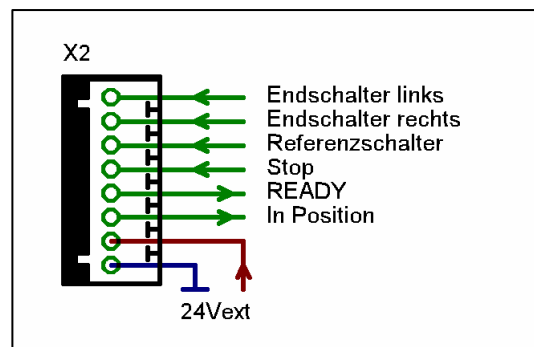
CANopen kommt mit den in DS301 definierten standardisierten vorgelegten Kommunikationsobjekten [predefined connection set] aus.

Folgende Kommunikationsobjekte stehen zur Verfügung:

COB	Node-ID	COB-IDs
NMT	---	0
SYNC	---	80 <sub>h</sub>
TIME	---	100 <sub>h</sub>
EMCY	1...127	081 <sub>h</sub> ...0FF <sub>h</sub>
PDO1 (tx)	1...127	181 <sub>h</sub> ...1FF <sub>h</sub>
PDO1 (rx)	1...127	201 <sub>h</sub> ...27F <sub>h</sub>
PDO2 (tx)	1...127	281 <sub>h</sub> ...2FF <sub>h</sub>
PDO2 (rx)	1...127	301 <sub>h</sub> ...37F <sub>h</sub>
SDO (tx)	1...127	581 <sub>h</sub> ...5FF <sub>h</sub>
SDO (rx)	1...127	601 <sub>h</sub> ...67F <sub>h</sub>
NMT (error)	1...127	701 <sub>h</sub> ...77F <sub>h</sub>

nicht benutzt

### Anschluss: Peripheral-Interface:



### Signalbeschreibung

#### Endschalter links/rechts:

Endschalter werden zum Einrichten der Achse benötigt. Dies erfolgt in der Regel beim Einschalten der Maschine durch eine Referenzfahrt, um die Nullposition zu definieren. Endschalter sind in der Regel Öffner. Das bedeutet bei Fahrt auf Endschalter ist der Eingang stromlos. Das Eingangssignal muss dazu invertiert werden. (siehe Parameter 15xx)

#### Referenzschalter:

Zur genauesten Nullpunktdefinition kann ein zusätzlicher Referenzschalter verwendet. Bei einem Öffner muss das Signal invertiert werden. (siehe Parameter 15xx)

#### Stop-Eingang:

Mit dem Stop-Eingang kann die Motorbewegung asynchron von extern abgebrochen werden. Dies wird auch generell als Fehler bewertet.

#### READY: (Bereitschaft)

Dieser Ausgang ist bei ordnungsgemäßer Funktion stromführend. Treten fatale Fehler auf wird der Bereitschaftsausgang stromlos geschaltet. Fatale Fehler können sein Überstrom, Übertemperatur, Schleppfehler zu groß usw. Dieser Zustand bleibt gespeichert und kann nur mittels des RESET-Eingangs oder durch Drücken des Tasters an der Frontplatte aufgehoben werden. Das Leistungsteil meldet erst dann Bereitschaft, wenn die Versorgung für ca. 200ms stabil ansteht.

#### In Position:

Punkt zu Punkt Mode:

Der Ausgang In Position wird gesetzt, wenn die Achse das Ziel erreicht hat.

Drehzahlprofil Mode:

Der Ausgang In Position wird gesetzt, wenn der Motor die Sollzahl erreicht hat.

## Objekte von pdc-x85 mit CANopen

Zur Steuerung sind nur 8 Objekte relevant wie folgt:

- CAN-ErrorRegister [Index 1001]
- CAN-Control [Index 2000]
- CAN-Status [Index 2001]
- CAN-Command [Index 2002]
- CAN-Argument [Index 2003]
- ERROR-ID [Index 2004]
- CAN-DataId [Index 2005]
- CAN-DeviceData [Index 2006]
- CAN-UserExtin [Index 2007]

<b>CAN-Control</b> [Index 2000]		15	9	8	7	6	0
			Halt	Fehler Reset			

Bei CAN-Control sind nur die Bits 7 und 8 belegt und haben folgende Bedeutung:

Fehler Reset:

Befindet sich die pdc-x85 im Fehlerzustand (signalisiert im CAN-Status Word oder im CAN-ErrorRegister), kann der Fehlerzustand durch setzen des Bits „Fehler Reset“ zurück gesetzt werden. Dies kann auch über das Kommando Nr.16(hex) erfolgen.

Halt:

Durch setzen des Bits Halt kann eine Fahrt abgebrochen werden. Dies kann auch über das Kommando Nr. 03(hex) erfolgen.

! Positionierungen bei HALT aktiv sind nicht möglich.

<b>CAN-Status</b> [Index 2001]		15...11	10	9...6	5	4	3	2...0
x	Ziel erreicht	x	Halt	x	Error			x

Die mit x gekennzeichneten Bits sind nicht definiert und können beliebige Zustände haben.

Das StatusWord wird bei Änderung automatisch mit dem Kommunikationsobjekt PDO1(tx) gesendet.

Ziel erreicht:

Wird gesetzt, wenn der Antrieb das Ziel erreicht hat.

