

MS205D
C-863 Mercury Controller
Benutzerhandbuch

Version: 2.0.0

Datum: 20.03.2013



Dieses Dokument beschreibt folgendes Produkt:

- **C-863.11**
Mercury DC-Motorcontroller, 1 Kanal, mit Weitbereichsnetzteil



Physik Instrumente (PI) GmbH & Co. KG ist Inhaberin der nachfolgend aufgeführten Marken:

PI®, PIC®, PICMA®, Picoactuator®, PIFOC®, PILine®, PInano®, PiezoWalk®, NEXACT®, NEXLINE®, NanoCube®, NanoAutomation®

Bei den nachfolgend aufgeführten Bezeichnungen handelt es sich um geschützte Firmennamen bzw. eingetragene Marken fremder Inhaber:

Microsoft, Windows, LabVIEW

© 2013 Physik Instrumente (PI) GmbH & Co. KG, Karlsruhe, Deutschland. Die Texte, Bilder und Zeichnungen dieses Handbuchs sind urheberrechtlich geschützt. Physik Instrumente (PI) GmbH & Co. KG behält insoweit sämtliche Rechte vor. Die Verwendung dieser Texte, Bilder und Zeichnungen ist nur auszugsweise und nur unter Angabe der Quelle erlaubt.

Originalbetriebsanleitung

Erstdruck: 20.03.2013

Dokumentnummer: MS205D, CBo, Version 2.0.0

Änderungen vorbehalten. Dieses Handbuch verliert seine Gültigkeit mit Erscheinen einer neuen Revision. Die jeweils aktuelle Revision ist auf unserer Website zum Herunterladen (S. 5) verfügbar.



Inhalt

1	Über dieses Dokument	1
1.1	Ziel und Zielgruppe dieses Benutzerhandbuchs	1
1.2	Symbole und Kennzeichnungen	1
1.3	Begriffserklärung	3
1.4	Mitgeltende Dokumente	4
1.5	Handbücher herunterladen	5
2	Sicherheit	7
2.1	Bestimmungsgemäße Verwendung	7
2.2	Allgemeine Sicherheitshinweise	7
2.2.1	Organisatorische Maßnahmen	8
2.2.2	Maßnahmen bei der Installation	8
2.2.3	Maßnahmen bei der Inbetriebnahme	9
2.2.4	Maßnahmen beim Betrieb	10
2.2.5	Maßnahmen bei der Wartung	11
3	Produktbeschreibung	13
3.1	Merkmale und Anwendungsbereich	13
3.2	Modellübersicht	15
3.3	Produktansicht	15
3.3.1	Vorderwand	15
3.3.2	Rückwand	16
3.4	Lieferumfang	17
3.5	Optionales Zubehör	18
3.6	Funktionsprinzipien	19
3.6.1	Blockdiagramm	19
3.6.2	Kommandierbare Elemente	20
3.6.3	Wichtige Komponenten der Firmware	22
3.6.4	Betriebsarten	23
3.6.5	Physikalische Einheiten	23
3.6.6	Auslösen von Bewegungen	25
3.6.7	Erzeugung des Dynamikprofils	26
3.6.8	Regelalgorithmus und weitere Stellwertkorrekturen	29
3.6.9	On-Target-Status	32
3.6.10	Referenzschaltererkennung	33
3.6.11	Endschaltererkennung	34
3.6.12	Stellweg und Verfahrbereichsgrenzen	35
3.6.13	Referenzwertbestimmung	40

3.7	Kommunikationsschnittstellen	44
3.8	PC-Softwareübersicht.....	45
3.9	Verstellerdatenbanken.....	47
4	Auspacken	49
5	Schnellstart	51
6	Installation	57
6.1	PC-Software installieren	57
6.1.1	Erstinstallation ausführen	57
6.1.2	Updates installieren	58
6.1.3	Kundenspezifische Verstellerdatenbank installieren	60
6.2	C-863 montieren	62
6.3	C-863 erden.....	63
6.4	Netzteil an C-863 anschließen	63
6.5	Versteller anschließen	64
6.6	PC anschließen	64
6.6.1	An RS-232-Schnittstelle anschließen	65
6.6.2	An USB-Schnittstelle anschließen.....	66
6.6.3	Daisy-Chain-Netzwerk aufbauen.....	66
6.7	Analogen Joystick anschließen	68
6.8	Digitale Ein- und Ausgänge anschließen	69
6.8.1	Digitale Ausgänge anschließen	69
6.8.2	Digitale Eingänge anschließen	70
6.9	Analoge Signalquellen anschließen	71
7	Inbetriebnahme	73
7.1	Allgemeine Hinweise zur Inbetriebnahme	73
7.2	DIP-Schalter-Einstellungen anpassen.....	74
7.2.1	Generelle Vorgehensweise	74
7.2.2	Controlleradresse	75
7.2.3	Baudrate	76
7.2.4	Logikpegel der Endschalter	76
7.2.5	Update-Modus	77
7.3	C-863 einschalten.....	77
7.4	Kommunikation herstellen	78
7.4.1	Kommunikation über RS-232 herstellen.....	78
7.4.2	Kommunikation über USB herstellen.....	80
7.4.3	Kommunikation für vernetzten Controller herstellen	81



7.5	Bewegungen starten.....	87
7.6	Sprungantwort aufzeichnen.....	92
7.7	Regelparameter optimieren.....	96
8	Betrieb.....	99
8.1	Bewegungsfehler.....	99
8.1.1	Schutzfunktionen des C-863.....	99
8.1.2	Betriebsbereitschaft wiederherstellen.....	100
8.2	Datenrekorder.....	101
8.2.1	Eigenschaften des Datenrekorders.....	101
8.2.2	Datenrekorder einrichten.....	101
8.2.3	Aufzeichnung starten.....	103
8.2.4	Aufgezeichnete Daten auslesen.....	103
8.3	Digitale Ausgangssignale.....	103
8.3.1	Befehle für digitale Ausgänge.....	104
8.3.2	Triggermodus "Position Distance" einrichten.....	106
8.3.3	Triggermodus "On Target" einrichten.....	108
8.3.4	Triggermodus "Motion Error" einrichten.....	109
8.3.5	Triggermodus "In Motion" einrichten.....	109
8.3.6	Triggermodus "Position + Offset" einrichten.....	110
8.3.7	Triggermodus "Single Position" einrichten.....	112
8.3.8	Signalpolarität einstellen.....	113
8.4	Digitale Eingangssignale.....	113
8.4.1	Befehle und Parameter für digitale Eingänge.....	114
8.4.2	Digitale Eingangssignale in Makros verwenden.....	116
8.4.3	Digitale Eingangssignale als Schaltersignale verwenden.....	117
8.5	Analoge Eingangssignale.....	119
8.5.1	Befehle für analoge Eingänge.....	119
8.5.2	Analoge Eingangssignale in Makros verwenden.....	120
8.6	Joystick-Steuerung.....	121
8.6.1	Funktionsweise der Joystick-Steuerung.....	121
8.6.2	Befehle und Parameter für Joystick-Steuerung.....	121
8.6.3	Achsenbewegung steuern.....	123
8.6.4	Joystick kalibrieren.....	125
8.6.5	Verfügbare Joystick-Geräte.....	129
8.7	Controllermakros.....	131
8.7.1	Übersicht: Makrofunktionalitäten und Beispielmakros.....	131
8.7.2	Befehle und Parameter für Makros.....	132
8.7.3	Mit Makros arbeiten.....	133
8.7.4	Makrobeispiel: Synchronisation zweier Controller.....	144
8.7.5	Makrobeispiel: Bewegung per Tastendruck stoppen.....	145
8.7.6	Makrobeispiel: Joysticksteuerung mit Speicherung von Positionen.....	147
8.7.7	Makrobeispiel: Joysticksteuerung mit Änderung der Geschwindigkeit.....	152

9	GCS-Befehle	155
9.1	Schreibweise	155
9.2	GCS-Syntax für Syntaxversion 2.0.....	156
9.3	Empfänger- und Senderadresse	158
9.4	Variablen.....	160
9.5	Befehlsübersicht	162
9.6	Befehlsbeschreibungen für GCS 2.0.....	166
9.7	Fehlercodes	262
10	Anpassen von Einstellungen	279
10.1	Parameter im C-863 ändern	279
10.1.1	Allgemeine Befehle für Parameter.....	279
10.1.2	Spezielle Befehle für Parameter.....	280
10.1.3	Parameterwerte in Textdatei sichern.....	281
10.1.4	Parameter ändern: Generelle Vorgehensweise	282
10.2	Verstellertyp anlegen oder ändern	284
10.3	Parameterübersicht	288
11	Wartung	297
11.1	C-863 reinigen	297
11.2	Firmware aktualisieren	298
12	Störungsbehebung	303
13	Kundendienst	309
14	Technische Daten	311
14.1	Spezifikationen	311
14.1.1	Datentabelle.....	311
14.1.2	Bemessungsdaten	312
14.1.3	Umgebungsbedingungen und Klassifizierungen	313
14.2	Systemanforderungen	314
14.3	Abmessungen.....	314
14.4	Pinbelegung.....	315
14.4.1	DC Motor only.....	315
14.4.2	I/O	316
14.4.3	Kabel C-170.IO für den Anschluss an die I/O-Buchse	316
14.4.4	Joystick	317
14.4.5	Kabel C-819.20Y für Joystick C-819.20	318
14.4.6	RS-232 In und RS-232 Out	319
14.4.7	Netzteilanschluss 15-30 VDC	320



15	Altgerät entsorgen	321
16	EG-Konformitätserklärung	323

1 Über dieses Dokument

In diesem Kapitel

Ziel und Zielgruppe dieses Benutzerhandbuchs	1
Symbole und Kennzeichnungen.....	1
Begriffserklärung	3
Mitgeltende Dokumente.....	4
Handbücher herunterladen.....	5

1.1 Ziel und Zielgruppe dieses Benutzerhandbuchs

Dieses Benutzerhandbuch enthält die erforderlichen Informationen für die bestimmungsgemäße Verwendung des C-863.

Grundsätzliches Wissen zu geregelten Systemen, zu Konzepten der Bewegungssteuerung und zu geeigneten Sicherheitsmaßnahmen wird vorausgesetzt.

Die aktuellen Versionen der Benutzerhandbücher stehen auf unserer Website zum Herunterladen (S. 5) bereit.

1.2 Symbole und Kennzeichnungen

In diesem Benutzerhandbuch werden folgende Symbole und Kennzeichnungen verwendet:

HINWEIS



Gefährliche Situation

Bei Nichtbeachtung drohen Sachschäden.

- Maßnahmen, um die Gefahr zu vermeiden.

INFORMATION

Informationen zur leichteren Handhabung, Tricks, Tipps, etc.

Symbol/ Kennzeichnung	Bedeutung
1.	Handlung mit mehreren Schritten, deren Reihenfolge eingehalten werden muss
2.	
➤	Handlung mit einem Schritt oder mehreren Schritten, deren Reihenfolge nicht relevant ist
▪	Aufzählung
S. 5	Querverweis auf Seite 5
RS-232	Bedienelement-Beschriftung auf dem Produkt (Beispiel: Buchse der RS-232 Schnittstelle)
	Auf dem Produkt angebrachtes Warnzeichen, das auf ausführliche Informationen in diesem Handbuch verweist.
Start > Einstellungen	Menüpfad in der PC-Software (Beispiel: Zum Aufrufen des Menüs muss nacheinander auf die Menüeinträge Start und Einstellungen geklickt werden)
SV0?	Befehlszeile oder Befehl aus dem universellen Befehlssatz GCS von PI (Beispiel: Befehl zum Abfragen des Servomodus)
Device S/N	Parameterbezeichnung (Beispiel: Parameter, in dem die Seriennummer gespeichert ist)
5	Wert, der über die PC-Software eingegeben bzw. ausgewählt werden muss

1.3 Begriffserklärung

Begriff	Erklärung
PC-Software	Software, die auf dem PC installiert wird.
Firmware	Software, die auf dem Controller installiert ist.
Flüchtiger Speicher	RAM-Baustein, in dem bei eingeschaltetem Controller die Parameter gespeichert sind (Arbeitsspeicher).
Permanenter Speicher	EEPROM-Speicherchip (Festspeicher), von dem beim Start des Controllers die Standardwerte der Parameter in den flüchtigen Speicher geladen werden.
Achse	Auch als "logische Achse" bezeichnet. Die logische Achse bildet die Bewegung des Verstellers in der Firmware des C-863 ab. Bei Verstellern, die Bewegung in mehreren Richtungen erlauben (z. B. in X, Y und Z), entspricht jede Bewegungsrichtung einer logischen Achse.
Versteller	An den C-863 angeschlossene Mechanik. Bei Verstellern mit nur einer Bewegungsachse ist die Bezeichnung "Achse" gleichbedeutend mit "Versteller". Versteller, die Bewegung in mehreren Achsen erlauben, werden auch als "mehrachsiges Versteller" bezeichnet. Für diese Versteller muss zwischen den einzelnen Achsen unterschieden werden. In diesem Handbuch werden auch Aktoren, d. h. Antriebskomponenten ohne bewegte Plattform (z. B. Präzisionslinearaktoren) als Versteller bezeichnet.
Daisy Chain	Verkabelungs-Schema, bei dem ein Controller mit dem nächsten nacheinander verbunden wird (Reihenschaltungsprinzip). Dabei ist der erste Controller direkt mit dem PC verbunden. Die weiteren Controller sind jeweils mit ihren Vorgängern verbunden, so dass eine Kette entsteht. Das Signal zu und von einem Controller geht über seine Vorgänger bis zum PC hin.
Inkrementeller Positionssensor	Sensor (Encoder) zur Erfassung von Lageänderungen oder Winkeländerungen. Die Signale des inkrementellen Positionssensors werden für die Rückmeldung der Achsenposition verwendet. Nach dem Einschalten des Controllers muss eine Referenzwertbestimmung durchgeführt werden, bevor absolute Zielpositionen kommandiert und erreicht werden können.
Stellwert	Der Stellwert ist die Eingangsgröße für den PWM-Konverter des C-863. Der PWM-Konverter wandelt den Stellwert in das PWM-Signal für die Achse des Verstellers um.
Dynamikprofil	Umfasst die vom Profilgenerator des C-863 für jeden Zeitpunkt der Bewegung berechnete Zielposition, Geschwindigkeit und Beschleunigung der Achse. Die errechneten Werte werden "kommandierte Werte" genannt.
GCS	PI General Command Set; Befehlssatz für Controller von PI. Piezosteuerungen und Servocontroller können dank GCS mit minimalem Programmieraufwand gemeinsam betrieben werden.

1.4 Mitgelieferte Dokumente

Alle in dieser Dokumentation erwähnten Geräte und Programme von PI sind in separaten Handbüchern beschrieben.

Beschreibung	Dokument
Kurzversion des Handbuchs für C-863.11	MS205Dqu Benutzerhandbuch Kurzversion
Mercury GCS LabVIEW-Treiberbibliothek	MS206E Software Manual
PI GCS 2.0 DLL für C-x63.11	MS212E Software Manual
GCS Array Datenformatbeschreibung	SM146E Software Manual
PIMikroMove®	SM148E Software Manual
PIStageEditor Software für die Verwaltung von Verstellerdatenbanken	SM144E Software Manual
PI Update Finder: Updates suchen und herunterladen	Technical Note A000T0028
PI Update Finder: PC ohne Internetverbindung aktualisieren	Technical Note A000T0032
Anpassen von Software, die für C-863.10 geschrieben wurde, an die Verwendung mit C-863.11	Technical Note A000T0029

INFORMATION

Das für den Betrieb mit Schrittmotoren vorgesehene Modell C-663.11 ist in einem separaten Handbuch (MS208D) beschrieben.

1.5 Handbücher herunterladen

INFORMATION

Wenn ein Handbuch auf unserer Website fehlt oder Probleme beim Herunterladen auftreten:

- Wenden Sie sich an unseren Kundendienst (S. 309).

Auf unserer Website finden Sie die Handbücher in ihrer aktuellen Version. Um ein Handbuch herunterzuladen, gehen Sie wie folgt vor:

1. Öffnen Sie die Website **<http://www.pi-portal.ws>**.
2. Klicken Sie auf **Downloads**.
3. Klicken Sie auf die entsprechende Kategorie (z. B. **C Motion Controllers**).
4. Klicken Sie auf den entsprechenden Produktcode (z. B. **C-863.11**).

Eine Übersicht der verfügbaren Dateitypen zum gewählten Produkt wird angezeigt.

5. Wenn **(0 Files)** in der Zeile **Documents** angezeigt wird, loggen Sie sich zur Anzeige und zum Download der Dokumente wie folgt ein:
 - a) Legen Sie die CD des Produkts in das entsprechende PC-Laufwerk ein.
 - b) Öffnen Sie das Verzeichnis **Manuals**.
 - c) Öffnen Sie die Release News (z. B. **C-663.11_Releasenews_V_x_x_x.pdf**) von der CD des Produkts.
 - d) Entnehmen Sie dem Abschnitt **User login for software download** in den Release News den Benutzernamen (user name) und das Kennwort (password).
 - e) Geben Sie auf der Website im Bereich **User login** am linken Seitenrand den Benutzernamen und das Kennwort in die entsprechenden Felder ein.
 - f) Klicken Sie auf **Login**.

Wenn immer noch **Documents (0 Files)** angezeigt wird, sind keine Handbücher vorhanden:

- Wenden Sie sich an unseren Kundendienst (S. 309).

6. Klicken Sie auf **Documents**.
7. Klicken Sie auf das gewünschte Handbuch und speichern Sie es auf der Festplatte Ihres PC oder auf einem Datenträger.

2 Sicherheit

In diesem Kapitel

Bestimmungsgemäße Verwendung	7
Allgemeine Sicherheitshinweise	7

2.1 Bestimmungsgemäße Verwendung

Der C-863 ist ein Laborgerät im Sinne der DIN EN 61010. Er ist für die Verwendung in Innenräumen und in einer Umgebung vorgesehen, die frei von Schmutz, Öl und Schmiermitteln ist.

Entsprechend seinem Aufbau ist der C-863 dafür vorgesehen, Versteller von PI, die mit DC-Motoren oder Voice-Coil-Antrieben ausgestattet sind, zu betreiben.

Der C-863 ist für den geregelten Betrieb mit inkrementellen Positionssensoren vorgesehen. Außerdem kann er die Referenz- und Endschaltersignale des angeschlossenen Verstellers auslesen und weiterverarbeiten.

Der C-863 darf nur unter Einhaltung der technischen Spezifikationen und Anweisungen in diesem Benutzerhandbuch verwendet werden. Für die Prozessvalidierung ist der Benutzer verantwortlich.

2.2 Allgemeine Sicherheitshinweise

Der C-863 ist nach dem Stand der Technik und den anerkannten sicherheitstechnischen Regeln gebaut. Bei unsachgemäßer Verwendung des C-863 können Benutzer gefährdet werden und/oder Schäden am C-863 entstehen.

- Benutzen Sie den C-863 nur bestimmungsgemäß und in technisch einwandfreiem Zustand.
- Lesen Sie das Benutzerhandbuch.
- Beseitigen Sie Störungen, die die Sicherheit beeinträchtigen können, umgehend.

Der Betreiber ist für den korrekten Einbau und Betrieb des C-863 verantwortlich.

- Installieren Sie den C-863 in der Nähe der Stromversorgung, damit der Netzstecker schnell und einfach vom Netz getrennt werden kann.
- Verwenden Sie zum Anschließen des C-863 an die Stromversorgung die mitgelieferten Komponenten (Netzteil, Adapter, Netzkabel (S. 18)).
- Wenn eine der mitgelieferten Komponenten für das Anschließen an die Stromversorgung ersetzt werden muss, verwenden Sie eine ausreichend bemessene Komponente.

2.2.1 Organisatorische Maßnahmen

Benutzerhandbuch

- Halten Sie dieses Benutzerhandbuch ständig am C-863 verfügbar. Wenn das Benutzerhandbuch verloren geht oder unbrauchbar wird, wenden Sie sich an unseren Kundendienst (S. 309).
- Fügen Sie alle vom Hersteller bereitgestellten Informationen, z. B. Ergänzungen und Technical Notes, zum Benutzerhandbuch hinzu.
- Wenn Sie den C-863 an Andere weitergeben, fügen Sie dieses Handbuch bei sowie alle sonstigen vom Hersteller bereitgestellten Informationen.
- Führen Sie Arbeiten grundsätzlich anhand des vollständigen Benutzerhandbuchs durch. Fehlende Informationen aufgrund eines unvollständigen Benutzerhandbuchs können zu leichten Verletzungen und Sachschäden führen.
- Installieren und bedienen Sie den C-863 nur, nachdem Sie dieses Benutzerhandbuch gelesen und verstanden haben.

Personalqualifikation

Nur autorisiertes und entsprechend qualifiziertes Personal darf den C-863 in Betrieb nehmen, bedienen, warten und reinigen.

2.2.2 Maßnahmen bei der Installation

- Installieren Sie den C-863 in der Nähe der Stromversorgung, damit der Netzstecker schnell und einfach vom Netz getrennt werden kann.
- Verwenden Sie nur Kabel und Verbindungen, die den lokalen Sicherheitsbestimmungen genügen.

Das Anschließen eines Verstellers mit Schrittmotor an einen DC-Motorcontroller kann irreparable Schäden verursachen.

- Schließen Sie an den C-863 nur einen Versteller mit DC-Motor oder Voice-Coil-Antrieb an.

Das gleichzeitige Anschließen der USB- und der RS-232-Schnittstelle des Controllers an den PC kann den PC oder den Controller beschädigen.

- Schließen Sie entweder die USB- oder die RS-232-Schnittstelle an den PC an.

Die Ausgangsspannung an den Pins **Motor +** und **Motor -** der Buchse **DC Motor only** kann so hoch wie die vom Netzteil gelieferte Versorgungsspannung sein. Versteller ohne PWM-Verstärker sind an diese Pins angeschlossen und können durch zu hohe Ausgangsspannung beschädigt werden.

- Schließen Sie ein Netzteil an, dessen Ausgangsspannung die zulässige Betriebsspannung des Verstellers **nicht** überschreitet.

Ungeeignete Einstellungen der Regelparameter des C-863 können den Versteller zum Schwingen bringen. Schwingungen können den Versteller und/oder die auf ihm angebrachte Last beschädigen.

- Befestigen Sie den Versteller und alle Lasten ausreichend.

2.2.3 Maßnahmen bei der Inbetriebnahme

Beim Ändern von Systemeinstellungen im permanenten Speicher gehen die ursprünglichen Einstellungen verloren. Ungünstige Einstellungen können zum Schwingen des Verstellers, zu verschlechtertem Einschwingverhalten und zu verringerter Positioniergenauigkeit führen.

- Ändern Sie interne Systemeinstellungen des C-863 im permanenten Speicher nur wenn nötig.
- Halten Sie Rücksprache mit unserem Kundendienst (S. 309), wenn Sie sich nicht sicher sind, ob eine Änderung der Systemeinstellungen nötig ist.

Die Auswahl eines falschen Verstellertyps in der PC-Software kann Schäden am Versteller verursachen.

- Stellen Sie sicher, dass der in der PC-Software ausgewählte Verstellertyp mit dem angeschlossenen Versteller übereinstimmt.

2.2.4 Maßnahmen beim Betrieb

Ungeeignete Einstellungen der Regelparameter des C-863 können den Versteller zum Schwingen bringen. Schwingungen können den Versteller und/oder die auf ihm angebrachte Last beschädigen.

- Wenn der Versteller schwingt (ungewöhnliches Laufgeräusch), schalten Sie den Servomodus sofort aus oder trennen Sie den C-863 von der Stromversorgung.
- Schalten Sie den Servomodus erst wieder ein, nachdem Sie die Einstellungen der Regelparameter des C-863 geändert haben; siehe "Regelparameter optimieren" (S. 96).

Der Aufprall eines bewegten Teils am Ende des Stellwegs oder auf einem Hindernis sowie hohe Beschleunigungen können Schäden oder erheblichen Verschleiß an der Mechanik verursachen.

- Vermeiden Sie Bewegungen im ungeregelten Betrieb.
- Wenn Bewegungen im ungeregelten Betrieb notwendig sind:
 - Setzen Sie den Stellwert mit dem Befehl **SM0** so, dass sich die Achse mit niedriger Geschwindigkeit bewegt.
 - Stoppen Sie die Achse rechtzeitig. Verwenden Sie dazu die Befehle **#24**, **STP** oder **HLT**, oder setzen Sie den Stellwert mit dem Befehl **SM0** auf null.
- Deaktivieren Sie **nicht** per Parametereinstellung die Endschalterauswertung durch den C-863.
- Prüfen Sie die Funktion der Endschalter bei etwa 10 % bis 20 % der Maximalgeschwindigkeit.
- Halten Sie bei einer Fehlfunktion der Endschalter die Bewegung sofort an.
- Stellen Sie sicher, dass das Ende des Stellwegs mit geringer Geschwindigkeit angefahren wird.
- Wenn das bewegte Teil oder die an ihm montierte Last an ein Hindernis stößt, schalten Sie den Motor ab.
- Wenn möglich, passen Sie die Verfahrbereichsgrenzen in der zur Bewegungskommandierung verwendeten Software an Ihr mechanisches System an.
- Bestimmen Sie die Maximalgeschwindigkeit für Ihre Anwendung.

Wenn kein Joystick an den C-863 angeschlossen ist, kann die Aktivierung des Joysticks in der Software unbeabsichtigte Bewegungen der angeschlossenen Achse verursachen.

- Aktivieren Sie den Joystick in der Software nur, wenn tatsächlich ein Joystick an den C-863 angeschlossen ist.

Wenn der Servomodus ausgeschaltet ist, z. B. nach Auftreten eines Bewegungsfehlers, kann die Bremse des Verstellers per Befehl deaktiviert werden. Das Deaktivieren der Bremse kann unbeabsichtigte Bewegungen des Verstellers verursachen.

- Sichern Sie den Versteller gegen unbeabsichtigte Bewegungen, bevor Sie die Bremse per Befehl deaktivieren!

2.2.5 Maßnahmen bei der Wartung

Der C-863 enthält elektrostatisch gefährdete Bauteile.

- Öffnen Sie **nicht** das Gehäuse des C-863.
- Trennen Sie vor dem Reinigen den C-863 von der Stromversorgung, indem Sie den Netzstecker ziehen.

3 Produktbeschreibung

In diesem Kapitel

Merkmale und Anwendungsbereich	13
Modellübersicht	15
Produktansicht	15
Lieferumfang	17
Optionales Zubehör	18
Funktionsprinzipien	19
Kommunikationsschnittstellen	44
PC-Softwareübersicht	45
Verstellerdatenbanken	47

3.1 Merkmale und Anwendungsbereich

Der Mercury DC-Motor Controller ist die perfekte Lösung zum Aufbau preisgünstiger und flexibler Positioniersysteme, bei denen ein Präzisionsversteller mit einem PC oder programmierbaren Controller angesteuert werden soll. Neben dem C-863 gehört zur Mercury-Baureihe der erfolgreiche C-663 Mercury Step Schrittmotor-Controller. Die Controller der Mercury-Baureihe verwenden dieselben Befehlssätze und können miteinander vernetzt werden.

Die Produktmerkmale des C-863 umfassen:

- RS-232- und USB-Schnittstelle
- Stand-Alone-Betrieb
- Daisy-Chain-Vernetzbarkeit für Mehrachsenbetrieb
- Vernetzbar und kompatibel mit allen Motorcontrollern der Mercury-Baureihe, einschließlich Mercury Step
- Joystick-Anschluss für manuelle Bedienung
- Nichtflüchtiger Makrospeicher
- Parameteränderung während des Betriebs
- TTL-Eingänge für End- und Referenzschalter
- Steuersignal für Motorbremse
- Programmierbare Ein-/Ausgänge

Mehrachsenbetrieb von DC- und Schrittmotoren

Der C-863.11 Mercury DC-Motor Controller hat denselben Befehlssatz wie der Mercury Step Schrittmotor-Controller C-663.11. Bis zu 16 Mercury-Controller (für DC- und Schrittmotoren) können vernetzt und über dieselbe Computerschnittstelle betrieben werden.

Mercury-Netzwerke sind flexibel und auch zu einem späteren Zeitpunkt erweiterbar.

Flexible Automatisierung

Der C-863 bietet eine Reihe von Leistungsmerkmalen, die eine kostengünstige Durchführung von Automatisierungs- und Verarbeitungsaufgaben in Forschung und Industrie ermöglichen. Mit der leicht verständlichen Programmiersprache können Makros im nichtflüchtigen Speicher gespeichert werden.

Ein programmierbares Startup-Makro ermöglicht den Stand-Alone-Betrieb: Auch ohne externe Kommunikation erfolgt die automatische Ausführung interner Befehlsabläufe beim Einschalten.

Zur leichten Synchronisation von Bewegungsabläufen mit internen oder externen Ereignissen dienen jeweils vier I/O-Leitungen. Zur manuellen Bedienung kann ein Joystick angeschlossen werden.