Halt:

Wenn gesetzt, wird der Antrieb gerade verzögert (Ziel erreicht= 0) oder er steht (Ziel erreicht= 1)

! HALT muss explizit über das CAN-ControlWord zurückgesetzt werden

Error:

Zeigt einen allgemeinen Fehler an, wenn 1. Kann durch „Fehler Reset“ im CAN-ControlWord oder durch Kommando gelöscht werden.

**CAN-Command** [Index 2002]

**CAN-Argument** [Index 2003]

Mit dem Objekt CAN-Command wird der eigentliche Befehl an die pdc-x85 spezifiziert. Je nach Befehl ist auch ein Parameter erforderlich, der in CAN-Argument mitübertragen wird. Die Objekte werden im PDO2(rx) empfangen. Eine Befehlsübersicht können Sie aus der Tabelle „Kommandoliste“ in diesem Kapitel entnehmen.

**ERROR-ID** [Index 2004]

Das Objekt ERROR-ID beinhaltet eine Hersteller spezifische Fehlercodierung.

Im höher wertigen Byte ist die Fehlergruppe 1 bis 4 abgelegt, wobei mit steigender Zahl der Fehler schwerwiegender ist. Fehlergruppe 1 ist eher als Warnung zu interpretieren, dagegen ist Fehlergruppe 4 als fataler Fehler zu werten, der zum Abschalten des Motors führt.

Im nieder wertigen Byte ist eine Fehlernummer abgelegt, die den Fehler genauer spezifiziert. Eine Tabelle der Fehlernummern und deren Bedeutung ist weiter hinten im Kapitel Fehlerzustände zu finden.

Das Objekt ERROR-ID kann über das SDO0 [Index 2004] gelesen werden.

**CAN-DataId** [Index 2005]

**CAN-DeviceData** [Index 2006]

Daten, die an den Master nach Aufforderung zurück gesendet werden, sind durch den CAN-DataId eindeutig zugeordnet. Die Daten selbst sind in CAN-DeviceData. Die Struktur ist analog zu CAN-Command und CAN-Argument aufgebaut. Somit ist eine sehr flexible skalierbare Datenkommunikation möglich.

**CAN-UserExtin** [Index 2007]

Reserviert für zukünftige Erweiterungen.

**CAN-ErrorRegister** [Index 1001]

7	6.....1	0
Hersteller spezifisch	x	Fehler allgemein

Die mit x gekennzeichneten Bits sind nicht definiert und können beliebige Zustände haben.

Das Bit „Fehler allgemein“ wird unabhängig vom aufgetretenen Fehler als globale Fehlererkennung gesetzt. Bei einem geräteinternen herstellerspezifischem Fehler wird zusätzlich Bit 7 gesetzt.

Die Fehlerflags können nur durch das Bit „FehlerReset“ im CAN-ControlWord oder durch das Kommando Nr.16(hex) gelöscht werden

## Mapping der Objekte mit CANopen

Bei CANopen werden Prozessdaten in der Regel mit PDOs (Process Data Objects) übertragen. Dabei wird aus Sicht der pdc-x85 zwischen Sende- [PDO(tx)] und Empfangs-PDOs [PDO(rx)] unterschieden. Ein PDO kann bis zu maximal 8 Byte Nutzdaten haben. Über das Mapping wird nun festgelegt, an welcher Bytestelle welche Nutzdaten abgelegt sind. Die Übertragung erfolgt immer im Little Endian Format, d.h. bei einem Wert mit mehreren Bytes Länge wird das LSB (niederwertigste Byte) zuerst übertragen.

### PDO-Mapping von pdc-x85

2 Byte			
PDO1 (rx)	CAN-Control		
2 Byte      2 Byte			
PDO1 (tx)	CAN-Status	CAN-UserExtin	
2 Byte      1 Byte      4 Byte			
PDO2 (rx)	CAN-Control	CAN-Command	CAN-Argument
2 Byte      1 Byte      4 Byte			
PDO2 (tx)	CAN-Status	CAN-DataId	CAN-DeviceData

## Kommunikationsobjekte bei CANopen

Vereinbarung im Folgenden:

- alle Zahlenwerte sind im Hexformat dargestellt
- für Beispiele ist die Node-ID 20 hex (32 dez)
- aus Sicht von pdc-x85    Empfang    Senden

### NMT-Dienste zur Gerätekontrolle

COB-ID	Kommando	Node-ID
--------	----------	---------

#### COB-ID:

- für Gerätedienste immer 0, damit höchste Priorität

#### Kommando:

NR.	Kommando	Zustandswechsel nach
- 01	Start Remote Node	operational
- 02	Stop Remote Node	Stop communication
- 80	Enter Pre-operational	pre-operational

#### Node-ID

- 0	Broadcast, an alle Slaves gleichzeitig
- n	an Slave mit Node-ID (Adresse) n      z.B. 20 <sub>h</sub>

## NMT-Dienste Rückmeldung

COB-ID	State
--------	-------

#### COB-ID:

- 700<sub>h</sub> + Node-ID      hier 720<sub>h</sub>

State:	Bedeutung:
- 0	Boot up
- 4	stopped
- 5	operational
- 7F <sub>h</sub>	pre-operational

Die pdc-x85 sendet ca. alle Sekunde einen Heartbeat, der den CANopen State der Kommunikation meldet.