Benutzerfreundlich: Umfangreiche Software und zwei Schnittstellen

Der Controller verfügt über eine USB-Schnittstelle zum bequemen Datenaustausch mit Laptop oder PC. Standardmäßig ist auch eine RS-232-Schnittstelle vorhanden.

Die mitgelieferte Software ermöglicht den vernetzten Betrieb mehrerer Controller. LabVIEW-Treiber und eine Programmbibliothek erleichtern Programmierern die Einbindung des Systems. Die Mercury-Controller sind mit dem Befehlssatz des PI General Command Set (GCS) direkt ansteuerbar. Mit PI GCS können unterschiedliche PI-Controller wie Piezosteuerungen und Servocontroller mit minimalem Programmieraufwand gemeinsam betrieben werden.

3.2 Modellübersicht

Zur Modellfamilie der Mercury-Controller gehört neben dem DC-Motor-Controller C-863.11 auch das Modell C-663.11 für Versteller mit Schrittmotor.

INFORMATION

Die Hardware des DC-Motor-Controllers C-863.11 ist identisch mit der Hardware des DC-Motor-Controllers C-863.10. Beide Modelle unterscheiden sich durch ihre Firmware und verwenden unterschiedliche Befehlssätze. C-863.10 und C-863.11 können nicht miteinander vernetzt werden.

Ein C-863.10 kann durch Installation der entsprechenden Firmware in einen C-863.11 umgewandelt werden.

- Wenn Sie einen C-863.10 in einen C-863.11 umwandeln wollen, wenden Sie sich an den Kundendienst (S. 309).
- Wenn Sie Software, die Sie für den C-863.10 geschrieben haben, mit C-863.11 nutzen wollen, lesen Sie die Technical Note A000T0029.
- Bei weiteren Fragen wenden Sie sich an unseren Kundendienst (S. 309).

3.3 Produktansicht

3.3.1 Vorderwand



Abbildung 1: C-863 Mercury DC-Motor Controller, Vorderansicht

Beschriftung	Typ	Funktion
RS-232 In	D-Sub 9(m) (S. 319)	Serielle Verbindung zum PC oder zum vorhergehenden Controller in einem Daisy-Chain-Netzwerk; nicht mit dem PC verbinden, wenn die USB-Schnittstelle bereits verbunden ist.
RS-232 Out	D-Sub 9(f) (S. 319)	Serielle Verbindung zum nachfolgenden Controller in einem Daisy-Chain-Netzwerk
	Mini-USB Typ B	Universal Serial Bus für Verbindung zum PC; nicht verbinden, wenn RS-232 In bereits verbunden ist.

Beschriftung	Typ	Funktion
STA	LED grün/aus	Controllerstatus: ■ Grün: C-863 ist bereit für den Normalbetrieb ■ Aus: C-863 ist nicht an der Versorgungsspannung angeschlossen oder befindet sich im Firmware-Update-Modus (Auswahl über DIP-Schalter 8)
ERR	LED rot/aus	Fehleranzeige: ■ Dauerhaftes Leuchten: Fehler (Fehlercode $\neq 0$) ■ Aus: Kein Fehler (Fehlercode = 0) Der Fehlercode kann mit dem Befehl ERR? abgefragt werden. Durch die Abfrage wird der Fehlercode auf null zurückgesetzt, und die LED wird ausgeschaltet.
Mode, Baud, Addr	8-Bit-DIP-Schalter (S. 74)	Einstellung der Geräteadresse, der Baudrate für die Kommunikation mit dem PC, der Endschaltersignallogik und des Update-Modus.

3.3.2 Rückwand



Abbildung 2: C-863 Mercury DC-Motor Controller, Rückansicht

Beschriftung	Typ	Funktion
15-30 VDC	Hohlstecker-Buchse (Eingang) (S. 320)	Anschluss für die Versorgungsspannung
I/O	Mini-DIN-Buchse, 9-polig (S. 316)	Digitale Ein-/Ausgänge: ■ Ausgänge: Triggerung externer Geräte ■ Eingänge: Verwendung in Makros oder als Schaltersignale Analoge Eingänge: ■ Verwendung in Makros oder für Scanvorgänge

Joystick	Mini-DIN-Buchse, 6-polig (S. 317)	Analoger Joystick ▪ Eingänge für Signale der Joystick-Achsen und –Tasten ▪ Ausgang für Versorgungsspannung des Joysticks
DC Motor only	D-Sub 15(f) (S. 315)	Anschluss des Verstellers. Nur für DC-Motoren! ▪ Ausgabe der PWM-Signale für den Versteller ▪ Eingang der Signale des inkrementellen Positionssensors ▪ Eingang der Signale der Endschalter und des Referenzschalters
	Schraube und Unterlegscheibe	Erdungsanschluss (S. 63) Wenn ein Potentialausgleich erforderlich ist, kann die Schraube mit dem Erdungssystem verbunden werden.

3.4 Lieferumfang

Bestellnummer	Komponenten
C-863.11	Mercury DC-Motor Controller
C-890.PS	Weitbereichsnetzteil für Mercury
3763	Netzkabel
C-815.34	RS-232-Nullmodemkabel, 3 m, 9/9-pol.
C-862.CN	Netzwerkkabel für Mercury, 30 cm
000014651	USB-Kabel (Typ A auf Mini-B) zur Verbindung mit dem PC
C-663.CD1	Mercury Produkt-CD mit Software und Benutzerhandbüchern für Mercury Produkte Die Mercury Produkt-CD gilt für den C-663.11 und den C-863.11.
MS205Dqu	Kurzversion des Benutzerhandbuchs für den C-863.11

3.5 Optionales Zubehör

Bestell-nummer	Beschreibung
C-815.38	Motorkabel, 3 m, D-Sub, 15-pol. (m/w)
C-862.CN2	Netzwerkkabel für Mercury, 180 cm
C-819.20	Analoger Joystick für Mercury Controller, 2 Achsen, Details siehe "Verfügbare Joystick-Geräte" (S. 129)
C-819.20Y	Y-Kabel für 2 Controller an Joystick C-819.20
C-819.30	Analoger Joystick für 3 Achsen, Details siehe "Verfügbare Joystick-Geräte" (S. 130)
C-170.PB	Pushbutton-Box mit 4 Tasten und 4 LEDs. Anschluss an Buchse I/O des C-863, liefert 4 TTL-Eingangssignale und zeigt den Status der 4 digitalen Ausgänge anhand der LEDs an. 
C-170.IO	I/O-Kabel, 2 m, offenes Ende (S. 316)

Wenden Sie sich bei Bestellungen an den Kundendienst (S. 309).

3.6 Funktionsprinzipien

3.6.1 Blockdiagramm

Der C-863 steuert die Bewegung der logischen Achse eines Verstellers. Das nachfolgende Blockdiagramm zeigt, wie der C-863 das Ausgangssignal für die verbundene Achse erzeugt:

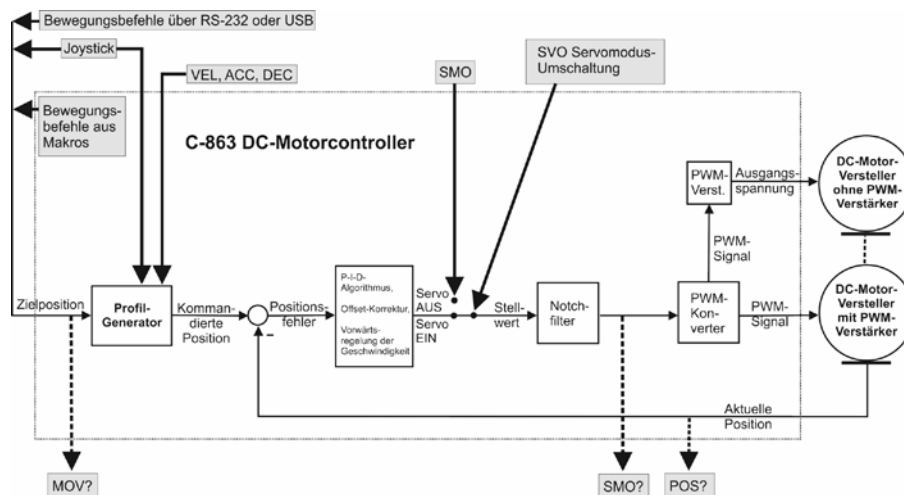


Abbildung 3: C-863.11: Erzeugung des Stellwerts

Der C-863 unterstützt sowohl Versteller mit PWM-Verstärker als auch Versteller ohne PWM-Verstärker (S. 64).

Die Höhe der Ausgangsspannung für DC-Motor-Versteller ohne PWM-Verstärker hängt von der Versorgungsspannung ab, die das an den C-863 angeschlossene Netzteil liefert.

3.6.2 Kommandierbare Elemente

Die folgende Tabelle enthält die mit den Befehlen des GCS (S. 166) kommandierbaren Elemente.

Element	Anzahl	Ken-nung	Beschreibung
Logische Achse	1	1 (änderbar)	Die logische Achse bildet die Bewegung des Verstellers in der Firmware des C-863 ab. Sie entspricht einer Achse eines linearen Koordinatensystems. Alle Befehle zur Bewegung eines Verstellers beziehen sich auf logische Achsen. In der Firmware des C-863 werden Bewegungen für logische Achsen kommandiert (d. h. für die Bewegungsrichtungen eines Verstellers). Im geregelten Betrieb stehen z. B. die Bewegungsbefehle MOV und MVR zur Verfügung. Der Bewegungsbefehl für den ungeregelten Betrieb lautet SMO. Die Achsenkennung kann mit dem Befehl SAI? abgefragt und mit dem Befehl SAI geändert werden. Sie kann aus bis zu 8 Zeichen bestehen; gültige Zeichen sind 1234567890ABCDEFGHIJKLMNOPQRSTUVWXYZ-_ Die neue Achsenkennung wird sofort im permanenten Speicher gespeichert und ist daher auch nach einem Neustart oder nach dem nächsten Einschalten noch verfügbar.
Analoge Eingänge	7	1 bis 7	Die analogen Eingangsleitungen mit den Kennungen 1 bis 4 sind die Eingänge 1 bis 4 der Buchse I/O (S. 316). Ihre Anzahl wird mit dem Befehl TAC? angezeigt, und ihre Werte können mit dem Befehl TAV? abgefragt werden. Beachten Sie, dass diese Leitungen auch als digitale Eingänge verwendet werden können (siehe unten). Weitere analoge Eingangsleitungen befinden sich an der Buchse Joystick (S. 317). Diese Leitungen werden nicht über TAC? und TAV? ausgegeben. Die Werte aller Eingänge können mittels der Aufzeichnungsoption 81 des Befehls DRC aufgenommen werden.
Digitale Ausgänge	4	1 bis 4	1 bis 4 kennzeichnen die digitalen Ausgangsleitungen 1 bis 4 der Buchse I/O (S. 316). Weitere Informationen siehe "Digitale Ausgangssignale" (S. 103).

Element	Anzahl	Kennung	Beschreibung
Digitale Eingänge	4	1 bis 4	1 bis 4 kennzeichnen die digitalen Eingangsleitungen 1 bis 4 der Buchse I/O (S. 316), die auch als analoge Eingänge verwendet werden können (siehe oben). Weitere Informationen siehe "Digitale Eingangssignale" (S. 113).
Joystick-Geräte	2	1, 2	An die Buchse Joystick (S. 317) des C-863 kann eine Achse eines Joystick-Geräts angeschlossen werden. Anschlussmöglichkeiten:
Joystick-Achsen pro Joystick-Gerät	1	1	Pin 4 (0 bis 3,3 V): Kommandierung als Achse 1 von Joystick-Gerät 1 oder Pin 2 (-10 bis 10 V): Kommandierung als Achse 1 von Joystick-Gerät 2
Joystick-Tasten	1	1	Eine Taste eines Joystick-Geräts kann an Pin 6 der Buchse Joystick (S. 317) angeschlossen werden: Kommandierung als Taste 1 von Joystick-Gerät 1. Weitere Informationen siehe "Joystick-Steuerung" (S. 121). Für die Konfiguration des Datenrekorders mit dem Befehl DRC gelten abweichend von den Angaben oben die folgenden Datenquellen-Kennungen: 5 = Achse 1 von Joystick-Gerät 1 6 = Taste 1 von Joystick-Gerät 1 7 = Achse 1 von Joystick-Gerät 2
Datenrekordertabelle	2	1, 2	Der C-863 hat 2 Datenrekordertabellen (Abfrage mit TNR?) mit 1024 Datenpunkten pro Tabelle.
Controlleradresse	1	1 bis 16	Die Controlleradresse kann im Bereich von 1 bis 16 mit den DIP-Schaltern auf der Vorderwand des C-863 gesetzt werden (S. 75). In einem Daisy-Chain-Netzwerk (S. 66) muss jeder Controller eine eindeutige Adresse haben (S. 158).

3.6.3 Wichtige Komponenten der Firmware

Die Firmware des C-863 stellt die folgenden funktionalen Einheiten bereit:

Firmware-Komponente	Beschreibung
ASCII-Befehle	Die Kommunikation mit dem C-863 kann mit den Befehlen des PI General Command Set (GCS; Version 2.0) geführt werden. Der GCS ist von der Hardware (Controller, angeschlossene Versteller) unabhängig. Beispiele für die Verwendung des GCS: ▪ C-863 konfigurieren ▪ Betriebsart einstellen ▪ Bewegungen des Verstellers starten ▪ System- und Positionswerte abfragen Eine Liste der verfügbaren Befehle finden Sie im Abschnitt "Befehlsübersicht" (S. 162).
Parameter	Parameter spiegeln die Eigenschaften des angeschlossenen Verstellers wider (z. B. Stellweg) und bestimmen das Verhalten des C-863 (z. B. Einstellungen für den Regelalgorithmus oder für die Verwendung der digitalen Eingänge). Parameterwerte können geändert werden, um das System an die jeweilige Anwendung anzupassen. Weitere Informationen finden Sie im Abschnitt "Anpassen von Einstellungen" (S. 279).
Profilgenerator und Regelalgorithmus	Im geregelten Betrieb errechnet ein Profilgenerator das Dynamikprofil. Der Positionsfehler, der sich aus der Differenz zwischen dem errechneten Dynamikprofil und der tatsächlichen Position (Rückmeldung des Sensors) ergibt, durchläuft einen P-I-D-Regelalgorithmus. Weitere Informationen finden Sie in den Abschnitten "Erzeugung des Dynamikprofils" (S. 26), "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 29) und "Auslösen von Bewegungen" (S. 25).
Datenrekorder	Der C-863 besitzt einen Echtzeit-Datenrekorder (S. 101). Der Datenrekorder kann verschiedene Signale (z. B. Position, analoger Eingang) aus verschiedenen Datenquellen (z. B. logische Achsen oder Eingangskanäle) aufzeichnen.
Makros	Der C-863 kann Makros (S. 131) speichern. Über die Makrofunktion können Befehlssequenzen festgelegt und dauerhaft im nichtflüchtigen Speicher des Geräts gespeichert werden. Ein Startup-Makro kann festgelegt werden, das bei jedem Einschalten oder Neustart des C-863 ausgeführt wird. Das Startup-Makro vereinfacht den Stand-Alone-Betrieb (Betrieb ohne Verbindung zum PC). Weitere Informationen finden Sie im Abschnitt "Controllermakros" (S. 131).

Die Firmware kann mit einem Hilfsprogramm aktualisiert werden (S. 298).

3.6.4 Betriebsarten

Der C-863 unterstützt die folgenden Betriebsarten:

Betriebsart	Beschreibung
Geregelter Betrieb (Servomodus Ein)	Ein Profilgenerator berechnet das Dynamikprofil aus den vorgegebenen Werten für Zielposition, Geschwindigkeit, Beschleunigung und Abbremsung. Der Positionsfehler, der sich aus der Differenz zwischen dem errechneten Dynamikprofil und der tatsächlichen Position (Rückmeldung des Sensors) ergibt, durchläuft einen P-I-D-Regelalgorithmus (p roportional- i ntegral- d ifferenzial). Außerdem können weitere Korrekturen erfolgen. Das Ergebnis ist der Stellwert für den im C-863 integrierten PWM-Konverter. Weitere Informationen finden Sie in den Abschnitten "Erzeugung des Dynamikprofils" (S. 26) und "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 29).
Ungeregelter Betrieb (Servomodus Aus)	Im ungeregelten Betrieb berechnet der C-863 kein Dynamikprofil und wertet nicht die Signale des Positionssensors aus. Daher kann der Versteller ungebremst an das Ende des Stellwegs fahren und trotz Endschaltefunktion auf dem mechanischen Anschlag aufprallen.

INFORMATION

Der C-863 ist für den geregelten Betrieb mit inkrementellen Positionssensoren vorgesehen (Servomodus Ein). Nach dem Einschalten ist standardmäßig der ungeregelte Betrieb aktiviert (Servomodus Aus).

- Fragen Sie die aktuelle Betriebsart mit den Befehlen `SV0?`, `#4` oder `SRG?` ab.
- Aktivieren Sie den geregelten Betrieb mit dem Befehl `SV0`.
- Wenn nötig, programmieren Sie ein Startup-Makro, das den C-863 über den Befehl `SV0` im geregelten Betrieb startet; siehe "Startup-Makro einrichten" (S. 142).
- Vermeiden Sie Bewegungen im ungeregelten Betrieb.

3.6.5 Physikalische Einheiten

Der C-863 unterstützt verschiedene Längeneinheiten für Positionsangaben. Die Anpassung erfolgt durch einen Faktor, mit dem die Impulse des inkrementellen Encoders in die gewünschte physikalische Längeneinheit umgerechnet werden. Der Umrechnungsfaktor wird mit folgenden Parametern eingestellt:

Parameter	Beschreibung und mögliche Werte
Numerator Of The Counts-Per-Physical-Unit Factor 0xE	Zähler und Nenner des Faktors für Impulse pro physikalischer Längeneinheit 1 bis 1.000.000 für jeden Parameter. Der Faktor für die Impulse pro physikalische Längeneinheit bestimmt die Längeneinheit für Positionsabfragen und Bewegungsbefehle im geregelten Betrieb.
Denominator Of The Counts-Per-Physical-Unit Factor 0xF	An den eingestellten Faktor werden automatisch die Werte aller Parameter angepasst, deren Einheit entweder die physikalische Längeneinheit selbst oder eine darauf basierende Maßeinheit ist. Der Faktor für die Impulse pro physikalische Längeneinheit hat keinen Einfluss auf die Stabilität des Regelkreises, wird aber für die Eingangs- und Ausgangsskalierung von Positionswerten verwendet.

Das Einheitenzeichen kann für Anzeigezwecke mit folgendem Parameter angepasst werden:

Parameter	Beschreibung und mögliche Werte
Axis Unit 0x07000601	Einheitenzeichen Maximal 20 Zeichen. Das Einheitenzeichen ist z. B. "MM", wenn der Faktor für die Impulse pro physikalische Längeneinheit mit den Parametern 0xE und 0xF so eingestellt ist, dass die Encoderimpulse in Millimeter umwandelt werden. Das Einheitenzeichen für Rotationstische lautet normalerweise "deg". Der Wert des Parameters 0x07000601 wird nicht vom C-863 ausgewertet, sondern nur von der PC-Software für Anzeigezwecke genutzt. Beispiele: 1 Encoderimpuls = 100 nm Impulse pro physikalischer Längeneinheit: 10000:1 → Einheitenzeichen: mm 1 Encoderimpuls = 0,254 mm Impulse pro physikalischer Längeneinheit: 100:1 → Einheitenzeichen: Zoll

3.6.6 Auslösen von Bewegungen

Bewegungen im geregelten Betrieb

Auslöser der Bewegung	Befehle	Beschreibung
Bewegungsbefehle, gesendet von der Befehlszeile oder durch die PC-Software	MOV, MVR	Bewegung zu absoluter oder relativer Zielposition
	GOH	Bewegung zur Nullposition
	STE	Startet einen Schritt und zeichnet die Sprungantwort auf
	FNL, FPL, FRF	Starten von Referenzfahrten
	FED	Starten von Fahrten zu Signalfanken
Controllermakros mit Bewegungsbefehlen	MAC Weitere Makrobefehle und Informationen siehe "Controllermakros" (S. 131).	Ruft eine Makrofunktion auf. Erlaubt das Aufzeichnen, Löschen und Ausführen von Makros auf dem Controller. Von der Befehlszeile können sämtliche Befehle gesendet werden, während auf dem Controller ein Makro läuft. Der Makroinhalt und Bewegungsbefehle, die von der Befehlszeile empfangen werden, können sich gegenseitig überschreiben.
Joystick-Steuerung	JON JAX Weitere Joystick-Befehle siehe "Joystick-Steuerung" (S. 121).	Aktiviert oder deaktiviert ein Joystick-Gerät, das an den Controller angeschlossen ist. Bestimmt die Achse, die durch einen an den Controller angeschlossenen Joystick gesteuert wird. Der Joystick steuert die Geschwindigkeit der Achse (vom Profilgenerator ausgegebene kommandierte Geschwindigkeit). Bewegungsbefehle sind nicht zulässig, wenn ein Joystick für die Achse aktiv ist.

INFORMATION

Absolute Zielpositionen können nur kommandiert werden, wenn zuvor der Referenzwert der Achse bestimmt wurde; siehe "Referenzwertbestimmung" (S. 40).

Bewegungen im ungeregelten Betrieb

Bewegungen werden mit dem Befehl **SM0** ausgelöst, der den Stellwert für den PWM-Konverter im C-863 direkt festlegt.

Im ungeregelten Betrieb ist keine Joystick-Steuerung möglich.

3.6.7 Erzeugung des Dynamikprofils

Im geregelten Betrieb führt der Profilgenerator Berechnungen durch, um die Zielposition, Geschwindigkeit und Beschleunigung der Achse zu jedem Zeitpunkt der Bewegung festzulegen (Dynamikprofil). Die errechneten Werte werden kommandierte Werte genannt. Das vom Profilgenerator des C-863 erzeugte Dynamikprofil hängt von den Bewegungsgrößen ab, die durch Befehle (S. 166), Parameter und/oder per Joystick vorgegeben werden:

Bewegungsgröße	Befehle	Parameter	Anmerkungen
Beschleunigung (A)	ACC ACC?	Beschleunigung im geregelten Betrieb (Parameter 0xB; physikalische Längeneinheit/s ²); Änderung mit dem Befehl ACC oder mit SPA / SEP ; kann mit WPA gespeichert werden.	Wird durch Parameter 0x4A (Maximale Beschleunigung im geregelten Betrieb) begrenzt
Abbremsung (D)	DEC DEC?	Abbremsung im geregelten Betrieb (Parameter 0xC; physikalische Längeneinheit/s ²); Änderung mit dem Befehl DEC oder mit SPA / SEP ; kann mit WPA gespeichert werden.	Wird durch Parameter 0x4B (Maximale Abbremsung im geregelten Betrieb) begrenzt
Geschwindigkeit (V)	VEL VEL?	Geschwindigkeit im geregelten Betrieb (Parameter 0x49; physikalische Längeneinheit/s); Änderung mit dem Befehl VEL oder mit SPA / SEP ; kann mit WPA gespeichert werden.	Wird durch Parameter 0xA (Maximale Geschwindigkeit im geregelten Betrieb) begrenzt Bei Anschluss eines Joysticks an den C-863 und Aktivierung des Joysticks mit dem Befehl JON wird ein Faktor auf die aktuelle, mit dem Befehl VEL eingestellte Geschwindigkeit angewendet. Weitere Informationen siehe "Joystick-Steuerung" (S. 121).

Bewegungsgröße	Befehle	Parameter	Anmerkungen
Zielposition am Ende der Bewegung	MOV MVR GOH STE	-	Bei Anschluss eines Joysticks an den C-863 und Aktivierung des Joysticks mit dem Befehl JON werden die Verfahrbereichsgrenzen als jeweilige Zielposition eingestellt. Bei Deaktivierung des Joystick-Geräts wird die Zielposition auf die aktuelle Position für joystickgesteuerte Achsen eingestellt. Weitere Informationen siehe "Joystick-Steuerung" (S. 121). Beim Einschalten des Servomodus mit dem Befehl SVO oder beim Anhalten der Achsenbewegung mit den Befehlen #24, STP oder HLT wird die Zielposition auf die aktuelle Position eingestellt.

Der Profilgenerator des C-863 unterstützt ausschließlich trapezförmige Geschwindigkeitsprofile: Die Achse beschleunigt linear (auf Basis des vorgegebenen Beschleunigungswerts), bis sie die vorgegebene Geschwindigkeit erreicht. Sie bewegt sich weiter mit dieser Geschwindigkeit, bis sie linear (auf Basis des vorgegebenen Abbremsungswerts) abbrems und an der vorgegebenen Zielposition anhält.

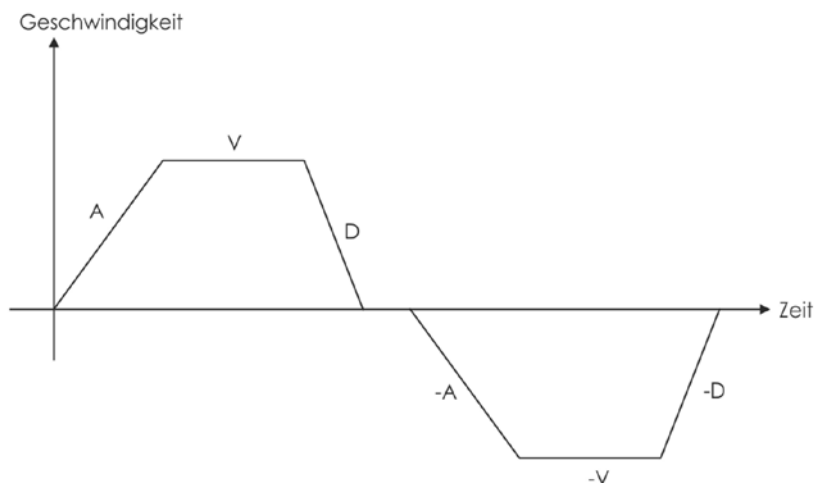


Abbildung 4: Einfaches trapezförmiges Geschwindigkeitsprofil; A = Beschleunigung, D = Abbremsung, V = Geschwindigkeit

Wenn die Abbremsung beginnen muss, bevor die Achse die vorgegebene Geschwindigkeit erreicht, wird das Profil keinen konstanten Geschwindigkeitsanteil haben, und aus dem Trapez wird ein Dreieck.

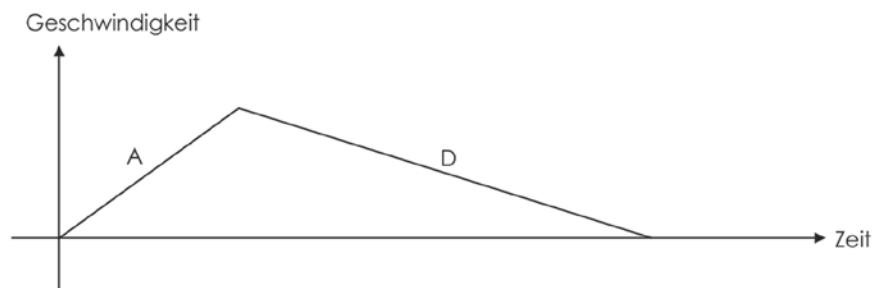


Abbildung 5: Einfaches trapezförmiges Geschwindigkeitsprofil; A = Beschleunigung, D = Abbremsung; keine konstante Geschwindigkeit

Die Flanken für Beschleunigung und Abbremsung können symmetrisch (Beschleunigung = Abbremsung) oder asymmetrisch (Beschleunigung $\neq$ Abbremsung) sein. Der Beschleunigungswert wird immer zu Beginn der Bewegung angewandt. Danach werden der Beschleunigungswert bei Zunahme der absoluten Geschwindigkeit und der Abbremsungswert bei Abnahme der absoluten Geschwindigkeit verwendet. Werden während der Bewegung keine Bewegungsgrößen verändert, so wird der Beschleunigungswert bis zum Erreichen der maximalen Geschwindigkeit verwendet, und der Abbremsungswert wird für die Verringerung der Geschwindigkeit bis auf null verwendet.