### Beispiel einer Einschaltsequenz für pdc-x85

pdc-x85 wird eingeschalten

pdc-x85 meldet „Boot up“

COB-ID	State
720 <sub>h</sub>	0

und dann

pdc-x85 Heartbeat meldet „pre-operational“

COB-ID	State
720 <sub>h</sub>	7F <sub>h</sub>

In diesem State können SDOs zur Geräteparametrierung verwendet werden

Master sendet Kommando “Start Remote Node”

COB-ID	Kommando	Node-ID
0	01	20 <sub>h</sub>

pdc-x85 Heartbeat meldet „operational“

COB-ID	State
720 <sub>h</sub>	5

Die pdc-x85 ist nun voll über CANopen steuerbar.

## SDO Kommunikationsobjekte

Datenrichtung: SDO-Client zum pdc-x85

COB-ID	CCD	Index	SubIndex	Daten
620 <sub>h</sub>	x	x	x	x
	1	2	1	4 Byte

### CCD: Kommando Code

- 23 <sub>h</sub>	4 Byte schreiben	SDO Schreib Anforderung
- 27 <sub>h</sub>	3 Byte schreiben	
- 2B <sub>h</sub>	2 Byte schreiben	
- 2F <sub>h</sub>	1 Byte schreiben	
- 40 <sub>h</sub>		SDO Lese Anforderung

Datenrichtung: pdc-x85 zum SDO-Client

COB-ID	CCD	Index	SubIndex	Daten
5A0 <sub>h</sub>	x	x	x	x
	1	2	1	4 Byte

### CCD: Kommando Code

- 60 <sub>h</sub>	schreiben erfolgreich	SDO Schreib Antwort
- 80 <sub>h</sub>	Fehler	
- 43 <sub>h</sub>	4 Byte lesen	SDO Lese Antwort
- 47 <sub>h</sub>	3 Byte lesen	
- 4B <sub>h</sub>	2 Byte lesen	
- 4F <sub>h</sub>	1 Byte lesen	

**Beispiel:** ERROR-ID [Index 2004] lesen  
Leseanforderung Index 2004

COB-ID	CCD	Index	SubIndex
620 <sub>h</sub>	40 <sub>h</sub>	0420 <sub>h</sub>	00

Antwort auf Leseanforderung

COB-ID	CCD	Index	SubIndex	Daten
5A0 <sub>h</sub>	4B <sub>h</sub>	0420 <sub>h</sub>	00	Wert

**Beispiel:** CAN-Control [Index 2000] schreiben  
Wert 4711<sub>h</sub> schreiben auf Index 2000

COB-ID	CCD	Index	SubIndex	Daten
620 <sub>h</sub>	2B <sub>h</sub>	0020 <sub>h</sub>	00	1147 <sub>h</sub>

Antwort nach Schreibzugriff

COB-ID	CCD	Index	SubIndex	Daten
5A0 <sub>h</sub>	60 <sub>h</sub>	0020 <sub>h</sub>	00	---

## PDO Kommunikationsobjekte

Datenrichtung: PDO-Client zum pdc-x85

COB-ID	Control	Command	Argument
320 <sub>h</sub>	x	x	x
	2	1	4 Byte

Über PDO2(rx) wird die pdc-x85 komplett gesteuert. CAN-Control [Index 2000], CAN-Command [Index 2002] und CAN-Argument [Index 2003] sind nach obiger Reihenfolge gemappt.

Über PDO1(tx) wird der Status der pdc-x85 automatisch bei jeder Änderung im CAN-Status gemeldet. Somit ist der CAN-Bus im Gegensatz zum Polling via dem Master nur bei Änderungen im CAN-Status Register zusätzlich belastet.

Die Liste der Kommandos und deren Bedeutung stehen in der Kommandoliste in diesem Kapitel.

! Das Kommando wird unmittelbar ausgeführt

Datenrichtung: pdc-x85 zum PDO-Client

COB-ID	Status	DataId	Daten
2A0 <sub>h</sub>	x	x	x
	2	1	4 Byte

Über PDO2(tx) sendet die pdc-x85 Daten zurück an den Master. Das StatusWord [Index 2001] wird dabei immer mitgesendet. Die Zugehörigkeit der Daten wird über den Daten-Identifikator DataID [Index 2005] eindeutig bestimmt. Die Daten selbst sind in DeviceData [Index 2006] gespeichert. Die Struktur ist gleich wie bei PDO2(rx).

Dieses Konzept erlaubt eine sehr hohe Flexibilität und Skalierung der Kommunikation bei geringstem Aufwand im Mapping der Anwenderdaten.

**Beispiel:** Kommandos an pdc-x85

COB-ID	Control	Command	Argument
--------	---------	---------	----------

Beschleunigung 1000<sub>d</sub> für Referenzfahrt setzen

320 <sub>h</sub>	0 <sub>h</sub>	09 <sub>h</sub>	3E8 <sub>h</sub>
------------------	----------------	-----------------	------------------

Startgeschwindigkeit 1<sub>d</sub> für Referenzfahrt setzen

320 <sub>h</sub>	0 <sub>h</sub>	0A <sub>h</sub>	1 <sub>h</sub>
------------------	----------------	-----------------	----------------

Hinweis: Referenzschalter-Suchfahrgeschwindigkeit und Freifahrgeschwindigkeit vom Referenzschalter werden mit den Maschinenparametern 44 und 45 eingestellt.

Referenzfahrt auf Endschalter links starten

320 <sub>h</sub>	0 <sub>h</sub>	05 <sub>h</sub>	0 <sub>h</sub>
------------------	----------------	-----------------	----------------

warten, bis pdc-x85 Positionierende im CAN-Status (Bit10) über PDO1(tx) meldet. (erfolgt automatisch bei Änderung im CAN-Status Register)

COB-ID	CAN-Status	CAN-UserExtin
1A0 <sub>h</sub>	0400 <sub>h</sub>	0000 <sub>h</sub>

Endgeschwindigkeit 5000<sub>d</sub> für Positionierung setzen

320 <sub>h</sub>	0 <sub>h</sub>	0B <sub>h</sub>	1388 <sub>h</sub>
------------------	----------------	-----------------	-------------------

um 100<sub>d</sub> Inkremente vom Endschalter freifahren

320 <sub>h</sub>	0 <sub>h</sub>	02 <sub>h</sub>	64 <sub>h</sub>
------------------	----------------	-----------------	-----------------

warten, bis Ziel erreicht

COB-ID	CAN-Status	CAN-UserExtin
1A0 <sub>h</sub>	0400 <sub>h</sub>	0000 <sub>h</sub>

diese Position als Null markieren

320 <sub>h</sub>	0 <sub>h</sub>	07 <sub>h</sub>	---
------------------	----------------	-----------------	-----

fahre absolut auf Position 4000<sub>d</sub>

320 <sub>h</sub>	0 <sub>h</sub>	01 <sub>h</sub>	FA0 <sub>h</sub>
------------------	----------------	-----------------	------------------

...  
usw

Dezimalwert (d) Hexwert (h)

**Beispiel:** Position an Master melden

COB-ID	Control	Command	Argument
--------	---------	---------	----------

Melde aktuelle Position zurück

320 <sub>h</sub>	0	17 <sub>h</sub>	---
------------------	---	-----------------	-----

war (ist) die letzte Fahrt absolut, wird PDO2(tx) wie folgt gesendet:

Datenrichtung: pdc-x85 zum PDO-Client

COB-ID	Status	DataId	DeviceData
2A0 <sub>h</sub>	x	1 <sub>h</sub>	x <sub>h</sub>
	2	1	4 Byte

Bei DatenId= 1 sind die gesendeten Daten im absoluten Maß zu interpretieren

oder

war (ist) die letzte Fahrt , wird PDO2(tx) wie folgt gesendet:

Datenrichtung: pdc-x85 zum PDO-Client

COB-ID	Status	DataId	DeviceData
2A0 <sub>h</sub>	x	2 <sub>h</sub>	x <sub>h</sub>
	2	1	4 Byte

Bei DatenId= 2 sind die gesendeten Daten im relativen Maß zu interpretieren

## Kommandoliste für CANopen

Kom.Nr.	Argument Bereich [dec]	Einheit	Beschreibung
01 <sub>h</sub>	-99.999.999...99.999.999	steps	absolute Fahrt
02 <sub>h</sub>	-99.999.999...99.999.999	steps	relative Fahrt (absolute Position bleibt unverändert)
03 <sub>h</sub>	---	---	Fahrt abbrechen (kann mit 04 <sub>h</sub> fortgesetzt werden)
04 <sub>h</sub>	---	---	abgebrochene Fahrt fortsetzen
05 <sub>h</sub>	0	---	Referenzfahrt links auf Endschalter
	1	---	Referenzfahrt links auf Endschalter plus Referenzschalter
	2	---	Referenzfahrt links auf Endschalter plus Geber-Indexpuls
	3	---	Referenzfahrt links auf Referenzschalter (bei Rundtische)
	4	---	Referenzfahrt rechts auf Endschalter
	5	---	Referenzfahrt rechts auf Endschalter plus Referenzschalter
	6	---	Referenzfahrt rechts auf Endschalter plus Geber-Indexpuls
7	---	Referenzfahrt rechts auf Referenzschalter (bei Rundtische)	
06 <sub>h</sub>	1...99.999.999	steps	maximaler absoluter Positionierbereich
07 <sub>h</sub>	-99.999.999...99.999.999	steps	Position setzen
08 <sub>h</sub>	-40.000...40.000	steps/sec	Drehzahlprofil bei Schrittmotor
	-200.000...200.000	steps/sec	Drehzahlprofil bei DC-Motor
09 <sub>h</sub>	1.....800	steps/msec <sup>2</sup>	Beschleunigung Schrittmotor
	1.....1.000	steps/msec <sup>2</sup>	Beschleunigung DC-Motor
0A <sub>h</sub>	1.....16.000	steps/sec	Startgeschwindigkeit bei Schrittmotor
	1.....20.000	steps/sec	Startgeschwindigkeit bei DC-Motor
0B <sub>h</sub>	1.....40.000	steps/sec	Endgeschwindigkeit bei Schrittmotor
	1....200.000	steps/sec	Endgeschwindigkeit bei DC-Motor
0C <sub>h</sub>	1.000, 2.000, 4.000	steps/Umdr.	Schrittauflösung pro Umdrehung (nur Schrittmotor)
0D <sub>h</sub>	0...10.000	mA	Schrittmotor: Amplitude des Phasenstromes 0 -> OFF
			DC-Motor: Spitzenstrom
0E <sub>h</sub>	10...100	%	Stromabsenkung wirksam nach 2s (nur bei Schrittmotor) bei 100% keine Absenkung
0F <sub>h</sub>	1.000...10.000	mA	Motorstrombegrenzung wirksam nach 2s, (nur bei DC-Motor)
10 <sub>h</sub>	---	---	nicht verwendet
11 <sub>h</sub>	0...1.000	---	Getriebefaktor: Zähler ! nur aktiv, wenn XFER= 0
12 <sub>h</sub>	1...1.000	---	Getriebefaktor: Nenner ! nur aktiv, wenn XFER= 0
13 <sub>h</sub>	0...100	---	Maßeinheit XFER [mm] pro Rotor-Umdrehung bei 0: -> es gelten STEPS/Umdrehung
14 <sub>h</sub>	1,10,100,1.000	---	Messeinheit: UNIT Eingabe reziprok, => 10 entspricht 1/10
15 <sub>h</sub>	---	---	Status anfordern -> Rückmeldung via PDO1(tx) CAN-StatusWord
			Falls ein Fehler vorliegt, kann die genauere Spezifikation über den SDO-Dienst und Objekt ERROR-ID [Index 2004 gelesen werden.]
16 <sub>h</sub>	---	---	Fehler löschen
17 <sub>h</sub>	---	---	aktuelle Position an Master senden PDO2(tx)
			- bei absoluter Position: CAN-DataId [Index 2005]= 1 CAN-DeviceData [Index 2006]= abs. Position - bei relativer Position: CAN-DataId [Index 2005]= 2 CAN-DeviceData [Index 2006]= rel. Position