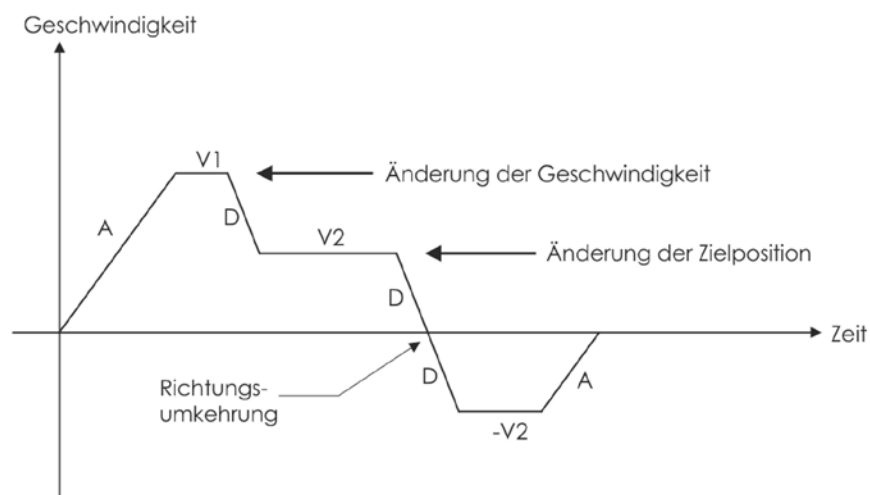


Abbildung 6: Komplexes trapezförmiges Profil mit Parameteränderungen; A = Beschleunigung; D = Abbremsung; V1, V2, -V2 = Geschwindigkeiten

Alle Bewegungsgrößen können geändert werden, während die Achse in Bewegung ist. Der Profilgenerator wird immer versuchen, innerhalb der zulässigen Bewegungsgrenzen zu bleiben, die durch die Bewegungsgrößen vorgegeben sind. Wenn die Zielposition während der Bewegung so geändert wird, dass ein Überschwingen unvermeidlich ist, wird der Profilgenerator bis zum Stillstand abbremsen und die Bewegungsrichtung umkehren, um die vorgegebene Position zu erreichen.

3.6.8 Regelalgorithmus und weitere Stellwertkorrekturen

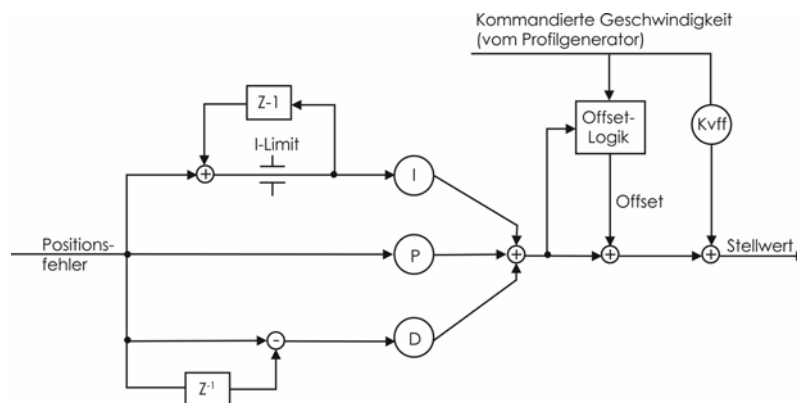


Abbildung 7: P-I-D-Algorithmus, Offset-Korrektur und Vorwärtsregelung der Geschwindigkeit (KVff); hier nicht dargestellt ist der Notchfilter

Im geregelten Betrieb wird der Stellwert für den im C-863 integrierten PWM-Konverter und damit das Einschwingverhalten des Systems durch folgende Korrekturen optimiert:

- Regelalgorithmus: Der Positionsfehler, der sich aus der Differenz zwischen dem errechneten Dynamikprofil (siehe "Erzeugung des Dynamikprofils" (S. 26)) und der tatsächlichen Position (Rückmeldung des Sensors) ergibt, durchläuft einen P-I-D-Regelalgorithmus (**p**roportional-**i**ntegral-**d**ifferential).
- Dynamikprofilkorrekturen: Das errechnete Dynamikprofil kann einer Offset-Korrektur und einer Vorwärtsregelung der Geschwindigkeit unterzogen werden.

Unabhängig von der Betriebsart kann der Stellwert durch den Notchfilter einer weiteren Korrektur unterzogen werden.

Regelalgorithmus

Der Regelalgorithmus verwendet die folgenden Regelparameter. Die optimale Einstellung der Regelparameter hängt von Ihrer Anwendung und Ihren Wünschen ab; siehe "Regelparameter optimieren" (S. 96).

Parameter	Beschreibung und mögliche Werte
P-Term 0x1	Proportionalkonstante (dimensionslos) 0 bis 32767 Ziel: Schnelle Korrektur des Positionsfehlers
I-Term 0x2	Integrationskonstante (dimensionslos) 0 bis 32767 Ziel: Reduzierung des statischen Positionsfehlers
D-Term 0x3	Differentialkonstante (dimensionslos) 0 bis 32767 Ziel: Dämpfung schneller Regelschwingungen
I-Limit 0x4	Begrenzung der Integrationskonstante (dimensionslos) 0 bis 32767
D-Term Delay (No. Of Servo Cycles) 0x71	D-Term-Verzögerung Der D-Term kann als fließender Mittelwert über mehrere Servozyklen berechnet werden. Der Parameter legt fest, wie viele Werte (d.h. Servozyklen) für die Mittelwertberechnung verwendet werden sollen.

INFORMATION

Aus Kompatibilitätsgründen besitzt der C-863 folgenden zusätzlichen Satz von Regelparametern:

- Parameter 0x411, 0x412, 0x413, 0x414

Die Werte dieser Parameter werden automatisch auf die Werte der tatsächlich verwendeten Regelparameter 0x1, 0x2, 0x3 und 0x4 gesetzt.

- Verwenden Sie für die Optimierung des Einschwingverhaltens des Systems nur die Parameter 0x1, 0x2, 0x3 und 0x4; siehe "Regelparameter optimieren" (S. 96).

Mit folgenden Parametern kann der Eingang des Regelalgorithmus des C-863 konfiguriert werden:

Parameter	Beschreibung und mögliche Werte
Numerator Of The Servo-Loop Input Factor 0x5A	Zähler und Nenner des Eingangsfaktors des Regelkreises 1 bis 1.000.000 für beide Parameter Der Eingangsfaktor des Regelkreises entkoppelt die Regelparameter von der Encoderauflösung.
Denominator Of The Servo-Loop Input Factor 0x5B	Der Eingangsfaktor des Regelkreises ist unabhängig vom Faktor für die Impulse pro physikalische Längeneinheit (0xE und 0xF). Zähler und Nenner des Eingangsfaktors des Regelkreises sollten nicht geändert werden.

Korrekturen des Dynamikprofils

Die Korrekturen des Dynamikprofils für den geregelten Betrieb können durch die nachfolgend aufgelisteten Parameter konfiguriert werden:

Parameter	Beschreibung und mögliche Werte
Motor Offset Positive 0x33	Offset für die positive Bewegungsrichtung (dimensionslos). 0 bis 32766
Motor Offset Negative 0x34	Offset für die negative Bewegungsrichtung (dimensionslos). 0 bis 32766
Motor Drive Offset 0x48	Geschwindigkeitsabhängiger Offset (dimensionslos). Wird angewendet, wenn die kommandierte Geschwindigkeit ungleich null ist (d.h. wenn das Ende des Dynamikprofils noch nicht erreicht ist). 0 bis 32766
Kvff 0x5	Vorwärtsregelung der kommandierten Geschwindigkeit 0 bis 32767 Ziel: Minimierung des Positionsfehlers

Betriebsartunabhängige Stellwertkorrekturen

Die nachfolgend aufgelisteten Parameter korrigieren den Stellwert sowohl im geregelten als auch im ungeregelten Betrieb.

Parameter	Beschreibung und mögliche Werte
Notch Filter Frequency 1 (Hz) 0x94	Frequenz des ersten Notchfilters 40 bis 20000 Hz Der entsprechende Frequenzanteil im Stellwert wird reduziert, um unerwünschte Resonanzen in der Mechanik zu kompensieren. Eine Anpassung kann vor allem bei sehr hohen Lasten sinnvoll sein.
Notch Filter Edge 1 0x95	Anstieg der Flanke des ersten Notchfilters (dimensionslos) 0,1 bis 10 Dieser Parameterwert sollte nicht geändert werden.

3.6.9 On-Target-Status

Im geregelten Betrieb kann das Erreichen der Zielposition anhand des On-Target-Status überprüft werden:

- On-Target-Status = wahr (1): die Zielposition gilt als erreicht
- On-Target-Status = falsch (0): die Zielposition ist nicht erreicht

Der C-863 ermittelt den On-Target-Status anhand folgender Kriterien:

- Einschwingfenster um die Zielposition (Parameter 0x36)
- Verzögerungszeit für Setzen des On-Target-Status (Parameter 0x3F)

Der On-Target-Status nimmt in folgenden Fällen den Wert **wahr** an:

- Die aktuelle Position ist im Einschwingfenster und bleibt dort mindestens für die Dauer der Verzögerungszeit.
- Wenn der Wert für die Verzögerungszeit auf 0 gesetzt ist: Das Ende des Dynamikprofils ist erreicht.

Der On-Target-Status kann mit den Befehlen **ONT?**, **#4** und **SRG?** ausgelesen werden.

Im Triggermodus *On Target* (S. 108) wird der On-Target-Status der gewählten Achse am gewählten Triggerausgang ausgegeben.

Parameter	Beschreibung und mögliche Werte
Settling Time (s) 0x3F	Verzögerungszeit für das Setzen des On-Target-Status 0 bis 1,000 s
Settling Window (encoder counts) 0x36	Einschwingfenster um die Zielposition 0 bis 2^{31} Impulse des inkrementellen Encoders Gibt die Fenstergrenzen vor. Wenn die aktuelle Position das Einschwingfenster verlässt, gilt die Zielposition nicht mehr als erreicht. Der Parameterwert entspricht der Hälfte der Fensterbreite. Er kann nur geändert werden, wenn der Servomodus ausgeschaltet ist.

3.6.10 Referenzschaltererkennung

Der C-863 nimmt auf Pin 13 der Buchse **DC Motor only** (S. 315) das Signal eines Referenzschalters entgegen.

Mit folgenden Parametern kann die Referenzschaltererkennung durch den C-863 konfiguriert werden:

Parameter	Beschreibung und mögliche Werte
Invert Reference? 0x31	Soll das Referenzsignal invertiert werden? 0 = Referenzsignal nicht invertiert 1 = Referenzsignal invertiert Dieser Parameter dient zur Invertierung des Referenzsignals, dessen Quelle entweder der Referenzschalter oder ein digitaler Eingang sein kann, der anstelle des Referenzschalters verwendet wird (S. 117).
Has Reference? 0x14	Hat der Versteller einen Referenzschalter? 0 = Kein Referenzschalter eingebaut 1 = Referenzschalter vorhanden (Signaleingang am D-Sub 15 (f) Motoranschluss) Dieser Parameter aktiviert bzw. deaktiviert Referenzfahrten zum eingebauten Referenzschalter.

Parameter	Beschreibung und mögliche Werte
Reference Signal Type 0x70	Art des Referenzsignals 0 = richtungserkennender Referenzschalter (Standard-Einstellung). Der Signalpegel ändert sich beim Überfahren des Referenzschalters. 1 = Pulssignal mit einer Pulsweite von mehreren Nanosekunden (Parameter 0x47 muss korrekt gesetzt sein).

Das Signal des Referenzschalters des Verstellers kann für Referenzfahrten verwendet werden. Nach einer Referenzfahrt zum Referenzschalter kennt der Controller die absolute Achsenposition; siehe "Referenzwertbestimmung" (S. 40).

3.6.11 Endschaltererkennung

Der C-863 nimmt auf der Buchse **DC Motor only** (S. 315) Endschaltersignale entgegen:

- Pin 5: positiver Endschalter
- Pin 12: negativer Endschalter

Mit folgenden Parametern kann die Endschaltererkennung durch den C-863 konfiguriert werden:

Parameter	Beschreibung und mögliche Werte
Limit Mode 0x18	Signallogik der Endschalter 0 = Positiver Endschalter high-aktiv (pos-HI), Negativer Endschalter high-aktiv (neg-HI) 1 = Positiver Endschalter low-aktiv (pos-LO), neg-HI 2 = pos-HI, neg-LO 3 = pos-LO, neg-LO Der C-863 kann außerdem über den DIP-Schalter 7 an den Logikpegel der Endschalter des angeschlossenen Verstellers angepasst werden (high-aktiv oder low-aktiv). Damit sich der Versteller bewegen kann, müssen die Einstellungen des DIP-Schalters und des Parameters mit dem Endschalter-Logikpegel übereinstimmen (S. 76).

Parameter	Beschreibung und mögliche Werte
<i>Has No Limit Switches?</i> 0x32	Hat der Versteller Endschalter? 0 = Versteller hat Endschalter (Signaleingänge am D-Sub 15 (f) Motoranschluss) 1 = Versteller hat keine Endschalter Dieser Parameter aktiviert bzw. deaktiviert das Anhalten der Bewegung an den eingebauten Endschaltern.

Die Signale der Endschalter (auch Endlagensensoren) eines Linearverstellers werden verwendet, um die Bewegung vor dem mechanischen Anschlag an beiden Enden des Stellwegs anzuhalten. Da die eingestellte Abbremsung dabei nicht berücksichtigt wird, besteht bei hohen Geschwindigkeiten die Gefahr, dass der Versteller trotzdem auf dem mechanischen Anschlag aufprallt. Um dies zu vermeiden, können Verfahrbereichsgrenzen (S. 35) über Parameter des C-863 eingestellt werden.

Die Endschaltersignale können auch für Referenzfahrten verwendet werden. Nach einer Referenzfahrt zu einem Endschalter kennt der Controller die absolute Achsenposition; siehe "Referenzwertbestimmung" (S. 40).

3.6.12 Stellweg und Verfahrbereichsgrenzen

Folgende Parameter des C-863 spiegeln den physikalischen Stellweg des Verstellers wider und definieren Verfahrbereichsgrenzen:

Parameter	Beschreibung und mögliche Werte
<i>Maximum Travel In Positive Direction (Phys. Unit)</i> 0x15	Verfahrbereichsgrenze in positiver Richtung (physikalische Einheit) Bezogen auf die Nullposition. Wenn dieser Wert kleiner als der Positionswert für den positiven Endschalter ist (welcher sich aus der Summe der Parameter 0x16 und 0x2F ergibt), kann der positive Endschalter nicht für Referenzfahrten verwendet werden. Der Wert kann negativ sein.
<i>Value At Reference Position (Phys. Unit)</i> 0x16	Positionswert am Referenzschalter (physikalische Einheit) Die aktuelle Position wird auf diesen Wert gesetzt, wenn die Achse eine Referenzfahrt zum Referenzschalter ausgeführt hat (Start mit FRF). Der Parameterwert wird außerdem zur Berechnung der Positionswerte verwendet, die nach Referenzfahrten zu den Endschaltern gesetzt werden; dies gilt auch, wenn die Mechanik keinen Referenzschalter hat.

Parameter	Beschreibung und mögliche Werte
<i>Distance From Negative Limit To Reference Position (Phys. Unit)</i> 0x17	Abstand zwischen Referenzschalter und negativem Endschalter (physikalische Einheit) Wenn die Achse eine Referenzfahrt zum negativen Endschalter ausgeführt hat (Start mit FNL), wird die aktuelle Position auf die Differenz zwischen den Werten der Parameter 0x16 und 0x17 gesetzt.
<i>Distance From Reference Position To Positive Limit (Phys. Unit)</i> 0x2F	Abstand zwischen Referenzschalter und positivem Endschalter (physikalische Einheit) Wenn die Achse eine Referenzfahrt zum positiven Endschalter ausgeführt hat (Start mit FPL), wird die aktuelle Position auf die Summe der Werte der Parameter 0x16 und 0x2F gesetzt.
<i>Maximum Travel In Negative Direction (Phys. Unit)</i> 0x30	Verfahrbereichsgrenze in negativer Richtung (physikalische Einheit) Bezogen auf die Nullposition. Wenn dieser Wert größer als der Positionswert für den negativen Endschalter ist (welcher sich aus der Differenz der Parameter 0x16 und 0x17 ergibt), kann der negative Endschalter nicht für Referenzfahrten verwendet werden. Der Wert kann negativ sein.
<i>Range Limit Min</i> 0x07000000	Zusätzliche Verfahrbereichsgrenze für die negative Bewegungsrichtung (physikalische Einheit) Wenn die aktuelle Position diesen Wert im geregelten oder ungeregelten Betrieb erreicht, wird der Stellwert auf null gesetzt und dadurch die Bewegung angehalten. Sobald der Wert für die Verfahrbereichsgrenze verringert wurde, kann die Achse wieder bewegt werden.
<i>Range Limit Max</i> 0x07000001	Zusätzliche Verfahrbereichsgrenze für die positive Bewegungsrichtung (physikalische Einheit) Wenn die aktuelle Position diesen Wert im geregelten oder ungeregelten Betrieb erreicht, wird der Stellwert auf null gesetzt und dadurch die Bewegung angehalten. Sobald der Wert für die Verfahrbereichsgrenze vergrößert wurde, kann die Achse wieder bewegt werden.

INFORMATION

Der C-863 unterstützt zwei Parameterpaare zur Festlegung von Verfahrbereichsgrenzen. Sie sind für unterschiedliche Einsatzzwecke vorgesehen:

- 0x15 (**Maximum Travel In Positive Direction (Phys. Unit)**) und 0x30 (**Maximum Travel In Negative Direction (Phys. Unit)**):
 - Die Grenzen legen den erlaubten Verfahrbereich im geregelten Betrieb fest.
 - Bewegungsbefehle werden nur ausgeführt, wenn die kommandierte Position innerhalb dieser Verfahrbereichsgrenzen liegt.
 - Die Grenzen beziehen sich immer auf die aktuelle Nullposition.
 - Passende Werte werden bei der Auswahl des Verstellertyps aus der Verstellerdatabank geladen.
- 0x07000000 (**Range Limit Min**) und 0x07000001 (**Range Limit Max**):
 - Die Verwendung dieser Grenzen ist nur dann empfohlen, wenn unregelmäßige Bewegungen erforderlich sind. Dabei liegen die Werte sinnvollerweise außerhalb der Verfahrbereichsgrenzen, die durch 0x15 und 0x30 festgelegt sind.
 - Gelten sowohl im geregelten als auch im unregelmäßigen Betrieb.
 - Bewegungen werden abrupt gestoppt, wenn die aktuelle Position eine Grenze erreicht.
 - Die Grenzen sind unabhängig von der aktuellen Nullposition.
 - Die Werte werden nicht aus der Verstellerdatabank geladen und sind in der Werkseinstellung so gesetzt, dass die Grenzen deaktiviert sind.

Beispiele

Die nachfolgenden Beispiele beziehen sich auf eine Achse eines Verstellers mit inkrementellem Sensor, Referenzschalter und Endschaltern.

Der Abstand zwischen negativem und positivem Endschalter der Achse beträgt 20 mm. Der Referenzschalter hat zum negativen Endschalter 8 mm Abstand und zum positiven Endschalter 12 mm Abstand.

Diese Schalteranordnung der Achse spiegelt sich in folgenden Parametern wider:

- Parameter 0x17: Abstand zwischen negativem Endschalter und Referenzschalter = 8 mm
- Parameter 0x2F: Abstand zwischen Referenzschalter und positivem Endschalter = 12 mm

INFORMATION

Der Schalteranordnung der Achse kann mit den Befehlen **FED** und **POS?** ermittelt werden.

Beispiel 1: Maximaler Stellweg verfügbar

Nach Referenzfahrten (S. 40) soll die aktuelle Position folgende Werte haben:

- Fahrt zum negativen Endschalter (Start mit **FNL**): aktuelle Position = 0
- Fahrt zum Referenzschalter (Start mit **FRF**): aktuelle Position = 8
- Fahrt zum positiven Endschalter (Start mit **FPL**): aktuelle Position = 20

Deshalb hat Parameter 0x16, der bei Referenzfahrten den Positionswert am Referenzschalter bestimmt und in die Berechnung der Positionswerte an den Endschaltern eingeht, den Wert 8.

Der Stellweg soll nicht durch Verfahrbereichsgrenzen beschränkt werden. Deshalb werden die entsprechenden Parameter wie folgt gesetzt:

- Parameter 0x15 = 20
- Parameter 0x30 = 0

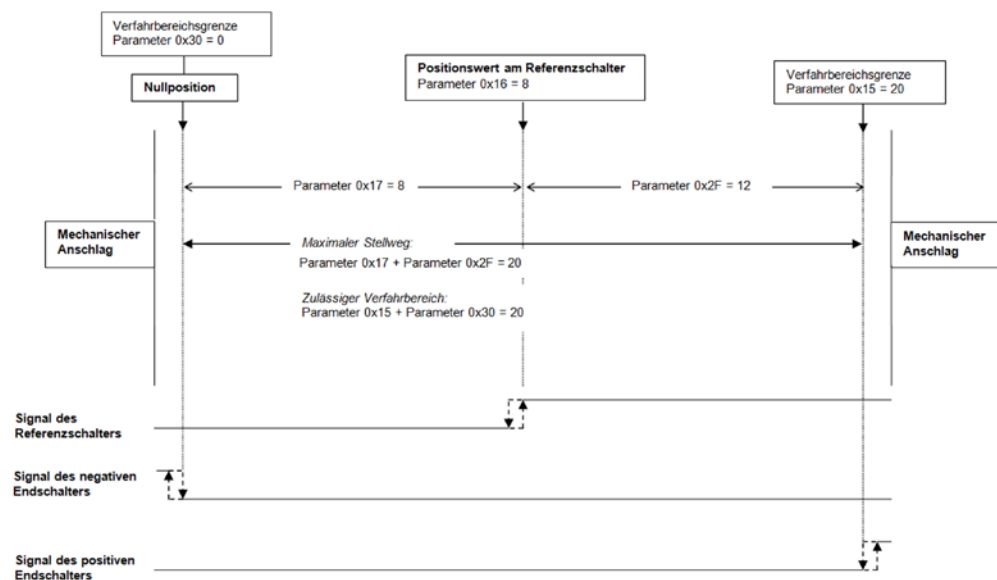


Abbildung 8: Der Stellweg der Achse wird nicht durch Verfahrbereichsgrenzen beschränkt.

Nach einer Referenzfahrt der Achse zum Referenzschalter (Befehl **FRF**) liefern Abfragebefehle folgende Antworten:

- **TMN?** liefert den Wert 0
- **TMX?** liefert den Wert 20
- **POS?** liefert den Wert 8

Beispiel 2: Stellweg durch Verfahrbereichsgrenzen beschränkt

Die Nullposition soll sich ungefähr bei einem Drittel des Abstands zwischen dem negativen Endschalter und dem Referenzschalter befinden. Parameter 0x16 hat deshalb nun den Wert 5,4.

An beiden Enden des Stellwegs soll durch Festlegung von Verfahrbereichsgrenzen ein Sicherheitsabstand eingerichtet werden. Die Verfahrbereichsgrenzen werden deshalb wie folgt eingestellt:

- Parameter 0x15 = 16,4
- Parameter 0x30 = -2,1

Demnach kann die Achse jeweils ab der Nullposition 16,4 mm in positiver Richtung und 2,1 mm in negativer Richtung fahren. Die Endschalter können nicht mehr für Referenzfahrten verwendet werden.

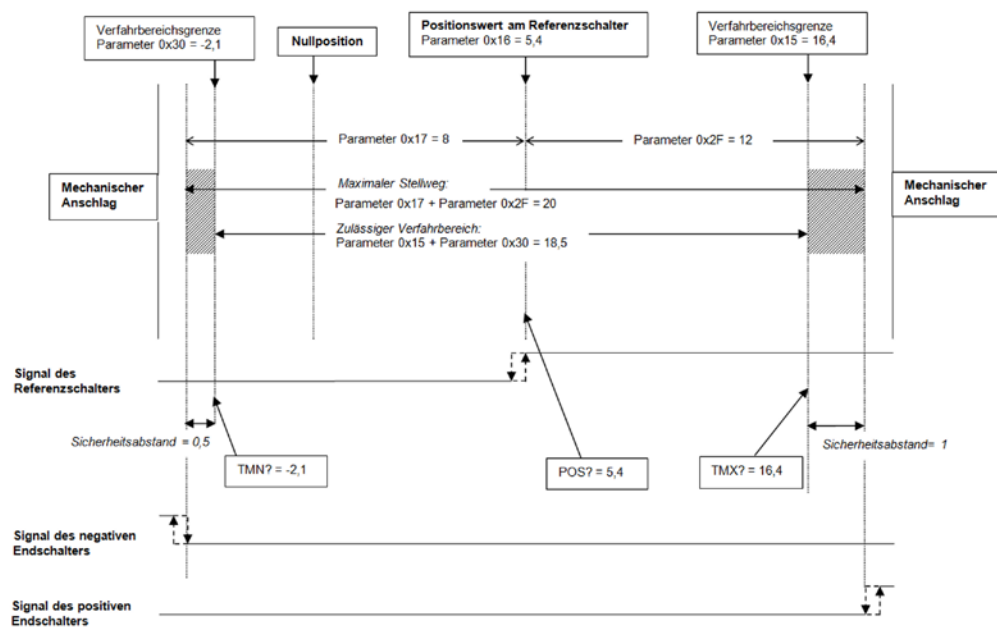


Abbildung 9: Der Stellweg der Achse wird durch Verfahrbereichsgrenzen beschränkt.

Nach einer Referenzfahrt der Achse zum Referenzschalter (Befehl **FRF**) liefern Abfragebefehle folgende Antworten:

- **TMN?** liefert den Wert -2,1
- **TMX?** liefert den Wert 16,4
- **POS?** liefert den Wert 5,4

3.6.13 Referenzwertbestimmung

Die inkrementellen Sensoren, die für die Rückmeldung der Achsenposition verwendet werden, liefern nur relative Bewegungsinformationen. Der Controller kennt deshalb die absolute Position einer Achse beim Einschalten nicht. Damit absolute Zielpositionen kommandiert und erreicht werden können, muss zuvor eine Referenzwertbestimmung durchgeführt werden.

Die Referenzwertbestimmung kann auf unterschiedliche Weise durchgeführt werden:

- **Referenzfahrt** (Standard): Eine Referenzfahrt bewegt die Achse zu einem fest definierten Punkt, z.B. zum Referenzschalter oder zu einem Endschalter. An diesem Punkt wird die aktuelle Position auf einen definierten Wert gesetzt. Der Controller kennt nun die absolute Achsenposition.
- **Manuelle Festlegung der absoluten Position:** Wenn diese Art der Referenzwertbestimmung mit dem Befehl **RON** (S. 231) aktiviert wurde, können Sie mit dem Befehl **POS** (S. 229) die aktuelle Position der Achse an einem beliebigen Punkt auf einen beliebigen Wert setzen. Dabei wird die Achse nicht bewegt. Der Controller kennt anschließend die absolute Achsenposition.

INFORMATION

Bei der Inbetriebnahme mit PIMikroMove® erfolgt die Referenzwertbestimmung standardmäßig durch eine Referenzfahrt. Die Kenntnis der hier beschriebenen Befehle und Parameter ist für die Referenzwertbestimmung mit PIMikroMove® nicht erforderlich.

INFORMATION

Um eine möglichst hohe Wiederholgenauigkeit der Referenzwertbestimmung zu erreichen, besteht jede Referenzfahrt aus folgenden Abschnitten:

1. Erste Fahrt zum ausgewählten Schalter. Die maximale Geschwindigkeit wird durch Parameter 0x49 vorgegeben (**Closed-Loop Velocity (Phys. Unit/s)**, entspricht der Einstellung mit dem Befehl VEL).
2. Stopp beim Erreichen der Schalterflanke. Je höher die Geschwindigkeit beim Anfahren war, desto weiter überfährt die Achse die Schalterflanke (Überschwingen).
3. Fahrt in die entgegengesetzte Richtung zum Ausgleichen des Überschwingens.
4. Zweite Fahrt zum ausgewählten Schalter. Die maximale Geschwindigkeit wird durch Parameter 0x50 vorgegeben (**Velocity For Reference Moves (Phys. Unit/s)**, spezielle Geschwindigkeitsvorgabe nur für Referenzfahrten).
5. Stopp beim Erreichen der Schalterflanke.
6. Fahrt in die entgegengesetzte Richtung zum Ausgleichen des Überschwingens.
7. Setzen der aktuellen Position auf einen definierten Wert, die Referenzwertbestimmung ist beendet.

Je geringer die Geschwindigkeit beim Anfahren des Schalters ist, desto geringer ist das Überschwingen und desto höher die Wiederholgenauigkeit. Deshalb sollte der Wert des Parameters 0x50 maximal so groß sein wie der Wert des Parameters 0x49, idealerweise jedoch wesentlich kleiner.

Die tatsächlichen Geschwindigkeiten während der Referenzfahrt werden aus den Werten folgender Parameter berechnet und können niedriger ausfallen als die Maximalwerte:

- Parameter 0x49 oder 0x50
- Parameter 0x63 (**Distance Between Limit And Hard Stop (Phys. Unit)**)
- Parameter 0xC (**Closed-Loop Deceleration (Phys. Unit/s²)**)

Befehle

Folgende Befehle stehen für die Referenzwertbestimmung zur Verfügung:

Befehl	Syntax	Funktion
RON	RON {<AxisID> <ReferenceOn>}	Bestimmt den Modus der Referenzwertbestimmung: ▪ <ReferenceOn> = 0: Für die Referenzwertbestimmung der Achse kann ein absoluter Positionswert mit POS zugewiesen werden, oder eine Referenzfahrt kann mit FRF, FNL oder FPL gestartet werden. ▪ <ReferenceOn> = 1 (Standard): Für die Referenzwertbestimmung der Achse muss eine Referenzfahrt mit FRF, FNL oder FPL gestartet werden. Die Verwendung von POS ist nicht zulässig.
RON?	RON? [{<AxisID>}]	Fragt den Modus der Referenzwertbestimmung ab.
FRF	FRF [{<AxisID>}]	Startet eine Referenzfahrt zum Referenzschalter. Das Anfahren erfolgt immer von derselben Seite, unabhängig von der Achsenposition beim Senden des Befehls.
FRF?	FRF? [{<AxisID>}]	Fragt ab, ob der Referenzwert für eine Achse bereits bestimmt wurde. 1 = Referenzwert wurde bestimmt 0 = Referenzwert wurde nicht bestimmt
FNL	FNL [{<AxisID>}]	Startet eine Referenzfahrt zum negativen Endschalter.
FPL	FPL [{<AxisID>}]	Startet eine Referenzfahrt zum positiven Endschalter.
POS	POS {<AxisID> <Position>}	Setzt die aktuelle Position (löst keine Bewegung aus) und bestimmt damit den Referenzwert.

Parameter

Mit folgenden Parametern können Referenzfahrten konfiguriert werden:

Parameter	Beschreibung und mögliche Werte
Closed-Loop Deceleration (Phys. Unit/s²) 0xC	Abbremsung im geregelten Betrieb Details siehe "Erzeugung des Dynamikprofils" (S. 26).
Reference Travel Direction 0x47	Standardrichtung für die Referenzfahrt 0 = automatische Erkennung 1 = negative Richtung 2 = positive Richtung
Closed-Loop Velocity (Phys. Unit/s) 0x49	Geschwindigkeit im geregelten Betrieb Details siehe "Erzeugung des Dynamikprofils" (S. 26).
Velocity For Reference Moves (Phys. Unit/s) 0x50	Geschwindigkeit für Referenzfahrt Gibt die maximale Geschwindigkeit für das zweite Anfahren des ausgewählten Schalters während einer Referenzfahrt an. Für hohe Wiederholgenauigkeit bei der Referenzwertbestimmung sollte dieser Wert maximal so groß sein wie der Wert des Parameters 0x49. Wenn der Wert des Parameters 0x50 auf 0 gesetzt wird, sind Referenzfahrten nicht möglich.
Distance Between Limit And Hard Stop (Phys. Unit) 0x63	Abstand zwischen eingebautem Endschalter und mechanischem Anschlag Legt den maximalen Bremsweg bei Referenzfahrten fest. Die tatsächlichen Geschwindigkeiten während einer Referenzfahrt werden auf der Grundlage dieses Werts, der eingestellten Abbremsung (0xC) und der eingestellten Geschwindigkeiten (0x49 und 0x50) berechnet.

INFORMATION

- Führen Sie für größtmögliche Wiederholgenauigkeit die Referenzfahrt immer auf dieselbe Weise aus.

INFORMATION

Die Endschalter können nur für Referenzfahrten verwendet werden, wenn der Stellweg nicht durch Verfahrbereichsgrenzen (S. 35) beschränkt wird.

INFORMATION

Sie können für Referenzfahrten auch die digitalen Eingänge des C-863 als Quelle für das Referenzsignal, das negative Endschaltersignal oder das positive Endschaltersignal verwenden. Siehe "Digitale Eingangssignale als Schaltersignale verwenden" (S. 117) für mehr Informationen.

INFORMATION

Wenn die absolute Position der Achse manuell festgelegt wird, können Konflikte mit den Einstellungen für die Verfahrbereichsgrenzen entstehen (Parameter 0x15, Abfrage mit **TMX?**, und 0x30, Abfrage mit **TMN?**).

- Setzen Sie die absolute Position der Achse nur manuell, wenn die Referenzwertbestimmung nicht anders möglich ist.

3.7 Kommunikationsschnittstellen

Verfügbare Kommunikations-Schnittstellen

Der C-863 kann von einem PC mit ASCII-Befehlen gesteuert werden. Der Anschluss an den PC kann über eine direkte Verbindung oder über ein Daisy-Chain-Netzwerk erfolgen. Für die direkte Verbindung zum PC können folgende Schnittstellen des C-863 verwendet werden:

- Serielle RS-232-Verbindung
- USB-Verbindung

Es darf immer nur eine von beiden Schnittstellen an den PC angeschlossen sein.

Standard-Kommunikationseinstellungen

Schnittstelle	Eigenschaft	Standardwert
RS-232	Baudrate	38400 Einstellungen der DIP-Schalter 5 und 6; siehe "Baudrate" (S. 76) Sonstiges: 8 Daten-Bits und 1 Stopp-Bit, ohne Parität; interne Zwischenspeicher erfordern keinen Handshake

Daisy-Chain-Netzwerk

Über ein Daisy-Chain-Netzwerk können bis zu 16 Controller über einen einzigen RS-232- oder USB-Anschluss an den PC angeschlossen werden. Die Vernetzung erfolgt in Reihe. Siehe auch "Begriffserklärung" (S. 3).

3.8 PC-Softwareübersicht

Die folgende Tabelle zeigt die PC-Software, die sich auf der Produkt-CD befindet. Die angegebenen Betriebssysteme stehen für folgende Versionen:

- Windows: Versionen XP (Service Pack 3), Vista (Service Pack 1) und 7
- Linux: Kernel 2.6, GTK 2.0, glibc 2.4 (für die Entwicklung der PC-Software verwendete Konfiguration)

PC-Software	Betriebs-system	Kurzbeschreibung	Empfohlene Verwendung
Dynamische Programm-bibliothek für GCS	Windows, Linux (USB nur unter Windows)	Ermöglicht die Software-Programmierung für den C-863 mit Programmiersprachen wie z. B. C++. Die Funktionen in der dynamischen Programmbibliothek basieren auf dem PI General Command Set (GCS).	Für Anwender, die für ihre Anwendung eine dynamische Programmbibliothek nutzen möchten. Wird für PIMikroMove® benötigt. Wird für die LabVIEW-Treiber benötigt, wenn die Kommunikation über USB oder ein Daisy-Chain-Netzwerk hergestellt werden soll.

PC-Software	Betriebs-system	Kurzbeschreibung	Empfohlene Verwendung
LabVIEW-Treiber	Windows, Linux	LabVIEW ist eine Software für die Datenerfassung und Prozesssteuerung (von National Instruments separat zu beziehen). Die C-863-LabVIEW-Software ist eine Sammlung von Virtual-Instrument-Treibern (VI-Treiber) für den C-863-Controller. Diese Treiber unterstützen das GCS.	Für Anwender, die LabVIEW zur Programmierung ihrer Anwendung verwenden möchten.
PIMikroMove®	Windows	Grafische Benutzerschnittstelle für Windows, mit der der C-863 und andere Controller von PI bedient werden können: ▪ Das System kann ohne Programmieraufwand gestartet werden ▪ Grafische Darstellung der Bewegungen im geregelten und ungeregelten Betrieb ▪ Makrofunktionalität zum Abspeichern von Befehlsfolgen auf dem PC (Hostmakros) ▪ Joystickunterstützung ▪ Komplette Umgebung für die Befehlseingabe, zum Ausprobieren von verschiedenen Befehlen Für die Bedienung von PIMikroMove® sind keine Befehlskenntnisse erforderlich. PIMikroMove® verwendet die dynamische Programmbibliothek zur Kommandierung des Controllers.	Für Anwender, die einfache Automatisierungsaufgaben ausführen oder ihre Ausrüstung vor oder anstelle der Programmierung einer Anwendung testen möchten. Ein Logfenster mit Anzeige der gesendeten Befehle ermöglicht auch das Erlernen der Befehlsverwendung.
PI Terminal	Windows	Einfache, grafische Benutzerschnittstelle für Windows, die für nahezu alle PI-Controller verwendet werden kann (siehe die Beschreibung des Fensters Command Entry im PIMikroMove® Benutzerhandbuch).	Für Anwender, die die Befehle des GCS direkt an den Controller senden möchten.
PIStageEditor	Windows	Programm zum Öffnen und Editieren von Verstellerdatenbanken.	Für Anwender, die sich intensiv mit den Inhalten der Verstellerdatenbanken auseinandersetzen möchten.
PI Update Finder	Windows	Überprüft die auf dem PC installierte Software von PI. Wenn auf dem PI-Server aktuellere Versionen der PC-Software vorhanden sind, wird das Herunterladen angeboten.	Für Anwender, die die PC-Software aktualisieren möchten.

PC-Software	Betriebs-system	Kurzbeschreibung	Empfohlene Verwendung
TMS320F28xx Updater	Windows	Programm zur Unterstützung des Anwenders bei der Aktualisierung der Firmware.	Für Anwender, die die Firmware aktualisieren möchten.
USB-Treiber	Windows	Treiber für die USB-Schnittstelle	Für Anwender, die den Controller über die USB-Schnittstelle an den PC anschließen möchten.

3.9 Verstellerdatenbanken

Sie können in der PC-Software von PI den für Ihren Versteller geeigneten Parametersatz aus einer Verstellerdatenbank auswählen. Die Software überträgt die Werte des ausgewählten Parametersatzes in den flüchtigen Speicher des Controllers.

Dateiname der Datenbank	Editierbar?	Beschreibung
PIStages2.dat	Nein, Updates können von der PI Website heruntergeladen werden (S. 58).	Enthält Parametersätze für alle Standard-Versteller von PI; wird bei der Installation der Software automatisch auf dem PC gespeichert.
PI_UserStages2.dat	Ja, neue Parametersätze können angelegt, editiert und gespeichert werden (S. 284).	Wird angelegt, wenn Sie mit der PC-Software erstmalig eine Verbindung zu Ihrem Versteller herstellen (d. h. bei der Verstellerauswahl in PIMikroMove® bzw. bei Verwendung der Befehle VST? oder CST).
M-xxx.dat	Nein, Updates erhalten Sie von unserem Kundendienst (S. 309).	Enthält den Parametersatz für einen kundenspezifischen Versteller, für die Installation siehe "Kundenspezifische Verstellerdatenbank installieren" (S. 60).

Weitere Informationen finden Sie auch in den Benutzerhandbüchern für PIMikroMove®, PIStageEditor und die PI GCS-Programmbibliothek.

4 Auspacken

1. Packen Sie den C-863 vorsichtig aus.
2. Vergleichen Sie die erhaltene Lieferung mit dem Inhalt laut Vertrag und mit der Packliste.
3. Überprüfen Sie den Inhalt auf Anzeichen von Schäden. Bei Anzeichen von Beschädigungen oder fehlenden Teilen wenden Sie sich sofort an PI.
4. Bewahren Sie das komplette Verpackungsmaterial auf für den Fall, dass das Produkt zurückgeschickt werden muss.

5 Schnellstart

HINWEIS



Falsche Verkabelung!

Das gleichzeitige Anschließen der USB- und der RS-232-Schnittstelle des Controllers an den PC kann den PC oder den Controller beschädigen.

- Schließen Sie entweder die USB- oder die RS-232-Schnittstelle an den PC an.

HINWEIS



Schwingungen!

Ungeeignete Einstellungen der Regelparameter des C-863 können den Versteller zum Schwingen bringen. Schwingungen können den Versteller und/oder die auf ihm angebrachte Last beschädigen.

- Befestigen Sie den Versteller und alle Lasten ausreichend.
- Wenn der Versteller schwingt (ungewöhnliches Laufgeräusch), schalten Sie den Servomodus sofort aus oder trennen Sie den C-863 von der Stromversorgung.
- Schalten Sie den Servomodus erst wieder ein, nachdem Sie die Einstellungen der Regelparameter des C-863 geändert haben; siehe "Regelparameter optimieren" (S. 96).
- Wenn aufgrund einer sehr hohen Last Schwingungen bereits während der Referenzfahrt auftreten, folgen Sie für die Referenzfahrt den Anweisungen in "Störungsbehebung" (S. 303).

Ziel des Schnellstarts ist es, in der PC-Software PIMikroMove® erste Testbewegungen eines Verstellers zu starten, der an einen nicht vernetzten C-863 angeschlossen ist.

1. Installieren Sie Folgendes auf dem PC:
 - die PC-Software und die USB-Treiber von der Produkt-CD
 - Updates für PC-Software und Verstellerdatenbank PISTages2.dat
 - wenn von PI separat zur Verfügung gestellt: kundenspezifische Verstellerdatenbank(en)

Details siehe "PC-Software installieren" (S. 57).

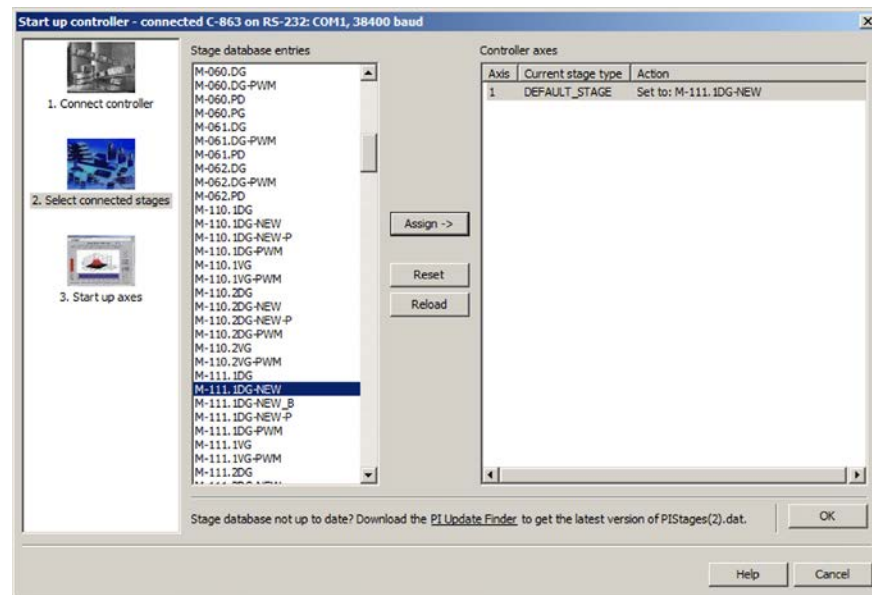
2. Schließen Sie den C-863 über die mit dem Erdanschlusssymbol gekennzeichnete Schraube an das Erdungssystem an (S. 63).
3. Schließen Sie Folgendes an den C-863 an:
 - das mitgelieferte Weitbereichsnetzteil (**nicht** über das Netzkabel an der Steckdose angeschlossen) an den Anschluss **15-30 VDC**. Details siehe "Netzteil anschließen" (S. 63).
 - den Versteller an die Buchse **DC Motor only**. Details siehe "Versteller anschließen" (S. 64).
 - den PC über die RS-232-Schnittstelle (Buchse **RS-232 In**) **oder** über die USB-Schnittstelle. Details siehe "PC anschließen" (S. 64).
4. Prüfen Sie die Einstellungen der DIP-Schalter (S. 74), und passen Sie sie bei Bedarf an Ihre Anwendung an. Die Controlleradresse 1 muss eingestellt sein.
5. Schalten Sie den C-863 ein (S. 77), indem Sie das Netzkabel des Weitbereichsnetzteils mit der Steckdose verbinden.
6. Starten Sie PIMikroMove® am PC.
7. Stellen Sie die Kommunikation zwischen dem C-863 und dem PC in PIMikroMove® über die RS-232-Schnittstelle oder die USB-Schnittstelle her. Details siehe "Kommunikation herstellen" (S. 78).
8. Wenn in PIMikroMove® der Schritt **Select connected stages** angezeigt wird, wählen Sie den Verstellertyp des angeschlossenen Verstellers aus:

Wenn rechts im Fenster in der Liste **Controller axes** bereits der korrekte Verstellertyp in der Spalte **Current stage type** eingetragen ist:

 - Klicken Sie auf **OK**.

Wenn der eingetragene Verstellertyp nicht korrekt ist:

 - a) Markieren Sie den Verstellertyp in der Liste **Stage database entries**.
 - b) Klicken Sie auf **Assign**.
 - c) Bestätigen Sie die Auswahl mit **OK**, um die Parametereinstellungen für den ausgewählten Verstellertyp aus der Verstellerdatenbank in den flüchtigen Speicher des C-863 zu laden.



Nach dem Klicken von **OK** wechselt das Fenster **Start up controller** zum Schritt **Start up axes**.

9. Führen Sie im Schritt **Start up axes** die Referenzfahrt für die Achse aus, damit der Controller die absolute Achsenposition kennt:
 - Wenn Sie die Referenzfahrt zum Referenzschalter starten wollen, klicken Sie auf **Ref. switch**.
 - Wenn Sie die Referenzfahrt zum negativen Endschalter starten wollen, klicken Sie auf **Neg. limit**.
 - Wenn Sie die Referenzfahrt zu positiven Endschalter starten wollen, klicken Sie auf **Pos. limit**.

Wenn eine Warnmeldung erscheint, dass der Servomodus ausgeschaltet ist:

- Schalten Sie den Servomodus durch einen Klick auf die Schaltfläche **Switch on servo** ein.

Die Achse führt die Referenzfahrt aus.

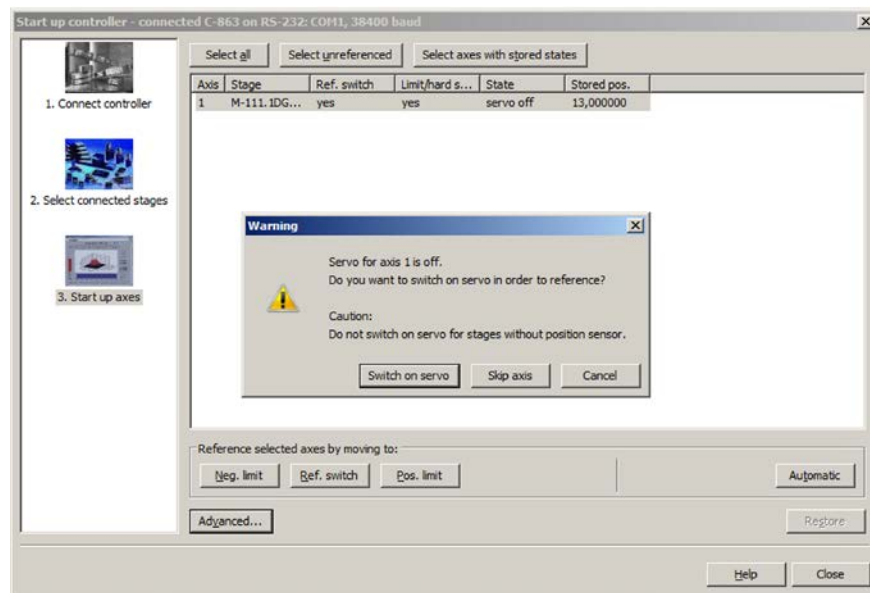


Abbildung 10: Start up controller – Start up axes

10. Nach erfolgreicher Referenzfahrt klicken Sie auf **OK** > **Close**.

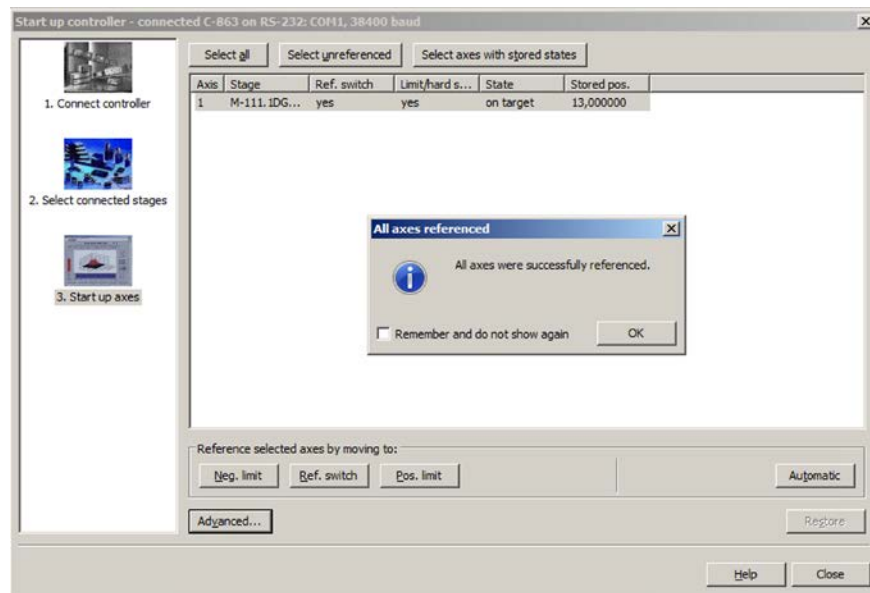


Abbildung 11: Start up controller – All axes referenced

Das Hauptfenster von PIMikroMove® öffnet sich.

11. Starten Sie einige Testbewegungen der Achse.

Im Hauptfenster von PIMikroMove® können Sie z. B. Schritte mit einer bestimmten Schrittweite ausführen, indem Sie auf die entsprechenden Pfeiltasten für die Achse klicken.

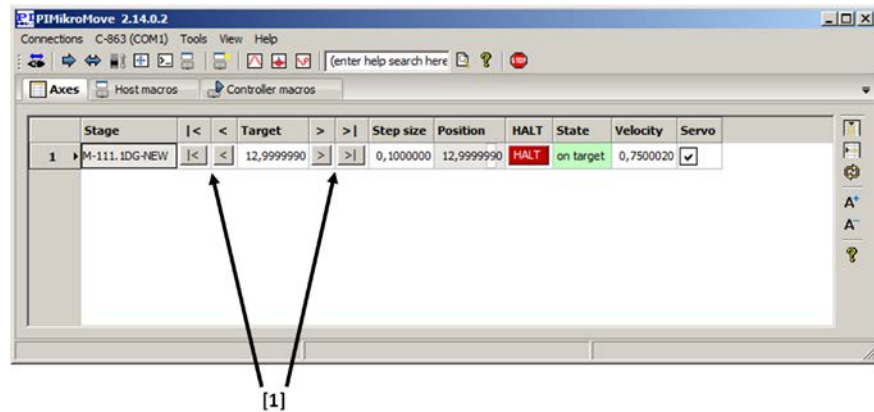


Abbildung 12: Hauptfenster von PIMikroMove®; [1] Pfeiltasten für Bewegung

6 Installation

In diesem Kapitel

PC-Software installieren	57
C-863 montieren	62
C-863 erden	63
Netzteil an C-863 anschließen	63
Versteller anschließen	64
PC anschließen	64
Analogen Joystick anschließen	68
Digitale Ein- und Ausgänge anschließen	69
Analoge Signalquellen anschließen	71

6.1 PC-Software installieren

Die Kommunikation zwischen dem C-863 und einem PC ist zur Konfiguration des C-863 und zur Bewegungskommandierung mit den Befehlen des GCS notwendig. Dafür stehen verschiedene PC-Software-Anwendungen zur Verfügung.

6.1.1 Erstinstallation ausführen

Zubehör

- PC mit Windows-Betriebssystem (XP, Vista, 7) oder Linux-Betriebssystem
- Produkt-CD (im Lieferumfang)

PC-Software auf Windows installieren

1. Starten Sie den Installationsassistenten, indem Sie im Installationsverzeichnis (Hauptverzeichnis der CD) auf das Icon  oder auf die Datei **setup.exe** doppelklicken.
2. Folgen Sie den Anweisungen am Bildschirm.
Sie können zwischen der Standardinstallation (typical) und der benutzerdefinierten Installation (custom) wählen. Bei der Standardinstallation (empfohlen) werden u. a. folgende Komponenten installiert:
 - LabVIEW-Treiber
 - Dynamische Programmbibliothek für GCS
 - PIMikroMove®

3. Wenn Sie den Controller über die USB-Schnittstelle an den PC anschließen möchten:
 - a) Starten Sie die Installation der USB-Treiber, indem Sie im entsprechenden Dialogfenster auf **Ja** klicken.
 - b) Folgen Sie den Anweisungen am Bildschirm.

PC-Software auf Linux installieren

1. Entpacken Sie das tar-Archiv aus dem Verzeichnis /linux der Produkt-CD in ein Verzeichnis auf Ihrem PC.
2. Öffnen Sie ein Terminal und wechseln Sie in das Verzeichnis, in das Sie das tar-Archiv entpackt haben.
3. Melden Sie sich als Superuser (Root-Rechte) an.
4. Geben Sie `./INSTALL` ein, um die Installation zu starten.
Achten Sie bei der Befehlseingabe auf Groß-/Kleinschreibung.
5. Folgen Sie den Anweisungen am Bildschirm.

Sie können einzelne Komponenten zur Installation auswählen.

6.1.2 Updates installieren

Die PC-Software wird von PI ständig verbessert.

- Installieren Sie immer die neueste Version der PC-Software und der Verstellerdatenbank `PIStages2.dat`.

Voraussetzung

- ✓ Aktive Verbindung des PC zum Internet.
 - Wenn Ihr PC über keine Internetverbindung verfügt:
Sie haben die Technical Note A000T0032 für den PI Update Finder parat.
Sie finden das Dokument entweder auf der Produkt-CD oder auf unserem Update-Portal (<http://www.update.pi-portal.ws>) in der Zip-Datei für den PI Update Finder.
- ✓ Wenn Ihr PC ein Windows-Betriebssystem verwendet:
 - Wenn das Programm PI Update Finder nicht auf Ihrer Produkt-CD enthalten ist:
Sie haben den PI Update Finder von unserem Update-Portal (<http://www.update.pi-portal.ws>) heruntergeladen und aus der Zip-Datei in ein Verzeichnis auf Ihrem PC entpackt.

- Sie haben die Technical Note A000T0028 für den PI Update Finder parat. Sie finden das Dokument entweder auf der Produkt-CD oder in der Zip-Datei, die Sie für den PI Update Finder heruntergeladen haben.
- ✓ Wenn Ihr PC ein Linux-Betriebssystem verwendet:
 - Sie haben Benutzername und Kennwort für den C-863 parat. Beide Angaben finden Sie in der Datei "C-863 Releasenews_V_x_x_x.pdf" (x_x_x: Versionsnummer der CD) im Ordner \Manuals auf der Produkt-CD.

PC-Software und PIStages2.dat auf Windows aktualisieren

- Verwenden Sie den PI Update Finder:
 - Wenn der zu aktualisierende PC direkt mit dem Internet verbunden ist: Folgen Sie den Anweisungen in der Technical Note A000T0028 (TECHNICAL_NOTE_PI_UPDATE_FINDER_xx.pdf).
 - Wenn der zu aktualisierende PC **nicht** direkt mit dem Internet verbunden ist: Folgen Sie den Anweisungen in der Technical Note A000T0032.

PC-Software auf Linux aktualisieren

1. Öffnen Sie die PI-Website (<http://www.pi-portal.ws>).
2. Klicken Sie auf **Downloads**.
3. Geben Sie im Bereich **User login** am linken Seitenrand den Benutzernamen (username) und das Kennwort (password) aus der Datei "C-663.11_Releasenews_V_x_x_x.pdf" von der Produkt-CD ein.
4. Klicken Sie auf **Login**.
5. Klicken Sie auf die Kategorie **C Motion Controllers**.
6. Klicken Sie auf **C-863 > C-863.11 > Software** (wenn Sie auf **Documents** klicken, werden die neuesten Versionen der entsprechenden Handbücher angezeigt).
7. Klicken Sie unterhalb der neuesten CD-Kopie (CD-Mirror) auf die Schaltfläche **Download** (beinhaltet auch die Handbücher).
8. Speichern Sie die heruntergeladene Archivdatei auf dem PC.
9. Entpacken Sie die Datei in ein separates Installationsverzeichnis.
10. Wechseln Sie im Verzeichnis mit den entpackten Dateien in das Unterverzeichnis linux.

11. Entpacken Sie die Archivdatei im Verzeichnis linux, indem Sie in der Konsole den Befehl `tar -xvpf <Name der Archivdatei>` eingeben.
12. Lesen Sie die Begleitinformationen (Readme-Datei) zum Software-Update durch.
13. Melden Sie sich am PC als Superuser (Root-Rechte) an.
14. Installieren Sie das Update nur, wenn es für Ihre Anwendung sinnvoll ist.

PIStages2.dat auf Linux aktualisieren

1. Öffnen Sie die PI-Website (<http://www.pi-portal.ws>).
2. Klicken Sie auf **Downloads**.
3. Geben Sie im Bereich **User login** am linken Seitenrand den Benutzernamen (username) und das Kennwort (password) aus der Datei "C-663.11_Releasenews_V_x_x_x.pdf" von der Produkt-CD ein.
4. Klicken Sie auf **Login**.
5. Klicken Sie auf die Kategorie **General Software**.
6. Klicken Sie auf **PI Stages**.
7. Klicken Sie auf **pistages2** oder auf die Schaltfläche **Download** darunter.
8. Melden Sie sich am PC als Superuser (Root-Rechte) an.
9. Installieren Sie die heruntergeladene Datei `pistages2.dat` auf dem PC. Sie können zwischen den folgenden Möglichkeiten wählen:
 - Speichern Sie die Datei `pistages2.dat` im Verzeichnis `/usr/local/PI/pi_gcs_translator/`
 - Speichern Sie die Datei `pistages2.dat` in dem Verzeichnis, in das Sie die Linux-Software von der Produkt-CD entpackt haben. Der Pfad lautet `/<Entpackungsverzeichnis>/pi_stages2_dat`. Starten Sie anschließend in diesem Unterverzeichnis das Skript `INSTALL.pi_stages2_dat`.