## Spezifische Angaben zur Kommandoliste

Folgende Kommandos dürfen nur im Stillstand gesendet werden. (CAN-StatusWord.Bit10 = 1)

Kommando Nr.	Bedeutung
01	absolute Fahrt
02	relative Fahrt
05	Referenzfahrt
07	Position nullen
09	Beschleunigung
0A	Startgeschwindigkeit
0B	Endgeschwindigkeit
0C	Schrittauflösung

### Verwendung von XFER(Nr.13<sub>h</sub>) und UNIT(Nr.14<sub>h</sub>):

XFER und UNIT sind Formfaktoren, um die Fahrwege nicht in Schritten, sondern in SI-Einheiten eingeben zu können. Sie beziehen sich auf die mit STEPS eingestellte Schrittauflösung von STEPS Schritten pro Umdrehung.

**STEPS muss vor XFER und UNIT definiert sein**

### Beispiel:

Es liegt eine Zahnriemenachse vor mit 70mm Hub pro Motorumdrehung, => XFER= 70

Es soll in Einheiten zu 1/10 mm positioniert werden. Dazu wird UNIT mit dem reziproken Wert der Einheit geladen, also UNIT= 10 (oder bei 1/100 -> 100)

Nun soll um 30mm verfahren werden. Da als Einheit 1/10mm gewählt wurde, ist folgendes Kommando zu übergeben:

fahre relativ um 300 1/10mm 12C<sub>h</sub>

COB-ID	Control	Command	Argument
320 <sub>h</sub>	0	02 <sub>h</sub>	012C <sub>h</sub>

### Bedingung:

XFER \* UNIT muss kleiner gleich STEPS sein

Ist XFER null, wird keine Formatierung vorgenommen. Dann wird die übergebene Position in Schritten interpretiert und mit dem Getriebefaktor. GNUM / GDEN verrechnet

### Verwendung von MREL? und MABS?:

**Relative Fahrt ändert die absolute Position nicht.**

## Handling im Fehlerfall:

Sobald ein Fehler auftritt, wird zunächst das allgemeine Fehlerflag im CAN-StatusWord.Bit3 [Index 2001] und im CAN-ErrorRegister [Index 1001] gesetzt.

Da das CAN-Status Register bei Änderung automatisch via PDO1(tx) gesendet wird, wird dieser Fehlerzustand unmittelbar an den Master gesendet.

Will man nähere Informationen über den Fehler wissen, muss diese über den SDO Dienst angefordert werden, in dem man eine Leseanforderung über das Fehleridentifikationsregister ERROR-ID [Index 2004] sendet. Danach muss der Fehler explizit gelöscht werden.

Eine Tabelle der Fehlernummern und deren Bedeutung ist weiter hinten im Kapitel Fehlerzustände zu finden.

### Beispiel:

Fehler liegt an, CAN-Status Register wird gesendet

COB-ID	CAN-Status	CAN-UserExtin
1A0 <sub>h</sub>	0008 <sub>h</sub>	0000 <sub>h</sub>

Wert immer 0

Leseanforderung: ERROR-ID [Index 2004]

COB-ID	CCD	Index	SubIndex
620 <sub>h</sub>	40 <sub>h</sub>	0420 <sub>h</sub>	00

Antwort auf Leseanforderung

COB-ID	CCD	Index	SubIndex	Daten
5A0 <sub>h</sub>	4B <sub>h</sub>	0420 <sub>h</sub>	00	ERROR-ID

Abhängig vom Fehlerzustand veranlasst der Master entsprechende Aktionen. Auf jeden Fall muss der Fehler explizit zurück gesetzt werden. Dies kann unmittelbar über das PDO1(rx) mit dem CAN-Control Register

COB-ID	CAN-Control
220 <sub>h</sub>	0080 <sub>h</sub>

oder über das Kommando Nr.16<sub>h</sub> erfolgen.

COB-ID	Control	Command	Argument
320 <sub>h</sub>	0	16 <sub>h</sub>	---

## Fehlerzustände: allgemein

Ein Fehler wird dadurch angezeigt, dass L4 nicht dauernd leuchtet und eine der LED's L1...L4 schnell blinkt. Dabei zeigt die entsprechende LED L1...L4 die jeweilige Fehlergruppe 1...4 an, wobei die Gruppe 4 ein schwerwiegenderes Problem darstellt als Grupp3 usw.