6.1.3 Kundenspezifische Verstellerdatenbank installieren

Mit einem kundenspezifischen Versteller erhalten Sie von PI gegebenenfalls eine Datei mit einer kundenspezifischen Verstellerdatenbank. Sie müssen diese Datei auf Ihrem PC installieren, damit Sie die Parameterwerte für den kundenspezifischen Versteller in den C-863 laden können.

Kundenspezifische Verstellerdatenbank auf Windows installieren

1. Öffnen Sie auf Ihrem PC das Verzeichnis \PI\GCSTranslator:

Wenn Sie mit PIMikroMove® arbeiten:

- a) Öffnen Sie aus dem Hauptfenster von PIMikroMove über den Menüeintrag **Connections** > **Search for controller software** das Fenster **Version Information**.
- b) Klicken Sie im Fenster **Version Information** auf die Taste **Show GCS PATH...**, um das Verzeichnis \PI\GCSTranslator im Windows-Explorer zu öffnen.

Der Pfad, in dem sich das Verzeichnis \PI befindet, wurde während der Installation der PC-Software festgelegt, normalerweise C:\Dokumente und Einstellungen\All Users\Anwendungsdaten (Windows XP) oder C:\ProgramData (Windows Vista, Windows 7).

2. Kopieren Sie die Datei der Verstellerdatenbank in das Verzeichnis \PI\GCSTranslator auf Ihrem PC.

INFORMATION

Wenn das Verzeichnis \PI\GCSTranslator auf Ihrem PC nicht vorhanden ist:

Damit eine ausführbare Datei (.exe) auf eine Verstellerdatenbank zugreifen kann, müssen beide Dateien im selben Verzeichnis liegen.

Kundenspezifische Verstellerdatenbank auf Linux installieren

1. Melden Sie sich am PC als Superuser (Root-Rechte) an.
2. Kopieren Sie die Datei der Verstellerdatenbank in das Verzeichnis /usr/local/PI/pi_gcs_translator/.

6.2 C-863 montieren

Der C-863 kann als Tischgerät verwendet oder in beliebiger Ausrichtung auf einer Unterlage montiert werden.

INFORMATION

Der C-863 ist stapelbar und kann in einen Schaltschrank eingebaut werden.

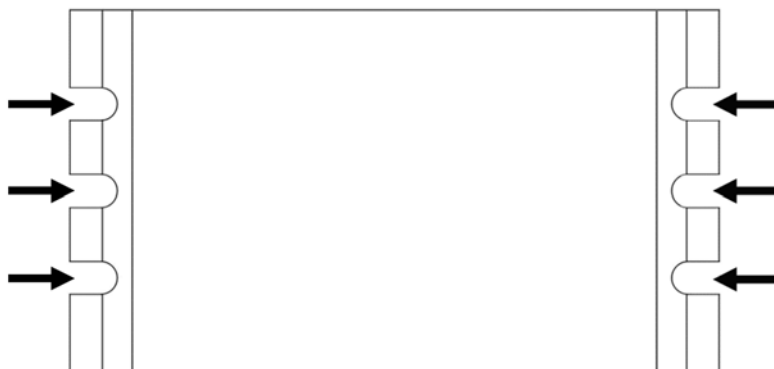


Abbildung 13: C-863.11: Montageleisten mit Aussparungen (siehe Pfeile)

Werkzeug und Zubehör

- Geeignete Schrauben; siehe Maßzeichnung (S. 314).
- Geeigneter Schraubendreher

C-863 auf einer Unterlage befestigen

1. Bringen Sie in die Unterlage die erforderlichen Bohrungen ein.
Die Anordnung der Aussparungen in den Montageleisten des C-863 können Sie der Maßzeichnung (S. 314) entnehmen.
2. Befestigen Sie den C-863 an den Aussparungen in den Montageleisten mit jeweils zwei geeigneten Schrauben pro Seite (siehe Abbildung).

6.3 C-863 erden



Der C-863 ist nicht über den Netzteilanschluss geerdet.

Wenn ein Potentialausgleich erforderlich ist:

- Schließen Sie die mit dem Erdanschlusssymbol gekennzeichnete Schraube (siehe Abbildung) an der Gehäuserückwand des C-863 an das Erdungssystem an.

6.4 Netzteil an C-863 anschließen

HINWEIS



Motorschaden durch zu hohe Betriebsspannung!

Die Ausgangsspannung an den Pins **Motor +** und **Motor -** der Buchse **DC Motor only** kann so hoch wie die vom Netzteil gelieferte Versorgungsspannung sein. Verstärker ohne PWM-Verstärker sind an diese Pins angeschlossen und können durch zu hohe Ausgangsspannung beschädigt werden.

- Schließen Sie ein Netzteil an, dessen Ausgangsspannung die zulässige Betriebsspannung des Verstärkers **nicht** überschreitet.

Voraussetzungen

- ✓ Das Netzteil ist **nicht** über das Netzkabel an der Steckdose angeschlossen.

Werkzeug und Zubehör

- Mitgeliefertes 15-V-Weitbereichsnetzteil (für Netzspannungen zwischen 100 und 240 Volt Wechselspannung bei 50 oder 60 Hz) oder anderes geeignetes Netzteil, das 15 bis 30 Volt Gleichspannung liefert.

Netzteil an den C-863 anschließen

- Verbinden Sie das Netzteil mit dem Anschluss **15-30 VDC** des C-863.

6.5 Versteller anschließen

HINWEIS



Schäden bei Anschluss eines falschen Motors!

Das Anschließen eines Verstellers mit Schrittmotor an einen DC-Motorcontroller kann irreparable Schäden verursachen.

- Schließen Sie an den C-863 nur einen Versteller mit DC-Motor oder Voice-Coil-Antrieb an.

INFORMATION

Der C-863 unterstützt sowohl Versteller mit PWM-Verstärker als auch Versteller ohne PWM-Verstärker. An der Buchse **DC Motor only** (S. 315) stehen separate Leitungen für beide Verstellervarianten zur Verfügung. Die Auswahl der passenden Leitungen muss über den Anschlussstecker des Verstellers gewährleistet werden. Bei Verstellern von PI ist die ordnungsgemäße Verbindung sichergestellt.

Voraussetzung

- ✓ Der C-863 ist ausgeschaltet, d. h. das Netzteil ist **nicht** über das Netzkabel an der Steckdose angeschlossen.

Werkzeug und Zubehör

- Wenn die Entfernung zwischen C-863 und Versteller zu groß ist:
Motorkabel C-815.38, 3 m, erhältlich als optionales Zubehör (S. 18)

Versteller anschließen

- Schließen Sie den Versteller an der Buchse **DC Motor only** des C-863 an.

6.6 PC anschließen

Die Kommunikation zwischen dem C-863 und einem PC ist zur Konfiguration des C-863 und zur Bewegungskommandierung mit den Befehlen des GCS notwendig. Der C-863 verfügt dazu über folgende Schnittstellen:

- RS-232-Schnittstelle
- USB-Schnittstelle

In diesem Abschnitt erfahren Sie, wie Sie die entsprechenden Kabelverbindungen zwischen C-863 und PC sowie in einem Daisy-Chain-Netzwerk herstellen. Alle weiteren Schritte, die für die Herstellung der Kommunikation zwischen C-863 und PC erforderlich sind, sind in den folgenden Abschnitten beschrieben:

- "Kommunikation über RS-232 herstellen" (S. 78)
- "Kommunikation über USB herstellen" (S. 80)
- "Kommunikation für vernetzten Controller herstellen" (S. 81)

INFORMATION

Über ein Daisy-Chain-Netzwerk können bis zu 16 Controller über einen einzigen RS-232- oder USB-Anschluss an den PC angeschlossen werden.

6.6.1 An RS-232-Schnittstelle anschließen

HINWEIS



Falsche Verkabelung!

Das gleichzeitige Anschließen der USB- und der RS-232-Schnittstelle des Controllers an den PC kann den PC oder den Controller beschädigen.

- Schließen Sie entweder die USB- oder die RS-232-Schnittstelle an den PC an.

Voraussetzungen

- ✓ Der PC verfügt über eine freie RS-232-Schnittstelle (auch als "serielle Schnittstelle" oder "COM-Port" bezeichnet, z. B. COM1 oder COM2).

Werkzeug und Zubehör

- RS-232-Nullmodemkabel (C-815.34 im Lieferumfang)

C-863 an den PC anschließen

- Verbinden Sie die Buchse **RS-232 In** an der Vorderwand des C-863 und die RS-232-Schnittstelle des PC (ein Einbaustecker D-Sub 9(m)) mit dem Nullmodemkabel.

6.6.2 An USB-Schnittstelle anschließen

HINWEIS

**Falsche Verkabelung!**

Das gleichzeitige Anschließen der USB- und der RS-232-Schnittstelle des Controllers an den PC kann den PC oder den Controller beschädigen.

- Schließen Sie entweder die USB- oder die RS-232-Schnittstelle an den PC an.

Voraussetzungen

- ✓ Der PC verfügt über eine freie USB-Schnittstelle.

Werkzeug und Zubehör

- USB-Kabel (Typ A auf Mini-B) zur Verbindung mit dem PC (Bestellnummer 000014651; im Lieferumfang)

C-863 an den PC anschließen

- Verbinden Sie die USB-Buchse des C-863 und die USB-Schnittstelle des PC mit dem USB-Kabel.

6.6.3 Daisy-Chain-Netzwerk aufbauen

INFORMATION

Die Vernetzung in einem Daisy-Chain-Netzwerk erfolgt in Reihe. Siehe auch "Begriffserklärung" (S. 3). Dabei ist der erste Controller direkt mit dem PC verbunden.

INFORMATION

Die DIP-Schalter des C-863 müssen passend eingestellt sein:

- Stellen Sie für jeden Controller in einem Daisy-Chain-Netzwerk eine eindeutige Adresse ein. Dabei muss einer der Controller die Adresse 1 haben. Dieser Controller braucht nicht derjenige zu sein, der direkt an den PC angeschlossen ist. Details siehe "Controlleradresse" (S. 75).
- Stellen Sie für alle Controller in einem Daisy-Chain-Netzwerk die gleiche Baudrate ein. Details siehe "Baudrate" (S. 76).

Werkzeug und Zubehör

- Ein Netzkabel für jeden an das Netzwerk anzuschließenden Controller. Verfügbar sind:
 - C-862.CN, 30 cm, im Lieferumfang
 - C-862.CN2, 180 cm, erhältlich als optionales Zubehör (S. 18)

Controller vernetzen

- Bauen Sie die Controller-Reihe auf. Verbinden Sie dazu jeweils den Anschluss **RS-232 Out** des vorhergehenden Controllers über das Netzkabel mit dem Anschluss **RS-232 In** des nachfolgenden Controllers.
- Schließen Sie den ersten Controller der Reihe an den PC an:
 - Verwenden Sie die RS-232-Schnittstelle (S. 65).

oder

- Verwenden Sie die USB-Schnittstelle (S. 66).

INFORMATION

Ein C-863.11 kann mit folgenden Controllern in einem gemeinsamen Daisy-Chain-Netzwerk betrieben werden:

- Mercury Step Schrittmotor-Controller C-663.11
- PILine® Piezomotor-Controller der Reihe C-867
- NEXACT® Controller E-861

6.7 Analogen Joystick anschließen

INFORMATION

An der Buchse **Joystick** können Sie eine Achse eines analogen Joysticks anschließen. Mit der Joystick-Achse können Sie die Geschwindigkeit des am C-863 angeschlossenen Verstellers steuern.

Sie haben folgende Anschlussmöglichkeiten:

- Pin 4: 0 bis 3,3 V Eingangsspannung (kommandierbar als Achse 1 von Joystick-Gerät 1)

oder

- Pin 2: -10 bis 10 V Eingangsspannung (kommandierbar als Achse 1 von Joystick-Gerät 2)

Eine Joystick-Taste kann an Pin 6 der Buchse **Joystick** angeschlossen werden (kommandierbar als Taste 1 von Joystick-Gerät 1).

INFORMATION

Die als optionales Zubehör erhältlichen Joysticks C-819.20 und C-819.30 nutzen die Pins 4 und 6 der Buchse **Joystick**. Über Pin 3 dieser Buchse erfolgt die Spannungsversorgung des Joysticks.

Sie können ein Y-Kabel C-819.20Y verwenden, um zwei C-863 an einen Joystick C-819.20 anzuschließen. In diesem Fall erfolgt die Spannungsversorgung des Joysticks über den C-863, der am X-Zweig des Kabels angeschlossen ist.

Werkzeug und Zubehör

- Analoges Joystick von PI für den Betrieb mit 0 bis 3,3 V, erhältlich als optionales Zubehör (S. 18):
 - C-819.20 analoger Joystick für 2 Achsen
 - Wenn ein Joystick C-819.20 an zwei Controller angeschlossen werden soll: C-819.20Y Y-Kabel

oder

- C-819.30 analoger Joystick für 3 Achsen
- Alternativ: Joystick, der -10 bis 10 V liefert

Analogen Joystick anschließen

- Schließen Sie den Joystick an der Buchse **Joystick** des C-863 an:
 - Wenn Sie einen Joystick C-819.20 nur mit diesem Controller betreiben wollen, verbinden Sie ihn direkt mit dem Controller.
 - Wenn Sie einen Joystick C-819.20 mit zwei Controllern (d.h. zwei Achsen) betreiben wollen, verbinden Sie den Joystick mit dem Y-Kabel C-819.20Y und schließen die beiden Controller an die X- und Y-Zweige des Kabels an. Die Spannungsversorgung des Joysticks erfolgt über den X-Zweig. Deshalb muss der X-Zweig auch dann an einen Controller angeschlossen sein, wenn für diesen Controller die Joystick-Steuerung nicht aktiviert werden soll.
 - Wenn Sie eine Achse des Joysticks C-819.30 anschließen wollen, verbinden Sie das entsprechende Kabel des Joysticks mit dem Controller.
 - Wenn Sie einen Joystick nutzen wollen, der -10 bis 10 V liefert, schließen Sie ihn an Pin 2 der Buchse **Joystick** an.

6.8 Digitale Ein- und Ausgänge anschließen

Die digitalen Ein- und Ausgänge auf der Buchse **I/O** des C-863 können wie folgt verwendet werden:

- Ausgänge: Triggern von externen Geräten; siehe "Digitale Ausgangssignale" (S. 103).
- Eingänge: Verwendung in Makros (S. 116) und/oder als Quelle für die Referenz- und Endschaltersignale der Achse (S. 117)

6.8.1 Digitale Ausgänge anschließen

INFORMATION

Digitale Ausgangssignale sind auf den Pins 5, 6, 7 und 8 der Buchse **I/O** verfügbar.

INFORMATION

Wenn die Pushbutton-Box C-170.PB von PI an der Buchse **I/O** angeschlossen ist, zeigt sie über LEDs den Status der digitalen Ausgangsleitungen an.

Werkzeug und Zubehör

- Geeignetes Kabel, z. B. C-170.IO IO-Kabel mit offenem Ende, erhältlich als optionales Zubehör (S. 18)
- Zu triggerndes Gerät mit digitalem Eingang für TTL-Signale

Zu triggerndes Gerät anschließen

- Schließen Sie ein geeignetes Gerät an einen der Pins 5, 6, 7 oder 8 der Buchse **I/O** des C-863 an.

6.8.2 Digitale Eingänge anschließen**INFORMATION**

Digitale Eingangssignale können über die Pins 1, 2, 3 und 4 der Buchse **I/O** in den C-863 eingespeist werden.

INFORMATION

Die digitalen Eingänge (Pins 1 bis 4) auf der Buchse **I/O** können auch als analoge Eingänge genutzt werden.

- Digital: TTL
- Analog: 0 bis +5 V

Werkzeug und Zubehör

- Geeignete Signalquelle:
 - Wenn die digitalen Eingänge in Makros verwendet werden sollen, kann z.B. die Pushbutton-Box C-170.PB angeschlossen werden, erhältlich als optionales Zubehör (S. 18).
 - Wenn die digitalen Eingänge als Quelle für die Referenz- und Endschalersignale der Achse verwendet werden sollen, darf der Signalpegel nur einmal über den gesamten Stellweg wechseln.
- Wenn notwendig: Geeignetes Kabel, z. B. C-170.IO IO-Kabel mit offenem Ende, erhältlich als optionales Zubehör (S. 18).

Digitale Signalquelle anschließen

- Schließen Sie eine geeignete Signalquelle an einen der Pins 1, 2, 3, oder 4 der Buchse **I/O** des C-863 an.

6.9 Analoge Signalquellen anschließen

Die analogen Eingänge auf der Buchse **I/O** des C-863 können wie folgt verwendet werden:

- Verwendung in Makros (S. 120): Details und Beispiele zu Makros finden Sie unter "Controllermakros" (S. 131).
- Scan-Anwendungen mit PIMikroMove® (siehe PIMikroMove® Handbuch)

INFORMATION

Analoge Eingangssignale können über die Pins 1, 2, 3 und 4 der Buchse **I/O** in den C-863 eingespeist werden.

INFORMATION

Die analogen Eingänge (Pins 1 bis 4) auf der Buchse **I/O** können auch als digitale Eingänge genutzt werden.

- Analog: 0 bis +5 V
- Digital: TTL

Werkzeug und Zubehör

- Geeignete Signalquelle
- Wenn notwendig: Geeignetes Kabel, z. B. C-170.IO IO-Kabel mit offenem Ende, erhältlich als optionales Zubehör (S. 18).

Analoge Signalquelle anschließen

- Schließen Sie eine geeignete Signalquelle an einen der Pins 1, 2, 3 oder 4 der Buchse **I/O** des C-863 an.

7 Inbetriebnahme

In diesem Kapitel

Allgemeine Hinweise zur Inbetriebnahme	73
DIP-Schalter-Einstellungen anpassen	74
C-863 einschalten	77
Kommunikation herstellen	78
Bewegungen starten	87
Sprungantwort aufzeichnen	92
Regelparameter optimieren	96

7.1 Allgemeine Hinweise zur Inbetriebnahme

HINWEIS



Schäden durch deaktivierte Endschalterauswertung!

Der Aufprall eines bewegten Teils am Ende des Stellwegs oder auf einem Hindernis sowie hohe Beschleunigungen können Schäden oder erheblichen Verschleiß an der Mechanik verursachen.

- Vermeiden Sie Bewegungen im ungeregelten Betrieb.
- Wenn Bewegungen im ungeregelten Betrieb notwendig sind:
 - Setzen Sie den Stellwert mit dem Befehl SMO so, dass sich die Achse mit niedriger Geschwindigkeit bewegt.
 - Stoppen Sie die Achse rechtzeitig. Verwenden Sie dazu die Befehle #24, STP oder HLT, oder setzen Sie den Stellwert mit dem Befehl SMO auf null.
- Deaktivieren Sie **nicht** per Parametereinstellung die Endschalterauswertung durch den C-863.
- Prüfen Sie die Funktion der Endschalter bei etwa 10 % bis 20 % der Maximalgeschwindigkeit.
- Halten Sie bei einer Fehlfunktion der Endschalter die Bewegung sofort an.

7.2 DIP-Schalter-Einstellungen anpassen

7.2.1 Generelle Vorgehensweise

INFORMATION

Geänderte DIP-Schalter-Einstellungen werden nach dem Einschalten des C-863 wirksam.

- Wenn Sie DIP-Schalter-Einstellungen bei eingeschaltetem C-863 geändert haben, schalten Sie den C-863 aus und wieder ein, um die neuen Einstellungen zu aktivieren.

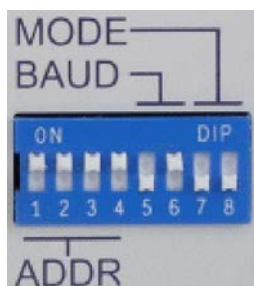


Abbildung 14: DIP-Schalter: Schalter oben = EIN; Schalter unten = AUS

Schalter	Funktion
1 bis 4	Controlleradresse (S. 75); 16 mögliche Kombinationen
5 und 6	Baudrate (S. 76)
7	Logikpegel der Endschalter (S. 76)
8	Update-Modus (S. 77)

Voraussetzung

- ✓ Der C-863 ist ausgeschaltet, d. h. das Netzteil ist **nicht** über das Netzkabel an der Steckdose angeschlossen.

DIP-Schalter-Einstellungen anpassen

- Bringen Sie die einzelnen DIP-Schalter in die für Ihre Anwendung passende Stellung. Details finden Sie in den nachfolgenden Tabellen.

7.2.2 Controlleradresse

Adresse*	S1	S2	S3	S4
1	EIN	EIN	EIN	EIN
2	EIN	EIN	EIN	AUS
3	EIN	EIN	AUS	EIN
4	EIN	EIN	AUS	AUS
5	EIN	AUS	EIN	EIN
6	EIN	AUS	EIN	AUS
7	EIN	AUS	AUS	EIN
8	EIN	AUS	AUS	AUS
9	AUS	EIN	EIN	EIN
10	AUS	EIN	EIN	AUS
11	AUS	EIN	AUS	EIN
12	AUS	EIN	AUS	AUS
13	AUS	AUS	EIN	EIN
14	AUS	AUS	EIN	AUS
15	AUS	AUS	AUS	EIN
16	AUS	AUS	AUS	AUS

*Werkseitige Voreinstellungen sind fett gedruckt.

INFORMATION

Für jeden Controller in einem Daisy-Chain-Netzwerk muss eine eindeutige Adresse eingestellt sein. Dabei muss einer der Controller die Adresse 1 haben. Dieser Controller braucht nicht derjenige zu sein, der direkt an den PC angeschlossen ist.

INFORMATION

Ein nicht vernetzter Controller muss die Adresse 1 haben, wenn

- er in PIMikroMove® verwendet werden soll.
- er in LabVIEW verwendet werden soll.
- er mit dem PITerminal ohne Angabe der Empfängeradresse angesprochen werden soll; in den Antworten des C-863 entfallen dann auch Empfänger- und Senderadresse (S. 158).

7.2.3 Baudrate

Baudrate*	S5	S6
9600	EIN	EIN
19200	EIN	AUS
38400	AUS	EIN
115200	AUS	AUS

*Werkseitige Voreinstellungen sind fett gedruckt.

INFORMATION

Für alle Controller in einem Daisy-Chain-Netzwerk muss die gleiche Baudrate eingestellt sein.

7.2.4 Logikpegel der Endschalter

Endschalter (Hardware-Einstellung)*	S7
low-aktiv	EIN
high-aktiv	AUS

*Werkseitige Voreinstellungen sind fett gedruckt.

INFORMATION

Der C-863 kann über den DIP-Schalter 7 und den Parameter **Limit Mode** (ID 0x18) an den Logikpegel der Endschalter des angeschlossenen Verstellers angepasst werden (high-aktiv oder low-aktiv). Damit sich der Versteller bewegen kann, müssen die Einstellungen des DIP-Schalters **und** des Parameters mit dem Endschalter-Logikpegel übereinstimmen.

DIP-Schalter 7 ist für die Verwendung von high-aktiv-Signalen voreingestellt (z. B. für DC-Motor-Versteller von PI). Wenn Sie Ihren PI-Versteller in der PC-Software von PI aus einer Verstellerdatabank auswählen, ist auch der Parameter **Limit Mode** bereits korrekt eingestellt.

7.2.5 Update-Modus

Update-Modus	S8
Firmware-Update	EIN
Normalbetrieb	AUS

*Werkseitige Voreinstellungen sind fett gedruckt.

INFORMATION

Wenn sich der C-863 im Firmware-Update-Modus befindet (DIP-Schalter 8 in Stellung "EIN" (oben)), bleiben nach dem Einschalten des C-863 alle LEDs ausgeschaltet.

7.3 C-863 einschalten

INFORMATION

Der C-863 ist für den geregelten Betrieb mit inkrementellen Positionssensoren vorgesehen (Servomodus Ein). Nach dem Einschalten ist standardmäßig der unregelte Betrieb aktiviert (Servomodus Aus).

- Fragen Sie die aktuelle Betriebsart mit den Befehlen `SV0?`, `#4` oder `SRG?` ab.
- Aktivieren Sie den geregelten Betrieb mit dem Befehl `SV0`.
- Wenn nötig, programmieren Sie ein Startup-Makro, das den C-863 über den Befehl `SV0` im geregelten Betrieb startet; siehe "Startup-Makro einrichten" (S. 142).
- Vermeiden Sie Bewegungen im unregelten Betrieb.

Voraussetzungen

- ✓ Sie haben die allgemeinen Hinweise zur Inbetriebnahme gelesen und verstanden (S. 73).
- ✓ Der C-863 wurde ordnungsgemäß installiert (S. 57).
- ✓ Sie haben die DIP-Schalter des C-863 passend zu Ihrer Anwendung eingestellt (S. 74).

C-863 einschalten

- Verbinden Sie das Netzkabel des Netzteils mit der Steckdose.

Der C-863 kopiert Informationen vom permanenten Speicher in den flüchtigen Speicher.

Die LED **STA** an der Vorderwand des C-863 zeigt den Status des C-863 an:

- grün: C-863 ist bereit für den Normalbetrieb
 - aus: Wenn sich DIP-Schalter 8 in Stellung "EIN" (oben) befindet, ist der C-863 im Firmware-Update-Modus. Andernfalls könnte der C-863 defekt sein.
- Wenn sich der DIP-Schalter 8 in Stellung "AUS" (unten) befindet und die LED **STA** nach dem Einschalten nicht leuchtet, kontaktieren Sie unseren Kundendienst (S. 309).

7.4 Kommunikation herstellen

Im Folgenden ist das Vorgehen für PIMikroMove® beschrieben.

INFORMATION

Verwenden Sie die Registerkarten **USB Daisy Chain** und **RS-232 Daisy Chain** in der PC-Software nur dann zum Aufbau der Kommunikation, wenn Sie tatsächlich ein Daisy-Chain-Netzwerk an den PC angeschlossen haben.

INFORMATION

Ein nicht vernetzter Controller muss die Adresse 1 haben, wenn er in PIMikroMove® verwendet werden soll. Details siehe "Controlleradresse" (S. 75).

7.4.1 Kommunikation über RS-232 herstellen

Voraussetzungen

- ✓ Sie haben die allgemeinen Hinweise zur Inbetriebnahme gelesen und verstanden (S. 73).
- ✓ Der C-863 ist an die RS-232-Schnittstelle des PC angeschlossen (S. 65).
- ✓ Sie haben vor dem Einschalten des C-863 folgende Einstellungen mit den entsprechenden DIP-Schaltern vorgenommen (S. 74):
 - Controlleradresse = 1
 - passende Baudrate

- ✓ Der C-863 ist eingeschaltet (S. 77).
- ✓ Der PC ist eingeschaltet.
- ✓ Die benötigte Software ist auf dem PC installiert (S. 57).
- ✓ Sie haben das Handbuch der verwendeten PC-Software gelesen und verstanden. Die Software-Handbücher finden Sie auf der Produkt-CD.

Kommunikation herstellen

1. Starten Sie PIMikroMove®.

Das Fenster **Start up controller** öffnet sich mit dem Schritt **Connect controller**.

- Wenn sich das Fenster **Start up controller** nicht automatisch öffnet, wählen Sie im Hauptfenster den Menüeintrag **Connections > New...**

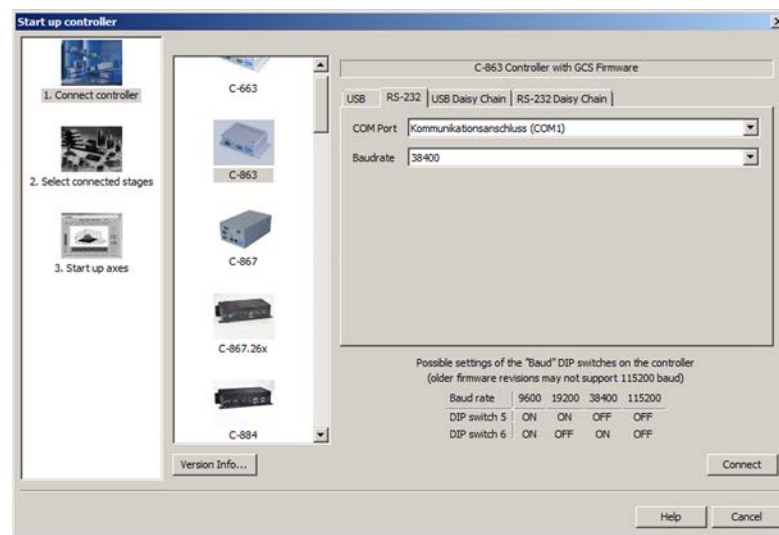


Abbildung 15: Start up controller – Connect controller

2. Wählen Sie im Feld für die Controllerauswahl **C-863** aus.
3. Wählen Sie auf der rechten Seite des Fensters die Registerkarte **RS-232** aus.
4. Wählen Sie im Feld **COM Port** den COM-Port des PC aus, an dem Sie den C-863 angeschlossen haben.
5. Stellen Sie im Feld **Baudrate** den Wert ein, der mit den DIP-Schaltern 5 und 6 des C-863 eingestellt ist.

Damit passen Sie die Baudrate des PC an die Baudrate des C-863 an.

6. Klicken Sie auf **Connect**, um die Kommunikation herzustellen.

Wenn die Kommunikation erfolgreich hergestellt wurde, wechselt das Fenster **Start up controller** zum Schritt **Select connected stages**.

7.4.2 Kommunikation über USB herstellen

INFORMATION

Wenn der Controller über den USB-Anschluss verbunden und eingeschaltet ist, wird die USB-Schnittstelle in der PC-Software auch als COM-Port angezeigt.

Voraussetzungen

- ✓ Sie haben die allgemeinen Hinweise zur Inbetriebnahme gelesen und verstanden (S. 73).
- ✓ Der C-863 ist an die USB-Schnittstelle des PC angeschlossen (S. 66).
- ✓ Sie haben vor dem Einschalten des C-863 die DIP-Schalter für die Controlleradresse auf die Adresse 1 eingestellt (S. 74).
- ✓ Der C-863 ist eingeschaltet (S. 77).
- ✓ Der PC ist eingeschaltet.
- ✓ Die benötigte Software sowie die USB-Treiber sind auf dem PC installiert.
- ✓ Sie haben das Handbuch der verwendeten PC-Software gelesen und verstanden. Die Software-Handbücher finden Sie auf der Produkt-CD.

Kommunikation herstellen

1. Starten Sie PIMikroMove®.

Das Fenster **Start up controller** öffnet sich mit dem Schritt **Connect controller**.

- Wenn sich das Fenster **Start up controller** nicht automatisch öffnet, wählen Sie im Hauptfenster den Menüeintrag **Connections > New...**

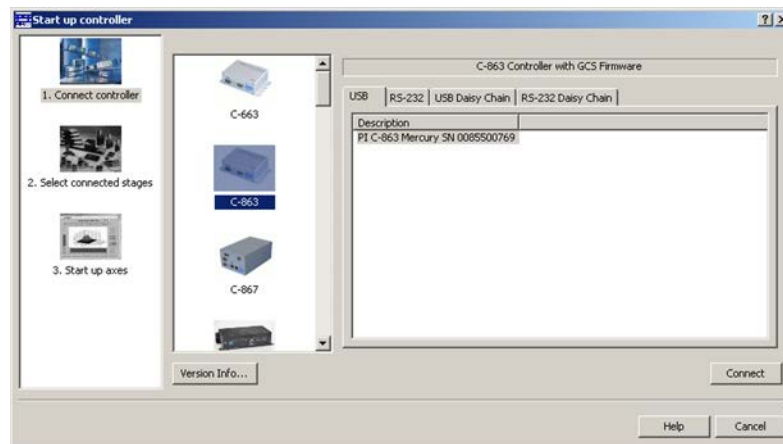


Abbildung 16: Start up controller – Connect controller

2. Wählen Sie im Feld für die Controllerauswahl **C-863** aus.
3. Wählen Sie auf der rechten Seite des Fensters die Registerkarte **USB** aus.
4. Wählen Sie auf der Registerkarte **USB** den angeschlossenen C-863 aus.
5. Klicken Sie auf **Connect**, um die Kommunikation herzustellen.

Wenn die Kommunikation erfolgreich hergestellt wurde, öffnet sich der Dialog **Select connected stages**.

- Wenn die Kommunikation nicht hergestellt werden konnte, suchen Sie in "Störungsbehebung" (S. 303) nach einer Lösung des Problems.

7.4.3 Kommunikation für vernetzten Controller herstellen

Im Folgenden ist das Vorgehen für PIMikroMove® und für PITerminal beschrieben.

INFORMATION

Wenn Sie die Kommunikation mit einem vernetzten Controller über PITerminal herstellen, ist die Adresse des anzusprechenden Controllers in jeder Befehlszeile erforderlich. Details siehe "Empfänger- und Senderadresse" (S. 158).

- Verwenden Sie PITerminal, um die Kommunikation mit vernetzten Controllern zu testen.

INFORMATION

Die RS-232-Ausgangsleitungen mancher PCs sind nicht für die Maximalanzahl von 16 Controllern in einem Netzwerk geeignet. Wenn Sie ein Daisy-Chain-Netzwerk über die RS-232-Schnittstelle an einen solchen PC angeschlossen haben, können Kommunikationsstörungen auftreten (z. B. Timeout). Bei Kommunikationsstörungen:

1. Entfernen Sie das Nullmodemkabel von der Buchse **RS-232 In** des Controllers, der mit dem PC verbunden ist.
2. Schließen Sie das Daisy-Chain-Netzwerk über die USB-Schnittstelle dieses Controllers an den PC an.

Voraussetzungen

- ✓ Sie haben die allgemeinen Hinweise zur Inbetriebnahme gelesen und verstanden (S. 73).
- ✓ Sie haben das Daisy-Chain-Netzwerk aufgebaut (S. 66).
- ✓ Sie haben vor dem Einschalten an allen vernetzten C-863 die DIP-Schalter für die Controlleradresse (S. 75) und die Baudrate (S. 76) passend eingestellt.
- ✓ Alle Controller im Daisy-Chain-Netzwerk sind eingeschaltet (S. 77).
- ✓ Der PC ist eingeschaltet.
- ✓ Die benötigte Software ist auf dem PC installiert (S. 57).
- ✓ Wenn Sie den ersten Controller in der Reihe über die USB-Schnittstelle an den PC angeschlossen haben: Die USB-Treiber sind auf dem PC installiert (S. 57).
- ✓ Sie haben das Handbuch der verwendeten PC-Software gelesen und verstanden. Die Software-Handbücher finden Sie auf der Produkt-CD.

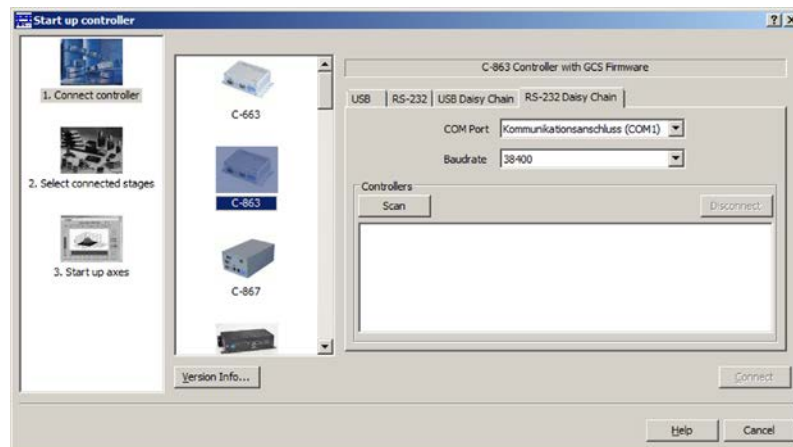
Kommunikation mit PIMikroMove® herstellen

1. Starten Sie PIMikroMove®.

Das Fenster **Start up controller** öffnet sich mit dem Schritt **Connect controller**.

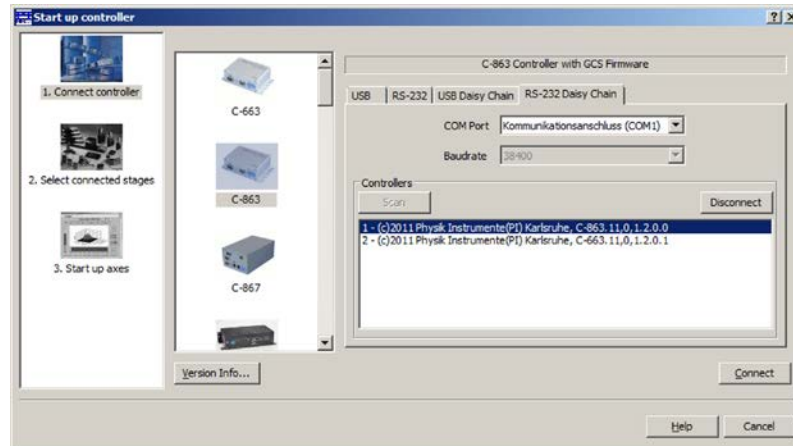
- Wenn sich das Fenster **Start up controller** nicht automatisch öffnet, wählen Sie im Hauptfenster den Menüeintrag **Connections > New....**
2. Wählen Sie im Feld für die Controllerauswahl den passenden Controllertyp aus.

Im Beispiel in den nachfolgenden Abbildungen besteht das Daisy-Chain-Netzwerk aus einem C-863.11 mit der Controlleradresse 1 und einem C-663.11 mit der Controlleradresse 2. Wenn Sie zuerst den C-863.11 verbinden möchten, wählen Sie **C-863**.



3. Wählen Sie auf der rechten Seite des Fensters die passende Registerkarte aus:
 - Wenn Sie den ersten Controller der Reihe über die RS-232-Schnittstelle an den PC angeschlossen haben, wählen Sie die Registerkarte **RS-232 Daisy Chain**.
 - Wenn Sie den ersten Controller der Reihe über die USB-Schnittstellen an den PC angeschlossen haben, wählen Sie die Registerkarte **USB Daisy Chain**.
4. Nehmen Sie auf der ausgewählten Registerkarte die Einstellungen für die Schnittstelle vor:
 - Registerkarte **RS-232 Daisy Chain**:
 - Wählen Sie im Feld **COM Port** den COM-Port des PC aus, an dem Sie den C-863 angeschlossen haben.
 - Stellen Sie im Feld **Baudrate** den Wert ein, der mit den DIP-Schaltern 5 und 6 des C-863 eingestellt ist.
 - Registerkarte **USB Daisy Chain**:
 - Wählen Sie im oberen Bereich der Registerkarte den angeschlossenen C-863 aus.

5. Klicken Sie im unteren Bereich der Registerkarte auf die Schaltfläche **Scan**, um alle Controller des Daisy-Chain-Netzwerks aufzulisten.

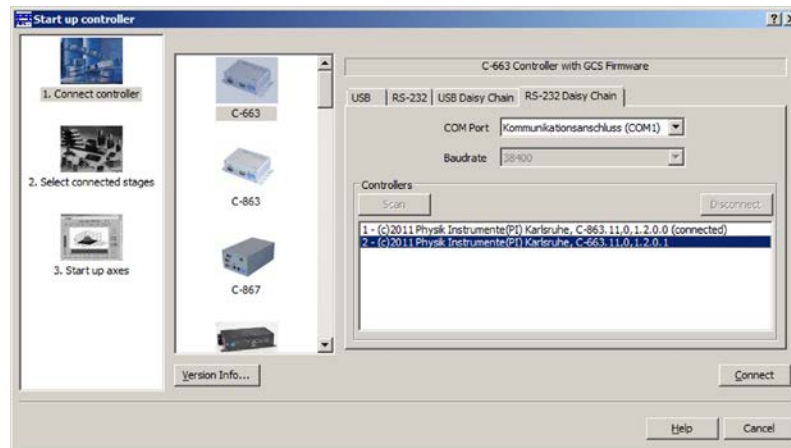


6. Wählen Sie einen Controller aus der Liste aus. Die Auswahl muss zum Controllertyp passen, den Sie in Schritt 2 ausgewählt haben.
7. Klicken Sie auf **Connect**, um die Kommunikation mit dem ausgewählten Controller herzustellen.

Wenn die Kommunikation erfolgreich hergestellt wurde, öffnet sich der Dialog **Select connected stages**:

- Gehen Sie weiter vor wie in "Bewegungen starten" (S. 87) beschrieben.
 - oder
 - Schließen Sie das Fenster **Select connected stages**, indem Sie auf die Schaltfläche **Cancel** klicken.
8. Wenn Sie einen weiteren Controller des Daisy-Chain-Netzwerks verbinden möchten, wählen Sie im Hauptfenster den Menüeintrag **Connections > New...**
 9. Führen Sie erneut die Schritte 2, 6 und 7 in der angegebenen Reihenfolge aus.

In der nachfolgenden Abbildung soll auch der **C-663** verbunden werden.



10. Wiederholen Sie die Schritte 8, 2, 6 und 7 für jeden weiteren Controller des Daisy-Chain-Netzwerks, den Sie verbinden möchten.

Wenn Sie die Kommunikation mit einem der Controller aus dem Daisy-Chain-Netzwerk beenden möchten:

- Wählen Sie im Hauptfenster den Menüeintrag **Connections > Close** für den entsprechenden Controller.

Kommunikation mit PITerminal herstellen

INFORMATION

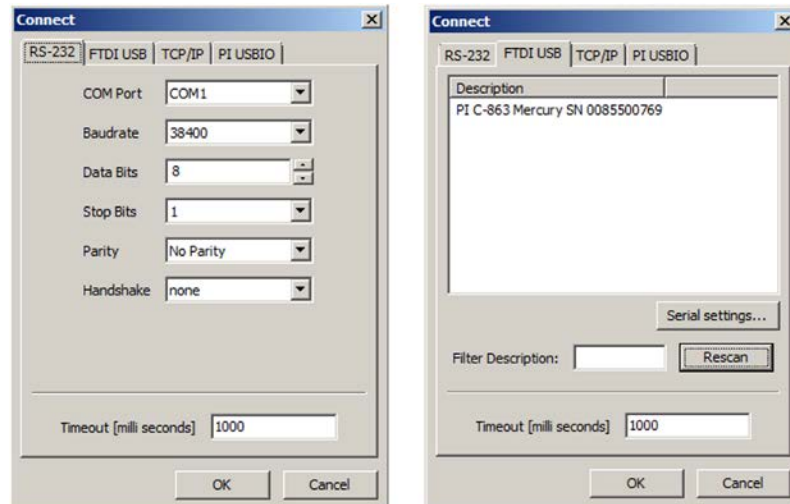
PITerminal unterstützt die Schaltfläche **Mercury** Controller mit älteren, nicht GCS-kompatiblen Firmware-Versionen.

- Stellen Sie im PITerminal sicher, dass **nicht** die Schaltfläche **Mercury** aktiviert ist.

1. Starten Sie PITerminal.
2. Klicken Sie auf **Connect....**

Das Fenster **Connect** öffnet sich.

3. Wählen Sie im Fenster **Connect** die Registerkarte **RS-232** oder **FTDI USB** aus, je nachdem, über welche die Schnittstelle Sie den ersten Controller in der Reihe an den PC angeschlossen haben.

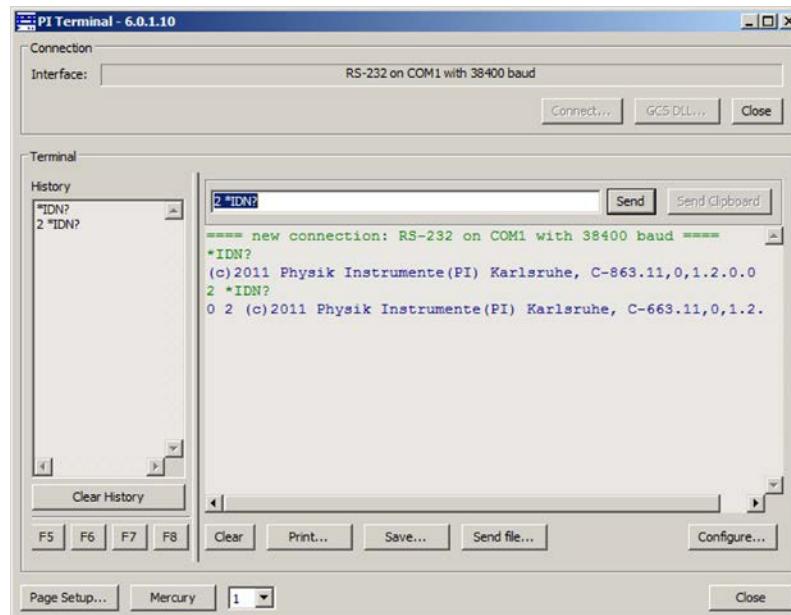


4. Nehmen Sie auf der ausgewählten Registerkarte die Einstellungen für die Schnittstelle vor:
 - Registerkarte **RS-232**:
 - Wählen Sie im Feld **COM Port** den COM-Port des PC aus, an dem Sie den C-863 angeschlossen haben.
 - Stellen Sie im Feld **Baudrate** den Wert ein, der mit den DIP-Schaltern 5 und 6 des C-863 eingestellt ist.
 - Registerkarte **FTDI USB**:
 - Wählen Sie den angeschlossenen C-863 aus.
5. Klicken Sie auf **OK**, um die Kommunikation herzustellen.
6. Senden Sie den Befehl ***IDN?** für alle Controller im Daisy-Chain-Netzwerk, um die Kommunikation zu prüfen.

Im Beispiel in der nachfolgenden Abbildung besteht das Daisy-Chain-Netzwerk aus einem C-863.11 mit der Controlleradresse 1 und einem C-663.11 mit der Controlleradresse 2. Sie senden:

- ***IDN?**, um die Ident-Bezeichnung des Controllers mit der Adresse 1 abzufragen; die Controlleradresse ist nicht erforderlich (da = 1)
- **2 *IDN?**, um die Ident-Bezeichnung des Controllers mit der Adresse 2 abzufragen.

Weitere Informationen siehe "Empfänger- und Senderadresse" (S. 158).



7.5 Bewegungen starten

Im Folgenden wird PIMikroMove® verwendet, um den Versteller zu bewegen. Das Programm leitet Sie dabei durch die folgenden Schritte, so dass Sie sich nicht mit den entsprechenden GCS-Befehlen auseinandersetzen müssen:

- Anpassung der Parametereinstellungen des C-863 an den angeschlossenen Versteller durch Laden eines Parametersatzes aus einer Verstellerdatenbank
- Einschalten des Servomodus (geregelter Betrieb)
- Ausführen einer Referenzfahrt; Details siehe "Referenzwertbestimmung" (S. 40).

HINWEIS



Auswahl eines falschen Verstellertyps

Die Auswahl eines falschen Verstellertyps in der PC-Software kann Schäden am Versteller verursachen.

- Stellen Sie sicher, dass der in der PC-Software ausgewählte Verstellertyp mit dem angeschlossenen Versteller übereinstimmt.

HINWEIS**Schwingungen!**

Ungeeignete Einstellungen der Regelparameter des C-863 können den Versteller zum Schwingen bringen. Schwingungen können den Versteller und/oder die auf ihm angebrachte Last beschädigen.

- Befestigen Sie den Versteller und alle Lasten ausreichend.
- Wenn der Versteller schwingt (ungewöhnliches Laufgeräusch), schalten Sie den Servomodus sofort aus oder trennen Sie den C-863 von der Stromversorgung.
- Schalten Sie den Servomodus erst wieder ein, nachdem Sie die Einstellungen der Regelparameter des C-863 geändert haben; siehe "Regelparameter optimieren" (S. 96).
- Wenn aufgrund einer sehr hohen Last Schwingungen bereits während der Referenzfahrt auftreten, folgen Sie für die Referenzfahrt den Anweisungen in "Störungsbehebung" (S. 303).

INFORMATION

Parameterwerte werden aus der Verstellerdatenbank nur in den flüchtigen Speicher des Controllers geladen.

INFORMATION

Der C-863 verfügt über einen permanenten Speicher für Parameterwerte. Deshalb können nach dem Einschalten bereits die korrekten Parametereinstellungen für den angeschlossenen Versteller geladen sein.

- Wenn Sie trotzdem einen Parametersatz aus der Verstellerdatenbank geladen und damit die ursprünglichen Einstellungen des C-863 im flüchtigen Speicher überschrieben haben, vermeiden Sie das Speichern der neuen Einstellungen im permanenten Speicher des C-863. Nach dem Aus- und Wiedereinschalten oder einem Neustart des C-863 sind wieder die ursprünglichen Einstellungen aktiv.

INFORMATION

Wenn in PIMikroMove® nicht der Schritt **Select connected stages** angezeigt wird, hat der Controller wahrscheinlich schon die korrekten Parametereinstellungen für den angeschlossenen Verstellertyp geladen.

1. Prüfen Sie im Schritt **Start up axes**, ob in der Spalte **Stage** im mittleren Bereich des Fensters der korrekte Verstellertyp steht.
2. Wenn der Verstellertyp nicht korrekt ist, klicken Sie im linken Bereich des Fensters **Start up controller** auf **Select connected stages**, um die Auswahl des Verstellertyps ändern zu können.

Voraussetzungen

- ✓ Sie haben die allgemeinen Hinweise zur Inbetriebnahme gelesen und verstanden (S. 73).
- ✓ PIMikroMove® ist auf dem PC installiert (S. 57).
- ✓ Sie haben das PIMikroMove® Handbuch gelesen und verstanden. Das Handbuch finden Sie auf der Produkt-CD.
- ✓ Sie haben die neueste Version der Verstellerdatenbank *PIUserStages2.dat* auf dem PC installiert (S. 59).
- ✓ Wenn Sie für Ihren Versteller eine kundenspezifische Verstellerdatenbank von PI erhalten haben, dann haben Sie diese Datenbank auf Ihrem PC installiert (S. 60).
- ✓ Sie haben den Versteller so installiert, wie er in Ihrer Anwendung eingesetzt wird (entsprechende Last und Ausrichtung).
- ✓ Sie haben den Versteller an den C-863 angeschlossen (S. 64).
- ✓ Sie haben die Kommunikation zwischen dem C-863 und dem PC mit PIMikroMove® hergestellt (S. 78).

Bewegungen starten mit PIMikroMove®

1. Wenn in PIMikroMove® der Schritt **Select connected stages** angezeigt wird, wählen Sie den Verstellertyp des angeschlossenen Verstellers aus:

Wenn rechts im Fenster in der Liste **Controller axes** bereits der korrekte Verstellertyp in der Spalte **Current stage type** eingetragen ist:

- Klicken Sie auf **OK**.

Wenn der eingetragene Verstellertyp nicht korrekt ist:

- Markieren Sie den Verstellertyp in der Liste **Stage database entries**.
- Klicken Sie auf **Assign**.
- Bestätigen Sie die Auswahl mit **OK**, um die Parametereinstellungen für den ausgewählten Verstellertyp aus der Verstellerdatenbank in den flüchtigen Speicher des C-863 zu laden.

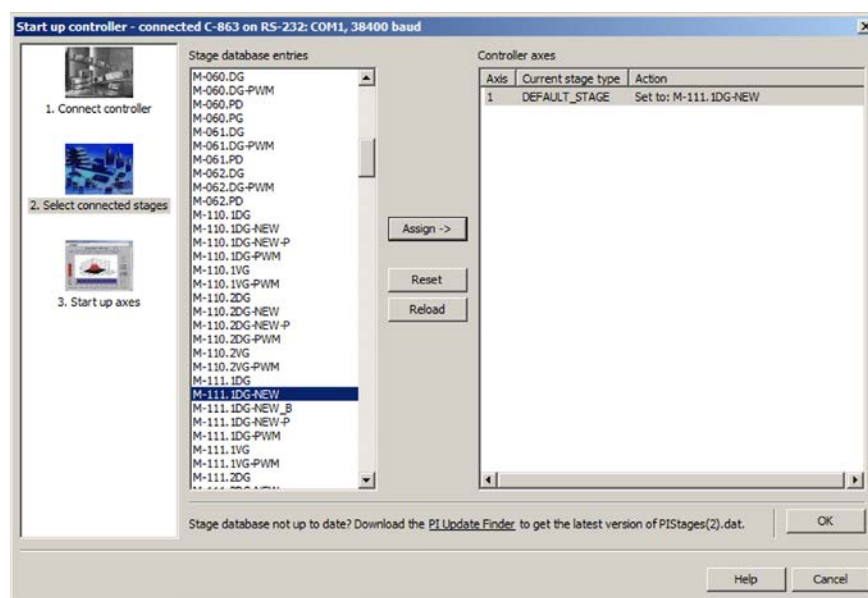


Abbildung 17: Start up controller – Select connected stages

Nach dem Anklicken von **OK** wechselt das Fenster **Start up controller** zum Schritt **Start up axes**.

- Führen Sie im Schritt **Start up axes** die Referenzfahrt für die Achse aus, damit der Controller die absolute Achsenposition kennt:
 - Wenn Sie die Referenzfahrt zum Referenzschalter starten wollen, klicken Sie auf **Ref. switch**.
 - Wenn Sie die Referenzfahrt zum negativen Endschalter starten wollen, klicken Sie auf **Neg. limit**.
 - Wenn Sie die Referenzfahrt zu positiven Endschalter starten wollen, klicken Sie auf **Pos. limit**.

Wenn eine Warnmeldung erscheint, dass der Servomodus ausgeschaltet ist:

- Schalten Sie den Servomodus durch einen Klick auf die Schaltfläche **Switch on servo** ein (geregelter Betrieb).

Die Achse führt die Referenzfahrt aus.

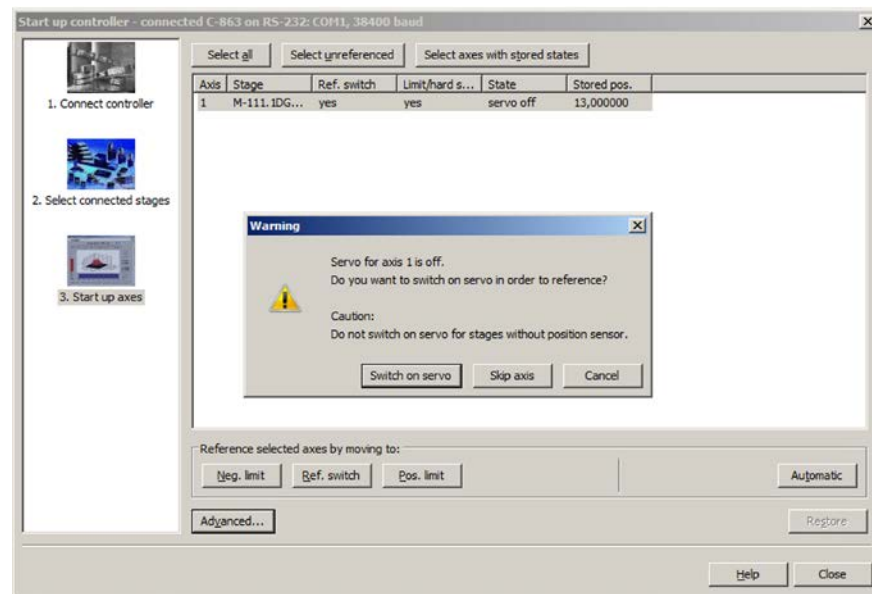


Abbildung 18: Start up controller – Start up axes

3. Nach erfolgreicher Referenzfahrt klicken Sie auf **OK** > **Close**.

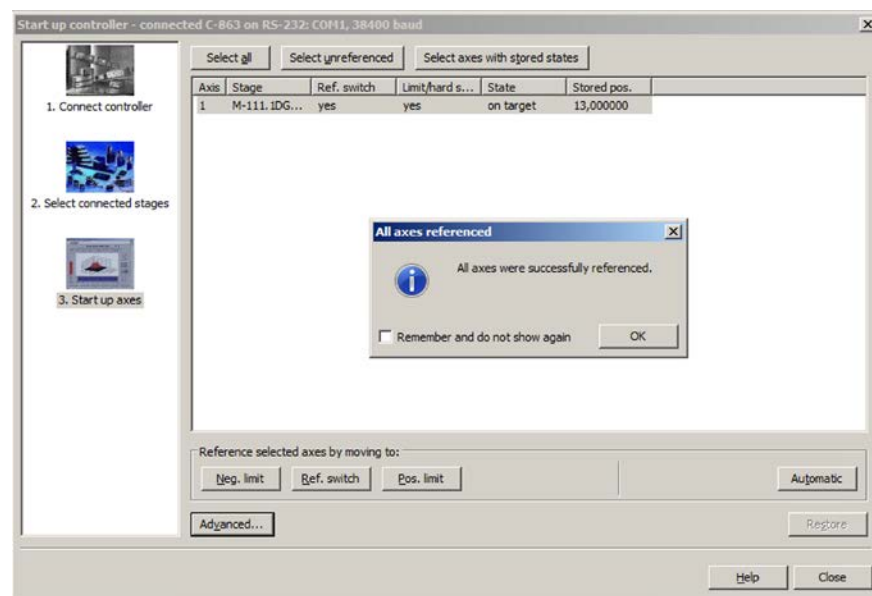


Abbildung 19: Start up controller – All axes referenced

Das Hauptfenster von PIMikroMove® öffnet sich.

4. Starten Sie einige Testbewegungen der Achse.

Im Hauptfenster von PIMikroMove® können Sie z. B. Schritte mit einer bestimmten Schrittweite ausführen, indem Sie auf die entsprechenden Pfeiltasten für die Achse klicken.