Bei Betrieb der Steuerung über serielle Interfaces (RS232, RS485, CANopen) kann ein Fehlerbyte gesendet werden, das genauer Auskunft über den Fehler gibt. Dabei ist jedem Fehler eine Nummer zugeordnet. (siehe Tabelle Fehlernummern). Die Fehlergruppe ist darin nicht angegeben.

Fehlergruppen ab Level 2 werden als fatal eingestuft, d.h. der Motor wird unter Umständen stromlos geschaltet und entsprechende Ausgänge wie Bereitschaft usw. werden deaktiviert.

Ein auftretender Fehlerzustand bleibt gespeichert. Dieser Zustand kann nur vom Anwender manuell über den Frontplattentaster ->] oder durch die Master-Steuerung zurückgesetzt werden. (z.B. durch das Kommando „CLRE“ beim RS232 Interface)

## Fehlergruppe 1: Bedienungsprobleme

Anzeige: L1, L2, L3, L4 = flash, off, off, off

Ursachen:

- Kommunikationsfehler (Schnittstelle),
- Anweisung unbekannt, Parameterfehler

## Fehlergruppe 2: Ablaufprobleme

Anzeige: L1, L2, L3, L4 = off, flash, off, off

Ursachen:

- Unterspannung, Übertemperatur
- Fahrt auf Endschalter, Stop-Signal aktiv
- Schleppfehler, Ablauffehler, usw)

## Fehlergruppe 3: Hardwareprobleme

Anzeige: L1, L2, L3, L4 = off, off, flash, off

Ursachen:

- Endschalter nicht gefunden, falsche Zuordnung,
- kein Indexpuls, kein Referenzsignal
- Überstrom, Kurzschluss in Endstufe
- Probleme bei Geber, Hallsensoren

## Fehlergruppe 4: Systemprobleme

Anzeige: L1, L2, L3, L4 = off, off, off, flash

Ursachen:

- ROM-, RAM-, Stack, usw.

## Tabelle Fehlernummern:

Nr.	Bezeichner:	Ursache:
00	kein Fehler	
01	---	---
02	Systemfehler	---
03	RAM Fehler	---
04	ROM Fehler	---
05	Programm Fehler	Ablaufprogramm fehlerhaft
06	Parameter Fehler	Maschinendaten fehlerhaft
07...10	---	---
11...13	intern	---
14...19	---	---
20	Leistungsteil defekt	Kurzschluss in Endstufe
21	Überstrom	Motor defekt, Motor blockiert
22	Hall-Sensor	Verdrahtungsfehler, Kabelbruch, Sensor defekt, EMV-Störungen
23	Hex-Schalter	mechanisch defekt
24	Eingang IN1	schaltet nicht
25	Eingang IN2	schaltet nicht
26	Eingang IN3	schaltet nicht
27	Eingang IN4	schaltet nicht
28	Eingang IN1...IN4	falsche Zuordnung, z.B. bei Endschalter- Referenzschalter oder Stopeingang
29	Indexpuls (Geber)	schaltet nicht
30	Geber	Verdrahtungsfehler, Kabelbruch, Geber defekt
31...39	---	---

40	Kommandobuffer läuft über	kein CR erkannt innerhalb 16 Zeichen, Schnittstellenformat stimmt nicht
41	Kommando unbekannt	falsche Anweisung, Störungen auf Kabel
42	Parameterfehler	Bereichsgrenze überschritten
43	---	---
44	Kommando overrun	vorherige Anweisung muss erst beendet sein
45	Sendefehler	---
46	Empfangsfehler	Schnittstellenformat, Störungen auf Leitung
47...49	---	---
50...59	---	---
60	Übertemperatur	Kühlung unzureichend Stromabsenkung aktiv? Option Kühlkörper ?, Fremdbelüftung ?
61	Schleppfehler	Regelabweichung zwischen Soll- und Istposition zu groß Regelparametereinstellung ? mehr Kp, mehr Integralanteil Versorgungsspannung zu gering, Endgeschwindigkeit zu hoch zu hohe Beschleunigungskräfte (Rampenparameter), Bearbeitungskräfte
62	Unterspannung	Netzteil bricht ein Motorspannung wurde abgeschalten
63	---	---
64	WatchDog	Sicherheitstimer Triggerzeit abgelaufen (Steuerung hängt)
65...66	---	---
67	Programm download	Ablaufprogramm konnte nicht geladen werden
68	Fahrt auf Endschalter	nur bei Referenzfahrt erlaubt, sonst nicht
69	Stop-Eingang war aktiv	asynchroner Abbruch von Extern
70...79	---	---
80...89	---	---
90...99	---	---

## Problemhilfen

### Motor ohne Haltemoment, obwohl Spannung anliegt

- die Sicherung im Leistungsteil ist defekt
- die Motorspannung liegt unter 24 Volt
- Eingang: OFF ist aktiv

### sofort Fehlermeldung nach dem Einschalten

- die Endstufe ist defekt
- der Motor hat einen Kurzschluss
- die Temperatur liegt noch über 70 Grad

### plötzliche Knackgeräusche im Motor

- Unterspannung an der Motorversorgung (<24Volt)
- zu niedriger Leiterquerschnitt im Versorgungskabel