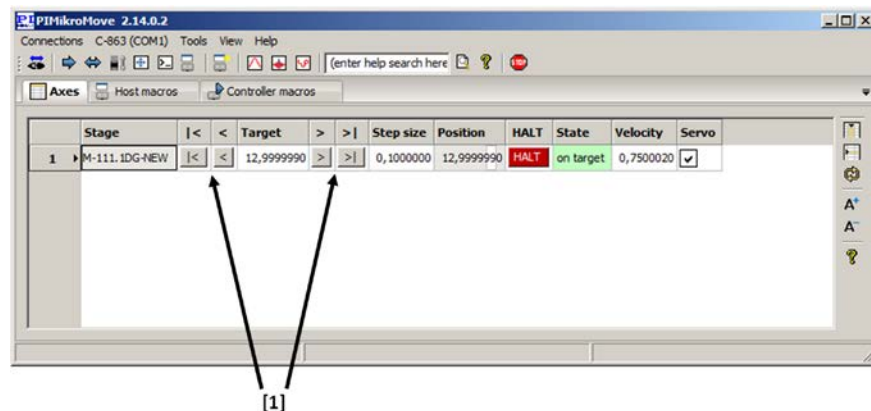


Abbildung 20: Hauptfenster von PIMikroMove®; [1] Pfeiltasten für Bewegung

7.6 Sprungantwort aufzeichnen

Mit der Aufzeichnung der Sprungantwort ermitteln Sie das Einschwingverhalten des Verstellers im geregelten Betrieb. Im Folgenden ist das Vorgehen für PIMikroMove® beschrieben.

Voraussetzung

- ✓ Sie haben mit PIMikroMove® erste Bewegungen gestartet (S. 87).
- ✓ Alle Geräte sind noch betriebsbereit.

Sprungantwort messen

1. Öffnen Sie im Hauptfenster von PIMikroMove® das Fenster **Data Recorder** über den Menüeintrag **C-863 > Show/Hide data recorder**.

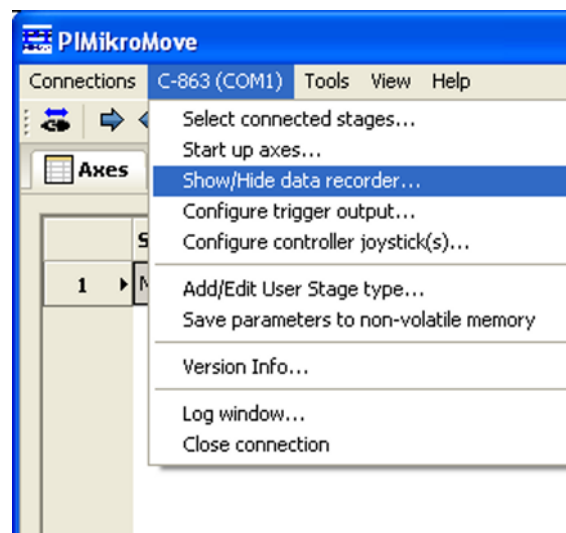


Abbildung 21: PIMikroMove: Show/Hide data recorder

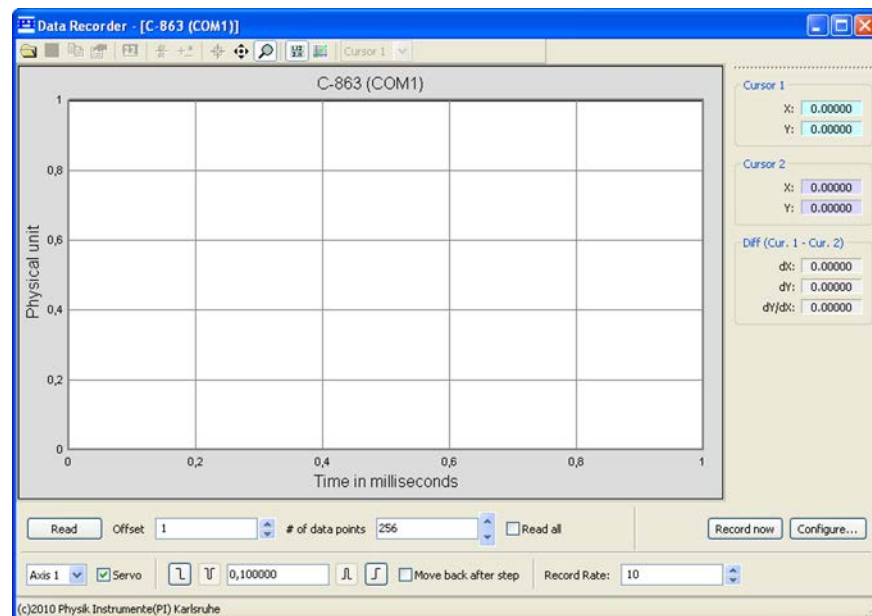


Abbildung 22: PIMikroMove: Datenrekorder

2. Stellen Sie mit dem Kontrollkästchen **Servo** sicher, dass der Servomodus eingeschaltet ist.

- Wenn das Kontrollkästchen **Servo** nicht markiert ist, ist der Servomodus ausgeschaltet. Markieren Sie das Kontrollkästchen, um den Servomodus anzuschalten.
3. Konfigurieren Sie den Datenrekorder.
 - a) Passen Sie im Fenster **Data Recorder** nach Bedarf Einstellungen wie die Amplitude des auszuführenden Sprungs, die Aufzeichnungsrate und die Anzahl der für die grafische Darstellung auszulesenden Datenpunkte an.
 - b) Klicken Sie auf die Schaltfläche **Configure...** und stellen Sie sicher, dass als aufzuzeichnende Größen "Commanded Position of Axis" und "Actual Position of Axis" ausgewählt sind.

Einzelheiten siehe PIMikroMove® Benutzerhandbuch (SM148E).

4. Starten Sie den Sprung in positive Richtung sowie die Aufzeichnung durch Anklicken der Schaltfläche .

Die Achse führt den Sprung aus, und die Sprungantwort wird aufgezeichnet und grafisch dargestellt.

5. Überprüfen Sie die dargestellte Sprungantwort anhand der nachfolgenden Abbildungen.
 - a) Wenn das Ergebnis zufriedenstellend ist (d. h. geringstmögliches Überschwingen, Einschwingzeit nicht zu lang), verfügen Sie bereits über optimale Parametereinstellungen. In diesem Fall brauchen Sie nichts weiter zu unternehmen.
 - b) Wenn das Ergebnis nicht zufriedenstellend ist, ändern Sie die Einstellungen des P-I-D-Reglers (S. 96).

Beispiele

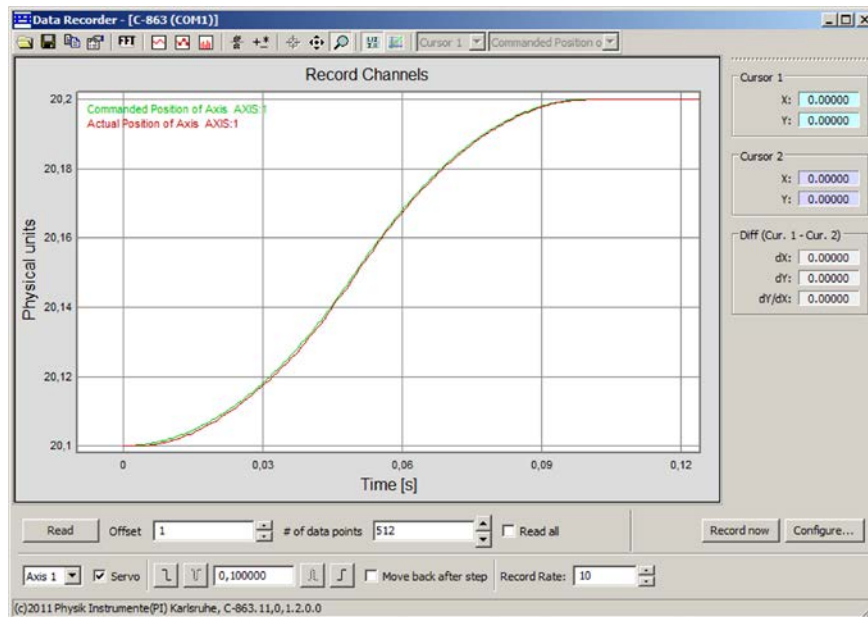


Abbildung 23: PIMikroMove®: Datenrekorder mit grafischer Darstellung eines optimalen Einschwingverhaltens (kommandierte und tatsächliche Position deckungsgleich)

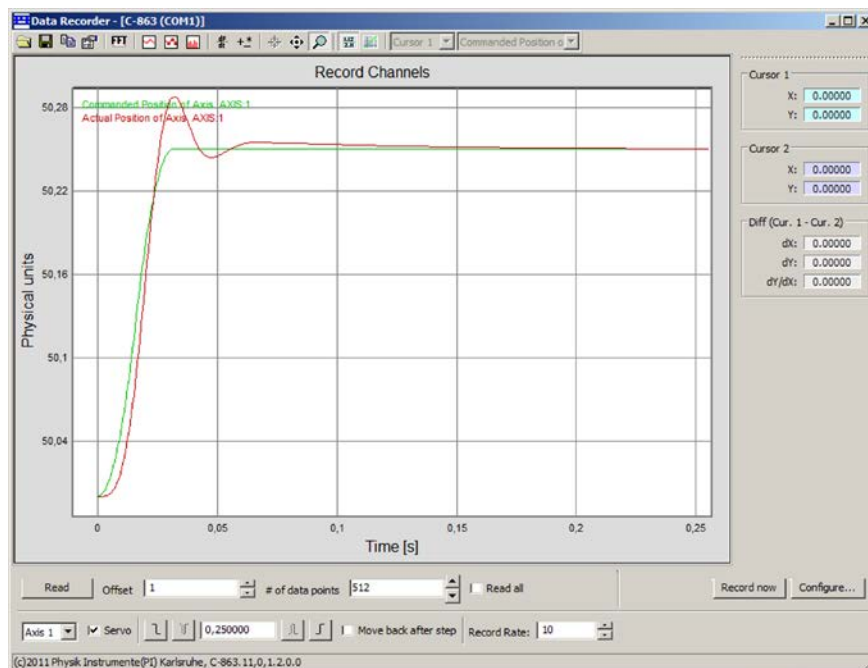


Abbildung 24: PIMikroMove®: Datenrekorder mit grafischer Darstellung eines Überschwingens

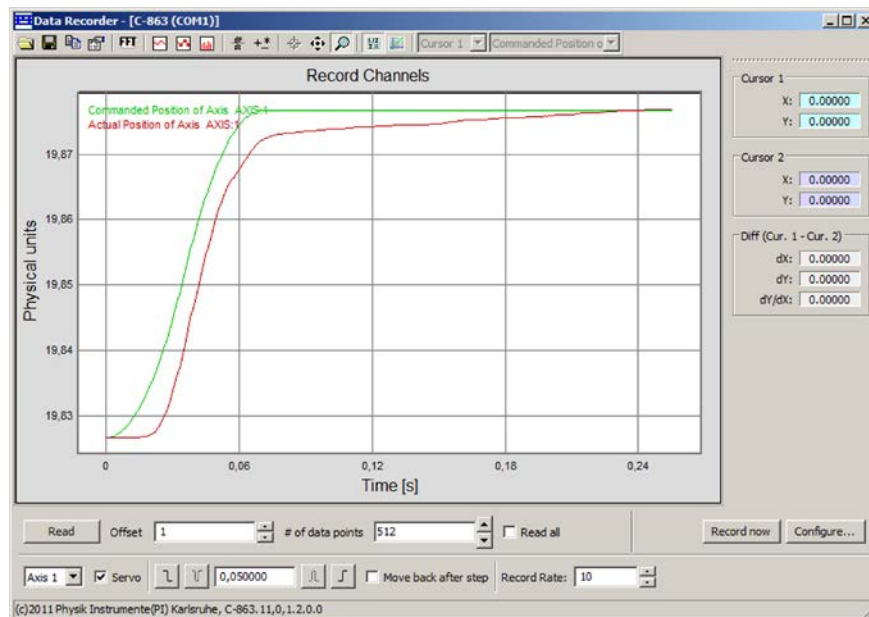


Abbildung 25: PIMikroMove®: Datenrekorder mit grafischer Darstellung eines zu langsamen Einschwingens

7.7 Regelparameter optimieren

Durch die Einstellung des P-I-D-Reglers werden die dynamischen Eigenschaften des Systems (Überschwingen und Einschwingzeit) optimiert. Die optimale Einstellung des P-I-D-Reglers hängt von Ihrer Anwendung und Ihren Wünschen ab.

Typischerweise erfolgt die Optimierung empirisch und umfasst die folgenden Parameter. Details siehe "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 29):

- **P-Term** (0x1)
- **I-Term** (0x2)
- **D-Term** (0x3)
- **I-Limit** (0x4)

Dabei wird das Verhalten des Verstellers bei verschiedenen Werten im geregelten Betrieb beobachtet.

Im Folgenden wird PIMikroMove® für die Optimierung der P-I-D-Regelparameter verwendet.

Voraussetzung

- ✓ Die aufgezeichnete Sprungantwort (S. 92) des Verstellers ist nicht zufriedenstellend ausgefallen.
- oder
- ✓ Der Versteller schwingt (ungewöhnliches Laufgeräusch) mit den aktuellen Regelparametern.

P-I-D-Regler einstellen

1. Öffnen Sie im Hauptfenster von PIMikroMove® das Einzelachsen-Fenster für den angeschlossenen Versteller, indem Sie den Versteller im Menü **View > Single Axis Window** auswählen.
2. Erweitern Sie die Ansicht des Einzelachsen-Fensters, indem Sie auf die Schaltfläche > am rechten Fensterrand klicken.

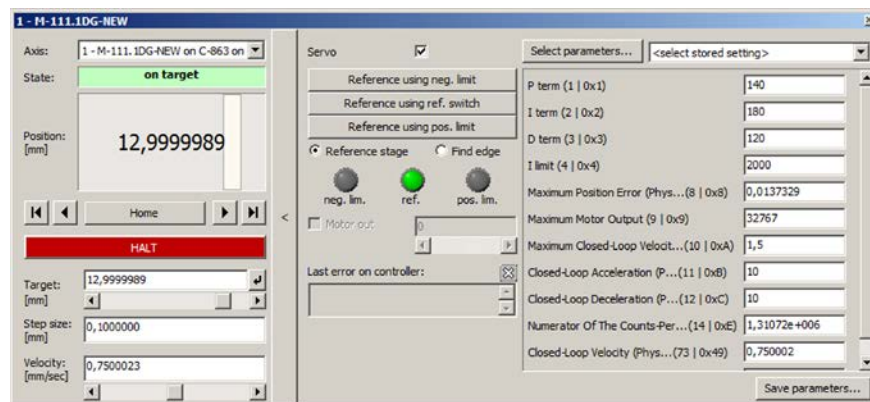


Abbildung 26: PIMikroMove: Erweitertes Einzelachsenfenster (Single Axis Window)

3. Geben Sie neue Werte für die Regelparameter ein:
 - a) Wenn der zu ändernde Parameter nicht in der Liste auf der rechten Seite des Fensters enthalten ist, klicken Sie auf **Select parameters...** und fügen ihn zur Liste hinzu.
 - b) Tippen Sie den neuen Parameterwert in das entsprechende Eingabefeld in der Liste ein.
 - c) Drücken Sie auf der Tastatur des PC die **Enter**-Taste, um den Parameterwert in den flüchtigen Speicher des Controllers zu übertragen. Dabei ändert der Eintrag seine Farbe von Blau in Schwarz.
4. Zeichnen Sie im Fenster **Data Recorder** erneut die Sprungantwort (S. 92) des Verstellers auf.

Wenn das Ergebnis nicht zufriedenstellend ist:

- Geben Sie andere Werte für die Regelparameter ein und zeichnen Sie die Sprungantwort erneut auf.

Wenn Sie mit dem Ergebnis zufrieden sind und die neuen Einstellungen der Regelparameter beibehalten wollen:

- Speichern Sie die neuen Einstellungen. Sie haben folgende Möglichkeiten:
 - Speichern Sie einen neuen Parametersatz in der Verstellerdatenbank PI_UserStages2.dat auf dem PC. Siehe "Verstellertyp anlegen oder ändern" (S. 284).
 - Übertragen Sie die aktuellen Werte **aller** Parameter aus dem flüchtigen in den permanenten Speicher des C-863 (S. 283).

8 Betrieb

In diesem Kapitel

Bewegungsfehler	99
Datenrekorder	101
Digitale Ausgangssignale	103
Digitale Eingangssignale	113
Analoge Eingangssignale	119
Joystick-Steuerung	121
Controllermakros	131

8.1 Bewegungsfehler

8.1.1 Schutzfunktionen des C-863

Bewegungsfehler können z. B. durch Störungen des Antriebs oder des Positionssensors des Verstellers verursacht werden.

Ein Bewegungsfehler liegt vor, wenn der Positionsfehler (d.h. der absolute Wert der Differenz zwischen der aktuellen Position und der kommandierten Position) im geregelten Betrieb den vorgegebenen Maximalwert überschreitet. Der Bereich, in dem die Abweichung liegen darf, ist durch den Parameter **Maximum Position Error (Phys. Unit)** (ID 0x8) festgelegt.

Wenn ein Bewegungsfehler auftritt, reagiert der C-863 wie folgt, um das System vor Schäden zu schützen:

- Der Servomodus wird für die betroffene Achse ausgeschaltet.
- Wenn vorhanden, wird die Bremse für die betroffene Achse aktiviert.
- Alle Bewegungen werden gestoppt.
- Der Fehlercode -1024 wird gesetzt.

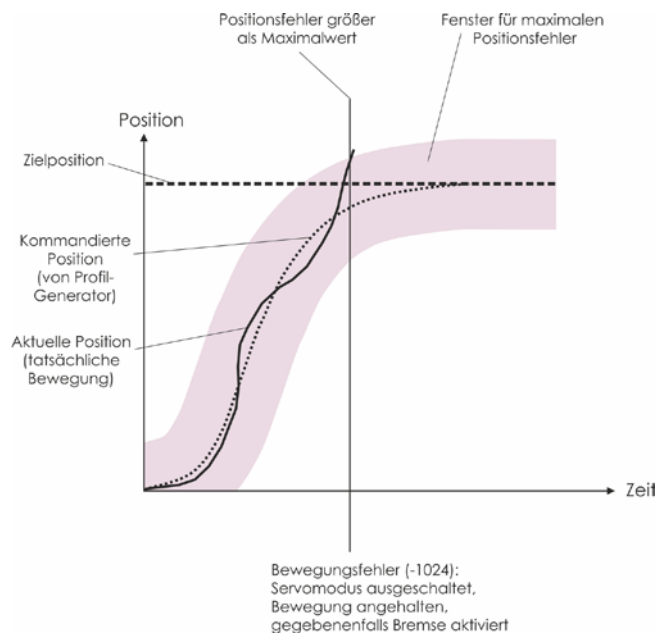


Abbildung 27: Verhalten bei Bewegungsfehlern

8.1.2 Betriebsbereitschaft wiederherstellen

HINWEIS



Unbeabsichtigte Bewegungen nach Bremsendeaktivierung!

Wenn der Servomodus ausgeschaltet ist, z. B. nach Auftreten eines Bewegungsfehlers, kann die Bremse des Verstellers per Befehl deaktiviert werden. Das Deaktivieren der Bremse kann unbeabsichtigte Bewegungen des Verstellers verursachen.

- Sichern Sie den Versteller gegen unbeabsichtigte Bewegungen, bevor Sie die Bremse per Befehl deaktivieren!

Betriebsbereitschaft wiederherstellen

1. Senden Sie den Befehl `ERR?`, um den Fehlercode auszulesen.

Wenn ein Bewegungsfehler vorliegt, wird der Fehlercode -1024 ausgegeben. `ERR?` setzt den Fehlercode bei der Abfrage auf null zurück.

2. Überprüfen Sie Ihr System und vergewissern Sie sich, dass alle Achsen gefahrlos bewegt werden können.
3. Schalten Sie mit dem Befehl **SV0** (S. 247) den Servomodus für die betroffene Achse ein.

Beim Einschalten des Servomodus wird die Zielposition auf die aktuelle Achsenposition gesetzt und gegebenenfalls die Bremse deaktiviert. Die Achse kann nun wieder bewegt werden, und Sie können eine neue Zielposition kommandieren.

INFORMATION

Mit den Befehlen **CT0** (S. 177) und **TR0** (S. 252) können Sie die digitalen Ausgangsleitungen des C-863 so programmieren, dass sie bei Bewegungsfehlern aktiviert werden. Die programmierten Ausgangsleitungen bleiben aktiv, bis der Fehlercode auf 0 zurückgesetzt wird. Details siehe "Triggermodus "Motion Error" einrichten" (S. 109).

8.2 Datenrekorder

8.2.1 Eigenschaften des Datenrekorders

Der C-863 enthält einen Echtzeit-Datenrekorder. Der Datenrekorder kann verschiedene Größen für die Achse (z. B. aktuelle Position) sowie die Signale der analogen Eingänge aufzeichnen.

Die aufgezeichneten Daten werden temporär in 2 Datenrekordertabellen mit jeweils 1024 Punkten gespeichert. Jede Datenrekordertabelle enthält die Daten von einer Datenquelle.

Sie können den Datenrekorder konfigurieren, indem Sie z. B. den aufzuzeichnenden Datentyp und die Datenquellen bestimmen und festlegen, wie die Aufzeichnung gestartet werden soll.

8.2.2 Datenrekorder einrichten

INFORMATION

Die Einstellungen zur Einrichtung des Datenrekorders lassen sich nur im flüchtigen Speicher des C-863 ändern. Nach dem Einschalten oder dem Neustart des C-863 sind werkseitige Standardeinstellungen aktiv, wenn nicht durch ein Startup-Makro bereits eine Konfiguration erfolgt.

Allgemeine Informationen über den Datenrekorder auslesen

- Senden Sie den Befehl **HDR?** (S. 203).

Die verfügbaren Aufzeichnungs- und Triggeroptionen sowie Informationen über zusätzliche Parameter und Befehle für die Datenaufzeichnung werden angezeigt.

Datenrekorder konfigurieren

Sie können den Datenrekordertabellen die Datenquellen und die Aufzeichnungsoptionen zuordnen.

- Senden Sie den Befehl **DRC?** (S. 190), um die aktuelle Konfiguration auszulesen. Datenrekordertabellen mit Aufzeichnungsoption 0 sind deaktiviert, d. h. es wird nichts aufgezeichnet. In der Standardeinstellung zeichnen die beiden Datenrekordertabellen des C-863 Folgendes auf:
 - Datenrekordertabelle 1: kommandierte Position der Achse
 - Datenrekordertabelle 2: aktuelle Position der Achse
- Konfigurieren Sie den Datenrekorder mit dem Befehl **DRC** (S. 189).

Sie können festlegen, wie die Aufzeichnung ausgelöst werden soll.

- Fragen Sie mit **DRT?** (S. 194) die aktuelle Triggeroption ab.
- Ändern Sie die Triggeroption mit dem Befehl **DRT** (S. 193). Die Triggeroption gilt für alle Datenrekordertabellen, deren Aufzeichnungsoption nicht auf 0 eingestellt ist.

Abtastintervall einstellen

- Senden Sie den Befehl **RTR?** (S. 234), um die Aufzeichnungsrate des Datenrekorders auszulesen.

Die Aufzeichnungsrate gibt an, nach wie vielen Servozyklen jeweils ein Datenpunkt aufgezeichnet wird. Der Standardwert beträgt 10 Servozyklen. Die Zykluszeit des C-863 beträgt 50 µs.

- Ändern Sie die Aufzeichnungsrate mit dem Befehl **RTR** (S. 234).

Mit zunehmender Aufzeichnungsrate erhöhen Sie die maximale Dauer der Datenaufzeichnung.

8.2.3 Aufzeichnung starten

- Starten Sie die Aufzeichnung durch die mit **DRT** eingestellte Triggeroption.

Unabhängig von der eingestellten Triggeroption wird die Datenaufzeichnung immer ausgelöst, wenn eine Sprungantwortmessung mit **STE** (S. 246) gestartet wird.

Die Datenaufzeichnung erfolgt immer für alle Datenrekordertabellen, deren Aufzeichnungsoption nicht auf 0 eingestellt ist. Sie endet, wenn die Datenrekordertabellen voll sind.

8.2.4 Aufgezeichnete Daten auslesen

INFORMATION

Das Auslesen der aufgezeichneten Daten kann abhängig von der Anzahl der Datenpunkte einige Zeit dauern.

Die Daten können auch bei laufender Datenaufzeichnung ausgelesen werden.

- Lesen Sie die zuletzt aufgezeichneten Daten mit dem Befehl **DRR?** aus.

Die Daten werden im GCS-Array-Format ausgegeben (siehe Benutzerhandbuch SM146E auf der Produkt-CD).

8.3 Digitale Ausgangssignale

Die digitalen Ausgänge des C-863 sind auf der Buchse **I/O** (S. 316) verfügbar.

- Fragen Sie die Anzahl der am C-863 verfügbaren Ausgangsleitungen mit dem Befehl **TI0?** (S. 250) ab.

Über die digitalen Ausgänge des C-863 können externe Geräte getriggert werden. Anwendungsmöglichkeiten:

- Koppeln der Triggerausgabe an die Bewegung der Achse. Details und Beispiele finden Sie in diesem Abschnitt.
- Direktes Schalten der Ausgangsleitungen, z. B. in Makros. Details und Beispiele zu Makros finden Sie unter "Controllermakros" (S. 131).

8.3.1 Befehle für digitale Ausgänge

Folgende Befehle stehen für die Verwendung digitaler Ausgänge zur Verfügung:

Befehl	Syntax	Funktion
CTO	CTO {<TrigOutID> <CTOPam> <Value>}	Konfiguriert die Bedingungen für die Triggerausgabe. Koppelt die Triggerausgabe an die Achsenbewegung.
DIO	DIO {<DIOID> <OutputOn>}	Schaltet digitale Ausgangsleitungen direkt in den Zustand low oder high, entweder einzeln oder alle Leitungen auf einmal. Sollte nicht für Ausgangsleitungen verwendet werden, bei denen die Triggerausgabe mit TRO aktiviert ist.
TRO	TRO {<TrigOutID> <TrigMode>}	Aktiviert oder deaktiviert die mit CTO gesetzten Bedingungen der Triggerausgabe. Standard: Triggerausgabe deaktiviert.

Pro Befehl CTO kann eine Konfigurationseinstellung vorgenommen werden:

CTO <TrigOutID> <CTOPam> <Value>

- <TrigOutID> ist eine digitale Ausgangsleitung des Controllers.
- <CTOPam> ist die CTO-Parameter-ID im Dezimalformat.
- <Value> ist der Wert, auf den der CTO-Parameter eingestellt ist.

Folgende Triggermodi (<Value>) können für <CTOPam> = 3 eingestellt werden:

<Value>	Triggermodus	Kurzbeschreibung
0 (Standard)	Position Distance	Sobald sich die Achse um eine vorgegebene Strecke bewegt hat, wird ein Triggerpuls ausgegeben (S. 106). Optional können Start- und Stoppwerte definiert werden, um die Triggerung auf einen Positionsbereich und eine bestimmte Bewegungsrichtung (negativ oder positiv) zu beschränken.
2	On Target	Der On-Target-Status der gewählten Achse wird am gewählten Triggerausgang ausgegeben (S. 108).

<Value>	Triggermodus	Kurzbeschreibung
5	Motion Error	Die gewählte digitale Ausgangsleitung wird aktiv, wenn ein Bewegungsfehler auftritt (S. 109). Die Leitung bleibt aktiv, bis der Fehlercode auf 0 zurückgesetzt wird (durch eine Abfrage mit ERR?).
6	In Motion	Die gewählte digitale Ausgangsleitung ist solange aktiv, wie die gewählte Achse in Bewegung ist (S. 109).
7	Position+Offset	Der erste Triggerpuls wird ausgegeben, wenn die Achse eine vorgegebene Position erreicht hat. Die nächsten Triggerpulse werden jeweils ausgegeben, wenn die Achsenposition gleich der Summe der letzten gültigen Triggerposition und einer vorgegebenen Strecke ist. Die Triggerausgabe wird beim Erreichen eines Stoppwertes beendet. Das Vorzeichen des Streckenwertes bestimmt, für welche Bewegungsrichtung Triggerpulse ausgegeben werden sollen. Details siehe "Triggermodus "Position + Offset" einrichten" (S. 110).
8	Single Position	Die gewählte digitale Ausgangsleitung ist aktiv, wenn die Achsenposition eine vorgegebene Position erreicht hat oder überschreitet (S. 112).

Darüber hinaus kann die Polarität (high-aktiv / low-aktiv) des Signals am digitalen Ausgang eingestellt werden (S. 113).