### Motor läuft an, kommt nicht auf die Enddrehzahl

- die Motorspannung ist zu gering
- zu hohe Beschleunigung oder Startfrequenz
- Drehmomentspitzen in der Fahrstrecke
- zu lange, zu dünne Motorleitungen
- Regelparameter nicht optimal

### der Motor verliert einzelne Schritte und driftet weg

- Signalamplituden (Puls/Richtung) zu gering
- Störungen auf Signalleitungen
- mechanische Wellenkopplung hat Schlupf

### der Motor vibriert

- zu hohe Start/Stop-Frequenz
- Motorwicklungen falsch angeschlossen
- Drehgeber/Hall-Sensoren falsch angeschlossen
- Motorkabelbruch
- niedere Schrittfrequenz bei Vollschritt ohne Last

### der Motor wird sehr warm

- bis 85 Grad Celsius kein Problem

### stark unterschiedliche Schrittwinkel im Mikroschritt

- der Motor hat zu große Wicklungsinduktivität
- der Motor wird weit unter dem Nennstrom betrieben
- der Motor wird über dem Nennstrom betrieben

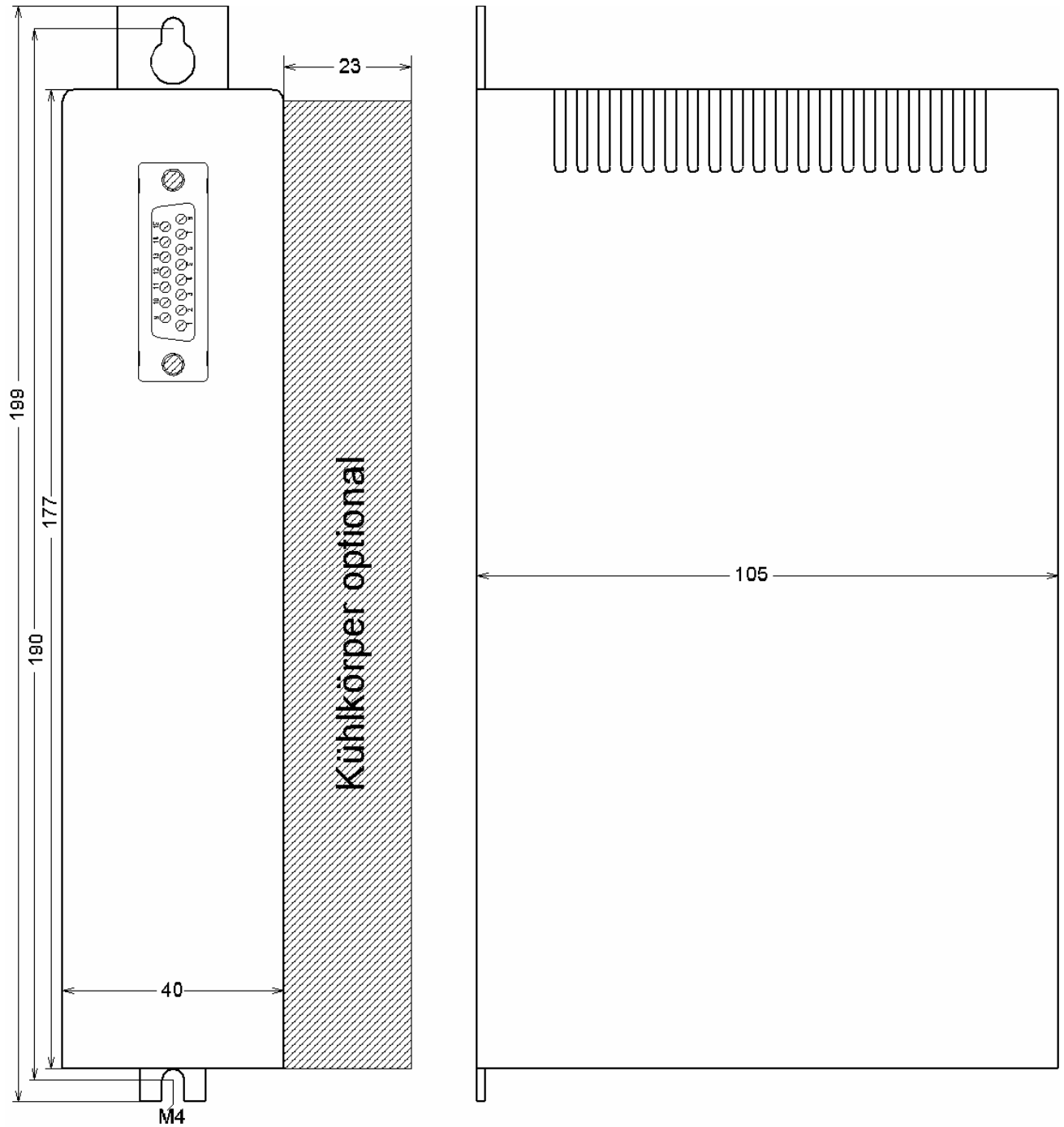
### Zirbelgeräusche in bestimmten Schrittpositionen

- zu hohe Motorspannung bei geringem Strom
- zu geringe Motorinduktivität

### sofort Fehlermeldung nach Fahrbefehl

- Endschalter ist aktiv
- Schleppfehler, Drehgeberproblem

# Abmessungen



## Technische Daten

### Versorgung:

absolute max. Spannung:	Typ: 80V, 130V
minimale Spannung:	85V, 135V
empfohlene Spannung:	24V, 60V
Spannungsrippel:	80V, 130V
Einschaltstrom:	< 2,0Vss
Absicherung:	< 4,0A
Netzteilko:	5,0Amt
Versorgungszuführung:	>6800yF
Distanz zum Netzteilko:	0,75mm <sup>2</sup>
	<0,3m

### Motoranschluß:

Kabelquerschnitt:	<4A	>0,75mm <sup>2</sup>
	>4A	>1,00mm <sup>2</sup>
Kabellänge:		<10m

### Interface X1:

galvanisch getrennt

#### Puls, Richtung, IN1, IN2

Eingangsspannung	low:	< 1,0V
	high:	>3,5V
	max.:	28V
Eingangsstrom:	max.:	16mA
Pulsdauer:		>5ys
Pulspause:		>5ys
Pulsflanke:		<2ys

#### Bereitschaft

Schaltspannung:	<30V
Innenwiderstand:	<15 Ohm
Schaltstrom:	< 50 mA
Last:	ohmisch

### Interface X2:

galvanisch getrennt

#### IN1...IN4

Eingangsspannung	low:	< 1,0V
	high:	>3,5V
	max.:	28V
Eingangsstrom:	max.:	18mA
Pulsdauer:		>1ms
Pulspause:		>1ms

#### OUT1, OUT2

Schaltspannung:	<30V
Innenwiderstand:	<15 Ohm
Schaltstrom:	< 50 mA
Last:	ohmisch

### Interface X3:

galvanisch getrennt

Drehgeberversorgung	5V, max. 70mA
Drehgeber A, B, I	TTL-Pegel
	max. 100kHz
Hall-Sensors	5V, >13mA

### Temperaturüberwachung:

Lüfterautomatik aktiv:	> ca. 60°
Schutz-Abschaltung:	> ca. 70 °

### Umgebungsbedingungen:

Betriebstemperaturbereich:	0...40°C
Lagertemperaturbereich:	-10...60°C
Schutz:	IP30
Verschmutzungsgrad:	2, keine Betauung
Gewicht:	0,5kg

### EMV / Störunterdrückung:

Funkstörfeldstärke:	EN55011B
ESD:	4kV
Burst:	IEC-Level 4 / 2,5kHz

## Allgemeine Installationsanforderungen

Das Gehäuse ist generell separat zu erden. Dafür ist an der Frontplatte eine Erdungsschraube vorgesehen. Jede Komponente ist mit einem separaten Erdungskabel an einem zentralen „Erdungspunkt“ anzuschließen. Meist ist dies das Maschinenbett oder eine Erdungsschiene im Schaltschrank.

Überprüfen Sie vor der Inbetriebnahme, ob die geforderte Leistung für Ihre Applikation ausreichend ist und dass angegebene Maximalwerte nicht überschritten werden.

Einbaulage senkrecht, Lüftereintritt und Lüftungsschlitze frei halten

Motorkabel sind generell in geschirmter Ausführung zu installieren. Bei gleichem Potential von Motorflansch und Steuerung (kurze Distanz) wird der Schirm beidseitig geerdet. Ansonsten wird nur eine einseitige Anbindung empfohlen in der Art, dass motorseits der Schirm über einen Kondensator galvanisch getrennt angebunden wird.

Generell darf der Potentialunterschied nur im Bereich von einigen wenigen mVolt liegen
--

Bei symmetrischen Motorleitungen wie beim 2-Phasen-Schrittmotor (Hin- und Rückleitung) werden verdrehte Aderpaare empfohlen.

Signalkabel sind ebenfalls zu schirmen. Bei Hin- und Rückleitung werden verdrehte Aderpaare empfohlen.

Signalkabel sind von Motorkabel getrennt zu verlegen. Lange parallele Führungen sind zu vermeiden, Kreuzungen möglichst senkrecht auszuführen.

Überprüfen Sie mögliche Einstellungen auf Richtigkeit.

## Sicherheitshinweise / Schutzanforderungen

Die Installation des Produkts darf nur durch eine ausgebildete Fachkraft (Elektro) durchgeführt werden. Es sind die länderspezifischen Bestimmungen wie Unfallverhütung, Errichten von elektrischen und mechanischen Anlagen und Funkentstörung zu beachten.

Bei nicht sachgemäßen Betrieb des Produkts können Personen verletzt, das Produkt und weitere extern angeschlossene Komponenten beschädigt oder die Umwelt unzulässig belastet werden

Der Betrieb ist nur mit geschlossenem Gehäuse erlaubt. Das Produkt darf wegen evtl. noch vorhandener Hochspannung grundsätzlich nicht geöffnet werden, auch nicht nach längerem Stillstand. Stellen Sie sicher, dass Kinder keinen direkten Zugang haben.

Es dürfen keinerlei technische Veränderungen am Gerät vorgenommen werden.

Das Gehäuse ist generell und separat zu erden. Dafür ist extra eine Erdungsschraube an der Frontplatte vorgesehen. Die Erdung hat vor der Inbetriebnahme zu erfolgen.

Unter keinen Umständen dürfen Stecker unter Spannung oder Betriebszuständen abgezogen oder gesteckt werden. Alle Montagearbeiten haben spannungslos zu erfolgen.

Der Betrieb in feuchter oder Spritzwasser gefährdeter Umgebung ist nicht zulässig