INFORMATION

Die Einstellungen zur Konfiguration der digitalen Ausgangsleitungen lassen sich nur im flüchtigen Speicher des C-863 ändern. Nach dem Einschalten oder dem Neustart des C-863 sind werkseitige Standardeinstellungen aktiv, sofern nicht durch ein Startup-Makro bereits eine Konfiguration erfolgt.

8.3.2 Triggermodus "Position Distance" einrichten

Der Triggermodus *Position Distance* eignet sich für Scananwendungen. Sobald sich die Achse die Strecke weiterbewegt hat, die mit der CTO-Parameter-ID = 1 (TriggerStep) eingestellt wurde, wird ein Triggerpuls ausgegeben. Die Pulsweite beträgt einen Servozyklus (50 µs).

Die Einheit der Strecke (TriggerStep) hängt von den Einstellungen der Parameter 0xE und 0xF ab. Standard ist mm.

1. Konfigurieren Sie die digitale Ausgangsleitung (<TrigOutID>), die als Triggerausgang verwendet werden soll:
 - Senden Sie CTO <TrigOutID> 2 A, wobei A die zu bewegendende Achse bezeichnet.
 - Senden Sie CTO <TrigOutID> 3 0, wobei 0 den Triggermodus *Position Distance* bestimmt.
 - Senden Sie CTO <TrigOutID> 1 S, wobei S die Strecke bezeichnet.
2. Wenn Sie die Bedingungen für die Triggerausgabe aktivieren wollen, senden Sie TR0 <TrigOutID> 1.

Beispiel:

Auf der digitalen Ausgangsleitung 1 wird immer dann ein Puls ausgegeben, wenn die Achse 1 des Verstellers eine Strecke von 0,1 µm zurückgelegt hat.

➤ Senden Sie:

CTO 1 2 1

CTO 1 3 0

CTO 1 1 0.0001

TR0 1 1

Triggermodus "Position Distance" mit Start- und Stoppwerten für positive Achsenbewegungsrichtung

Optional können Sie für die Triggerung Start- und Stoppwerte zur Begrenzung des Bereichs und zur Bestimmung der Achsenbewegungsrichtung (positiv oder negativ) festlegen.

INFORMATION

Wenn Start- und Stoppwert denselben Wert haben, werden sie ignoriert.

Wenn sich die Bewegungsrichtung umkehrt, bevor die Achsenposition den Stoppwert erreicht hat, werden weiterhin Triggerpulse ausgegeben.

1. Konfigurieren Sie die digitale Ausgangsleitung (<TrigOutID>), die als Triggerausgang verwendet werden soll:
 - Senden Sie `CT0 <TrigOutID> 2 A`, wobei *A* die zu bewegendende Achse bezeichnet.
 - Senden Sie `CT0 <TrigOutID> 3 0`, wobei 0 den Triggermodus *Position Distance* bestimmt.
 - Senden Sie `CT0 <TrigOutID> 1 S`, wobei *S* die Strecke bezeichnet.
 - Senden Sie `CT0 <TrigOutID> 8 Start`, wobei *Start* den Startwert bezeichnet.
 - Senden Sie `CT0 <TrigOutID> 9 Stopp`, wobei *Stopp* den Stoppwert bezeichnet.
2. Wenn Sie die Bedingungen für die Triggerausgabe aktivieren wollen, senden Sie `TR0 <TrigOutID> 1`.

Beispiel

Auf der digitalen Ausgangsleitung 1 wird immer dann ein Puls ausgegeben, wenn die Achse 1 des Verstellers eine Strecke von 0,1 µm zurückgelegt hat, solange sich Achse 1 in positiver Bewegungsrichtung im Bereich von 0,2 µm bis 0,55 µm bewegt (Startwert < Stoppwert).

- Senden Sie:

`CT0 1 2 1`

`CT0 1 3 0`

`CT0 1 1 0.0001`

`CT0 1 8 0.0002`

`CT0 1 9 0.00055`

`TR0 1 1`

Triggermodus "Position Distance" mit Start- und Stoppwerten für negative Achsenbewegungsrichtung

Im Folgenden ist das Beispiel von oben mit vertauschten Start- und Stoppwerten gezeigt. Die Triggerung erfolgt in negativer Achsenbewegungsrichtung (Stoppwert < Startwert) im Bereich zwischen 0,55 µm und 0,2 µm.

Beispiel:

- Senden Sie:

```
CT0 1 2 1
```

```
CT0 1 3 0
```

```
CT0 1 1 0.0001
```

```
CT0 1 8 0.00055
```

```
CT0 1 9 0.0002
```

```
TR0 1 1
```

8.3.3 Triggermodus "On Target" einrichten

Im Triggermodus *On Target* wird der On-Target-Status der gewählten Achse (S. 32) am gewählten Triggenergang ausgegeben.

1. Konfigurieren Sie die digitale Ausgangsleitung (<TrigOutID>), die als Triggenergang verwendet werden soll:
 - Senden Sie `CT0 <TrigOutID> 2 A`, wobei A die zu bewegend Achse bezeichnet.
 - Senden Sie `CT0 <TrigOutID> 3 2`, wobei 2 den Triggermodus *On Target* bestimmt.
2. Wenn Sie die Bedingungen für die Triggenergabe aktivieren wollen, senden Sie `TR0 <TrigOutID> 1`.

Beispiel:

Der On-Target-Status von Achse 1 soll auf der digitalen Ausgangsleitung 1 ausgegeben werden.

- Senden Sie:

```
CT0 1 2 1
```

```
CT0 1 3 2
```

```
TR0 1 1
```

8.3.4 Triggermodus "Motion Error" einrichten

Der Triggermodus *Motion Error* eignet sich für die Überwachung von Bewegungen. Die gewählte digitale Ausgangsleitung wird aktiv, wenn auf einer der angeschlossenen Achsen ein Bewegungsfehler auftritt. Die Leitung bleibt aktiv, bis der Fehlercode auf 0 zurückgesetzt wird (durch eine Abfrage mit **ERR?**).

INFORMATION

Ein Bewegungsfehler liegt vor, wenn die aktuelle Position während der Bewegung zu sehr von der kommandierten Position abweicht.

Weitere Informationen siehe "Bewegungsfehler" (S. 99).

1. Konfigurieren Sie die digitale Ausgangsleitung (<TrigOutID>), die als Triggerausgang verwendet werden soll:
 - Senden Sie **CT0 <TrigOutID> 3 5**, wobei 5 den Triggermodus *Motion Error* bestimmt.
2. Wenn Sie die Bedingungen für die Triggerausgabe aktivieren wollen, senden Sie **TR0 <TrigOutID> 1**.

8.3.5 Triggermodus "In Motion" einrichten

Im Triggermodus *In Motion* wird der Bewegungsstatus der gewählten Achse am gewählten Triggerausgang ausgegeben. Die Leitung ist aktiv, solange die gewählte Achse in Bewegung ist.

Der Bewegungsstatus kann auch mit den Befehlen **#5** (S. 167), **#4** (S. 166) und **SRG?** (S. 244) ausgelesen werden.

INFORMATION

Wenn die Achse in Bewegung ist, dann ist Bit 14 des Statusregisters 1 der Achse gesetzt.

1. Konfigurieren Sie die digitale Ausgangsleitung (<TrigOutID>), die als Triggerausgang verwendet werden soll:
 - Senden Sie **CT0 <TrigOutID> 2 A**, wobei A die zu bewegende Achse bezeichnet.

- Senden Sie `CT0 <TrigOutID> 3 6`, wobei 6 den Triggermodus *In Motion* bestimmt.
- 2. Wenn Sie die Bedingungen für die Triggerausgabe aktivieren wollen, senden Sie `TR0 <TrigOutID> 1`.

Beispiel:

Die digitale Ausgangsleitung 1 soll aktiv sein, wenn die Achse 1 des Verstellers in Bewegung ist.

- Senden Sie:

`CT0 1 2 1`

`CT0 1 3 6`

`TR0 1 1`

8.3.6 Triggermodus "Position + Offset" einrichten

Der Triggermodus *Position+Offset* eignet sich für Scananwendungen. Der erste Triggerpuls wird ausgegeben, wenn die Achse eine vorgegebene Position erreicht hat (TriggerPosition). Die nächsten Triggerpulse werden jeweils ausgegeben, wenn die Achsenposition gleich der Summe der letzten gültigen Triggerposition und einer vorgegebenen Strecke (TriggerStep) ist. Die Triggerausgabe wird beim Erreichen eines Stoppwertes beendet. Das Vorzeichen des Streckenwertes bestimmt, für welche Bewegungsrichtung Triggerpulse ausgegeben werden sollen.

Die Pulsweite beträgt einen Servozyklus (50 µs).

Die Einheit für TriggerPosition, TriggerStep und Stoppwert hängt von den Einstellungen der Parameter 0xE und 0xF ab. Standard ist mm.

1. Konfigurieren Sie die digitale Ausgangsleitung (<TrigOutID>), die als Triggerausgang verwendet werden soll:
 - Senden Sie `CT0 <TrigOutID> 2 A`, wobei A die zu bewegende Achse bezeichnet.
 - Senden Sie `CT0 <TrigOutID> 3 7`, wobei 7 den Triggermodus *Position+Offset* bestimmt.
 - Senden Sie `CT0 <TrigOutID> 1 S`, wobei S die Strecke bezeichnet.
 - Senden Sie `CT0 <TrigOutID> 10 TriPos`, wobei *TriPos* die Position für die Ausgabe des ersten Triggerpulses bezeichnet.

- Senden Sie `CT0 <TrigOutID> 9 Stopp`, wobei *Stopp* den Stoppwert bezeichnet.
- 2. Wenn Sie die Bedingungen für die Triggerausgabe aktivieren wollen, senden Sie `TR0 <TrigOutID> 1`.

Beispiel 1:

Auf der digitalen Ausgangsleitung 1 soll der erste Triggerpuls ausgegeben werden, wenn die absolute Position von Achse 1 1,5 mm beträgt. Danach soll auf dieser Leitung immer dann ein Puls ausgegeben werden, wenn Achse 1 eine Strecke von 0,1 μm in positiver Richtung zurückgelegt hat. Der letzte Triggerpuls soll ausgegeben werden, wenn die absolute Achsenposition 2,5 mm beträgt.

- Senden Sie:

```
CT0 1 2 1
```

```
CT0 1 3 7
```

```
CT0 1 1 0.0001
```

```
CT0 1 10 1.5
```

```
CT0 1 9 2.5
```

```
TR0 1 1
```

Beispiel 2:

Auf der digitalen Ausgangsleitung 2 soll der erste Triggerpuls ausgegeben werden, wenn die absolute Position von Achse B 0,4 mm beträgt. Danach soll auf dieser Leitung immer dann ein Puls ausgegeben werden, wenn Achse B eine Strecke von 1 μm in negativer Richtung zurückgelegt hat. Der letzte Triggerpuls soll ausgegeben werden, wenn die absolute Achsenposition 0,1 mm beträgt.

- Senden Sie:

```
CT0 2 2 B
```

```
CT0 2 3 7
```

```
CT0 2 1 -0.001
```

```
CT0 2 10 0.4
```

```
CT0 2 9 0.1
```

INFORMATION

Die Geschwindigkeitseinstellung der Achse muss für die mit dem Befehl **CT0** kommandierte Streckeneinstellung (TriggerStep) geeignet sein. Empfohlener Wert:
 Maximale Geschwindigkeit = Strecke * 20 kHz / 2
 wobei 20 kHz die Frequenz des Servozyklus des C-863 ist.

8.3.7 Triggermodus "Single Position" einrichten

Im Triggermodus *Single Position* ist die gewählte digitale Ausgangsleitung aktiv, wenn die Achsenposition eine vorgegebene Position (TriggerPosition) erreicht hat oder überschreitet.

Die Einheit für TriggerPosition hängt von den Einstellungen der Parameter 0xE und 0xF ab. Standard ist mm.

1. Konfigurieren Sie die digitale Ausgangsleitung (<TrigOutID>), die als Triggerausgang verwendet werden soll:
 - Senden Sie **CT0 <TrigOutID> 2 A**, wobei A die zu bewegendende Achse bezeichnet.
 - Senden Sie **CT0 <TrigOutID> 3 8**, wobei 8 den Triggermodus *Single Position* bestimmt.
 - Senden Sie **CT0 <TrigOutID> 10 TriPos**, wobei *TriPos* die Position bezeichnet, ab der die Ausgangsleitung aktiv sein soll.
2. Wenn Sie die Bedingungen für die Triggerausgabe aktivieren wollen, senden Sie **TR0 <TrigOutID> 1**.

Beispiel:

Die digitale Ausgangsleitung 1 soll aktiv sein, wenn die absolute Position von Achse 1 mindestens 1,5 mm beträgt.

- Senden Sie:

CT0 1 2 1

CT0 1 3 8

CT0 1 10 1.5

8.3.8 Signelpolarität einstellen

Mit dem CTO-Parameter *Polarity* kann die Polarität des Signals am digitalen Ausgang, der zur Triggerung verwendet wird, gewählt werden. Die Polarität kann folgende Werte haben:

- high-aktiv = 1 (Standardeinstellung)
- low-aktiv = 0
- Konfigurieren Sie die digitale Ausgangsleitung (<TrigOutID>), die als Triggerausgang verwendet werden soll:
 - Senden Sie `CTO <TrigOutID> 7 P`, wobei *P* die Polarität bezeichnet.

Beispiel:

Die Signelpolarität für die digitale Ausgangsleitung 1 soll auf low-aktiv eingestellt werden.

- Senden Sie:

`CTO 1 7 0`

8.4 Digitale Eingangssignale

Die digitalen Eingänge des C-863 sind auf der Buchse **I/O** (S. 316) verfügbar.

- Fragen Sie die Anzahl der am C-863 verfügbaren Eingangsleitungen mit dem Befehl `TI0?` (S. 250) ab.
- Fragen Sie den Status der Eingangsleitungen mit dem Befehl `DI0?` (S. 188) ab.

Anwendungsmöglichkeiten:

- Verwendung in Makros (S. 116). Details und Beispiele zu Makros finden Sie unter "Controllermakros" (S. 131).
- Verwendung als Schaltersignale (S. 117)

INFORMATION

Die digitalen Eingänge (Pins 1 bis 4) auf der Buchse **I/O** können auch als analoge Eingänge genutzt werden.

- Digital: TTL
- Analog: 0 bis +5 V

8.4.1 Befehle und Parameter für digitale Eingänge

Befehle

Folgende Befehle stehen für die Verwendung digitaler Eingänge zur Verfügung:

Befehl	Syntax	Funktion
CPY	CPY <Variable> <CMD?>	Kopiert in Kombination mit dem Abfragebefehl DIO? den Status einer digitalen Eingangsleitung in eine Variable. Verwendung in Makros zum Setzen lokaler Variablen (S. 160).
DIO?	DIO? [{<DIOID>}]	Fragt den Status digitaler Eingangsleitungen ab.
FED	FED {<AxisID> <EdgeID> <Param>}	Startet eine Fahrt zu einer Signalfanke. Die Quelle des Signals kann eine digitale Eingangsleitung sein.
FNL	FNL [{<AxisID>}]	Startet eine Referenzfahrt zur negativen physikalischen Grenze des Stellwegs. Eine digitale Eingangsleitung kann anstelle des negativen Endschafters als Quelle des negativen Endschaftersignals verwendet werden.
FPL	FPL [{<AxisID>}]	Startet eine Referenzfahrt zur positiven physikalischen Grenze des Stellwegs. Endschanter. Eine digitale Eingangsleitung kann anstelle des positiven Endschafters als Quelle des positiven Endschaftersignals verwendet werden.
FRF	FRF [{<AxisID>}]	Startet eine Referenzfahrt zum Referenzschalter. Eine digitale Eingangsleitung kann anstelle des Referenzschalters als Quelle des Referenzschaltersignals verwendet werden.
JRC	JRC <Jump> <CMD?> <OP> <Value>	Kann nur in Makros verwendet werden. Löst einen relativen Sprung des Makroausführungszeigers in Abhängigkeit vom Status einer digitalen Eingangsleitung aus, wenn in Kombination mit dem Abfragebefehl DIO? verwendet.
MEX	MEX <CMD?> <OP> <Value>	Kann nur in Makros verwendet werden. Stoppt Makroausführung in Abhängigkeit vom Status einer digitalen Eingangsleitung, wenn in Kombination mit dem Abfragebefehl DIO? verwendet.

Befehl	Syntax	Funktion
WAC	WAC <CMD?> <OP> <Value>	Kann nur in Makros verwendet werden. Wartet, bis eine digitale Eingangsleitung einen bestimmten Status erreicht, wenn in Kombination mit dem Abfragebefehl DIO? verwendet.

Parameter

Folgende Parameter stehen für die Konfiguration digitaler Eingänge zur Verfügung:

Parameter	Beschreibung und mögliche Werte
Source Of Reference Signal 0x5C	Bestimmt die Quelle des Referenzsignals für die Befehle FRF und FED: 0 = Referenzschalter 1 = Digitaler Eingang 1 2 = Digitaler Eingang 2 3 = Digitaler Eingang 3 4 = Digitaler Eingang 4
Source Of Negative Limit Signal 0x5D	Bestimmt die Quelle(n) des negativen Endschaltersignals für die Befehle FNL und FED über eine Bitmaske: 0 = Negativer Endschalter (Standard-Einstellung) 1 = Digitaler Eingang 1 (Bit 0) 2 = Digitaler Eingang 2 (Bit 1) 4 = Digitaler Eingang 3 (Bit 2) 8 = Digitaler Eingang 4 (Bit 3)
Source Of Positive Limit Signal 0x5E	Bestimmt die Quelle(n) des positiven Endschaltersignals für die Befehle FPL und FED über eine Bitmaske: 0 = Positiver Endschalter (Standard-Einstellung) 1 = Digitaler Eingang 1 (Bit 0) 2 = Digitaler Eingang 2 (Bit 1) 4 = Digitaler Eingang 3 (Bit 2) 8 = Digitaler Eingang 4 (Bit 3)

Parameter	Beschreibung und mögliche Werte
<i>Invert Digital Input Used For Negative Limit</i> 0x5F	Invertiert die Polarität der digitalen Eingänge, die als Quellen des negativen Endschaltersignals dienen, über eine Bitmaske: 0 = Kein digitaler Eingang invertiert (Standard-Einstellung). 1 = Digitaler Eingang 1 invertiert (Bit 0) 2 = Digitaler Eingang 2 invertiert (Bit 1) 4 = Digitaler Eingang 3 invertiert (Bit 2) 8 = Digitaler Eingang 4 invertiert (Bit 3)
<i>Invert Digital Input Used For Positive Limit</i> 0x60	Invertiert die Polarität der digitalen Eingänge, die als Quellen des positiven Endschaltersignals dienen, über eine Bitmaske: 0 = Kein digitaler Eingang invertiert (Standard-Einstellung). 1 = Digitaler Eingang 1 invertiert (Bit 0) 2 = Digitaler Eingang 2 invertiert (Bit 1) 4 = Digitaler Eingang 3 invertiert (Bit 2) 8 = Digitaler Eingang 4 invertiert (Bit 3)

8.4.2 Digitale Eingangssignale in Makros verwenden

Die digitalen Eingänge auf der Buchse **I/O** können in Makros wie folgt verwendet werden:

- Bedingte Ausführung des Makros
- Bedingtes Stoppen der Makroausführung
- Bedingter Sprung des Makroausführungs-Zeigers
- Kopieren des Eingangszustands in eine Variable

Weitere Informationen und Beispiele finden Sie unter "Controllermakros" (S. 131).

INFORMATION

Um die digitalen Eingangssignale für die Verwendung in Makros zu erzeugen, können Sie die Pushbutton-Box C-170.PB von PI an die Buchse **I/O** (S. 316) anschließen. Sie zeigt über LEDs auch den Status der digitalen Ausgangsleitungen an.

8.4.3 Digitale Eingangssignale als Schaltersignale verwenden

Die digitalen Eingänge auf der Buchse **I/O** können als Quelle der Referenz- und Endschaltersignale (z. B. für Referenzfahrten (S. 40)) für eine Achse verwendet werden.

Digitalen Eingang als Referenzsignal verwenden

INFORMATION

Der Pegel des digitalen Eingangssignals, das Sie anstelle des Referenzschalters verwenden, darf sich über den gesamten Stellweg hinweg nur einmal ändern.

- Verwenden Sie eine geeignete Signalquelle.
- Wenn nötig, invertieren Sie die Signallogik der digitalen Eingangsleitung durch entsprechendes Setzen des Parameters **Invert Reference?** (ID 0x31).

INFORMATION

Der Parameter **Has Reference?** (ID 0x14) hat keinen Einfluss auf die Verwendung einer digitalen Eingangsleitung als Quelle des Referenzsignals.

- Wählen Sie die Quelle des Referenzsignals für die Achse aus, indem Sie den Parameter **Source Of Reference Signal** (ID 0x5C) ändern.

Genaue Informationen zum Ändern von Parametern finden Sie in "Anpassen von Einstellungen" (S. 279).

Digitale Eingänge als Quelle der Endschaltersignale verwenden

INFORMATION

Mehrere digitale Eingänge können als Quelle für ein Endschaltersignal ausgewählt sein.

Wenn ein Endschaltersignal für Referenzfahrten verwendet wird, darf nur eine digitale Eingangsleitung als Quelle des Endschaltersignals ausgewählt sein.

INFORMATION

Der Pegel des digitalen Eingangssignals, das Sie anstelle eines eingebauten Endschalters verwenden, darf sich über den gesamten Stellweg hinweg nur einmal ändern.

- Verwenden Sie geeignete Signalquellen.
- Wenn nötig, invertieren Sie die Signallogik der digitalen Eingangsleitungen durch entsprechendes Setzen der Parameter **Invert Digital Input Used For Negative Limit** (ID 0x5F) und **Invert Digital Input Used For Positive Limit** (ID 0x60).

INFORMATION

Der Parameter **Has No Limit Switches?** (ID 0x32) legt fest, ob der C-863 die Signale der eingebauten Endschalter des Verstellers auswertet. Auf die Verwendung von digitalen Eingangsleitungen als Quelle des Endschaltersignals hat dieser Parameter keinen Einfluss.

- Wählen Sie die Quelle(n) des negativen Endschaltersignals für die Achse aus, indem Sie den Parameter **Source Of Negative Limit Signal** (ID 0x5D) ändern.
- Wählen Sie die Quelle(n) des positiven Endschaltersignals für die Achse aus, indem Sie den Parameter **Source Of Positive Limit Signal** (ID 0x5E) ändern.

Genaue Informationen zum Ändern von Parametern finden Sie in "Anpassen von Einstellungen" (S. 279).

Beispiel:

Die digitalen Eingangsleitungen 1, 3 und 4 sollen für Achse 1 als Quellen des positiven Endschaltersignals verwendet werden. Außerdem soll für Achse 1 die Signalpolarität der Leitungen 1 und 3 invertiert werden. Alle Anpassungen werden nur im flüchtigen Speicher des C-863 vorgenommen.

- Senden Sie:
 - SPA 1 0x5E 13, um die Leitungen 1, 3 und 4 auszuwählen.
 - SPA 1 0x60 5, um die Signalpolarität der Leitungen 1 und 3 zu invertieren.

8.5 Analoge Eingangssignale

Die analogen Eingänge des C-863 sind auf der Buchse **I/O** (S. 316) verfügbar.

- Fragen Sie die Anzahl der am C-863 verfügbaren analogen Eingangsleitungen mit dem Befehl **TAC?** (S. 249) ab.
- Fragen Sie die Spannung an den Analogeingängen mit dem Befehl **TAV?** (S. 249) ab.
- Verwenden Sie den Datenrekorder (S. 101), um die Signale der analogen Eingänge aufzuzeichnen.

Anwendungsmöglichkeiten:

- Verwendung in Makros (S. 120): Details und Beispiele zu Makros finden Sie unter "Controllermakros" (S. 131).
- Scan-Anwendungen mit PIMikroMove® (siehe PIMikroMove® Handbuch)

INFORMATION

Die analogen Eingänge (Pins 1 bis 4) auf der Buchse **I/O** können auch als digitale Eingänge genutzt werden.

- Analog: 0 bis +5 V
- Digital: TTL

8.5.1 Befehle für analoge Eingänge

Folgende Befehle stehen für die Verwendung analoger Eingänge zur Verfügung:

Befehl	Syntax	Funktion
CPY	CPY <Variable> <CMD?>	Kopiert in Kombination mit dem Abfragebefehl TAV? den Spannungswert einer analogen Eingangsleitung in eine Variable. Verwendung in Makros zum Setzen lokaler Variablen (S. 160).
DRC	DRC {<RecTableID> <Source> <RecOption>}	Konfiguriert den Datenrekorder. Mit der Aufzeichnungsoption 81 können die Werte der analogen Eingänge aufgenommen werden.

Befehl	Syntax	Funktion
JRC	JRC <Jump> <CMD?> <OP> <Value>	Kann nur in Makros verwendet werden. Löst einen relativen Sprung des Makroausführungszeigers in Abhängigkeit von der Spannung an einer analogen Eingangsleitung aus, wenn in Kombination mit dem Abfragebefehl TAV? verwendet.
MEX	MEX <CMD?> <OP> <Value>	Kann nur in Makros verwendet werden. Stoppt Makroausführung in Abhängigkeit von der Spannung an einer analogen Eingangsleitung, wenn in Kombination mit dem Abfragebefehl TAV? verwendet.
TAC?	TAC?	Fragt die Anzahl installierter Analogleitungen ab.
TAV?	TAV? [<AnalogInputID>]	Fragt die Spannung am Analogeingang ab.
WAC	WAC <CMD?> <OP> <Value>	Kann nur in Makros verwendet werden. Wartet, bis eine analoge Eingangsleitung eine bestimmte Spannung erreicht, wenn in Kombination mit dem Abfragebefehl TAV? verwendet.

8.5.2 Analoge Eingangssignale in Makros verwenden

Die analogen Eingänge auf der Buchse **I/O** können in Makros wie folgt verwendet werden:

- Bedingte Ausführung des Makros
- Bedingtes Stoppen der Makroausführung
- Bedingter Sprung des Makroausführungs-Zeigers
- Kopieren des Eingangszustands in eine Variable

Weitere Informationen und Beispiele finden Sie unter "Controllermakros" (S. 131).

8.6 Joystick-Steuerung

8.6.1 Funktionsweise der Joystick-Steuerung

Die Joystick-Achse steuert die Geschwindigkeit der am C-863 angeschlossenen Versteller-Achse (vom Profilgenerator ausgegebene kommandierte Geschwindigkeit).

Der Zusammenhang zwischen der Auslenkung der Joystick-Achse und der Geschwindigkeit der Versteller-Achse wird vom C-863 durch eine Lookup-Tabelle hergestellt. Die 256 Werte in der Lookup-Tabelle sind Faktoren, die während der Joysticksteuerung auf die mit dem Befehl **VEL** (S. 257) gesetzte Geschwindigkeit angewendet werden. Der Wertebereich reicht von -1.0000 bis 1.0000.

Die Firmware des Controllers bietet zwei vordefinierte Lookup-Tabellentypen (linear und parabolisch) zur Auswahl an und erlaubt das Füllen der Lookup-Tabelle mit individuellen Werten. Der Inhalt der Lookup-Tabelle ist automatisch im permanenten Speicher des C-863 gespeichert.

Während der Joystick-Steuerung wird als Zielposition die Verfahrbereichsgrenze verwendet, die durch den Parameter 0x15 bzw. 0x30 vorgegeben ist. Details siehe "Stellweg und Verfahrbereichsgrenzen" (S. 35). Bei Deaktivierung des Joysticks wird als neue Zielposition die aktuelle Position der joystickgesteuerten Achse gesetzt.

INFORMATION

Bewegungsbefehle sind nicht zulässig, wenn ein Joystick für die Achse aktiv ist. Im ungeregelten Betrieb (Servomodus aus) ist keine Joystick-Steuerung möglich.

8.6.2 Befehle und Parameter für Joystick-Steuerung

Befehle

Folgende Befehle stehen für die Verwendung des Joysticks zur Verfügung:

Befehl	Syntax	Funktion
JON	JON {<JoystickID> <uint>}	Aktiviert oder deaktiviert ein Joystick-Gerät, das an den Controller angeschlossen ist.
JON?	JON? [{<JoystickID>}]	Fragt den Status der Aktivierung des Joystick-Geräts ab.
JAX	JAX <JoystickID> <JoystickAxis> <AxisID>	Bestimmt die Achse, die durch einen an den Controller angeschlossenen Joystick gesteuert wird.

Befehl	Syntax	Funktion
JAX?	JAX? [{<JoystickID> <JoystickAxis>}]	Fragt die Achse ab, die durch einen an den Controller angeschlossenen Joystick gesteuert wird.
JAS?	JAS? [{<JoystickID> <JoystickAxis>}]	Fragt den aktuellen Status einer Joystick-Achse ab (Auslenkung).
JBS?	JBS? [{<JoystickID> <JoystickButton>}]	Fragt den aktuellen Status einer Joystick-Taste ab (gedrückt oder nicht gedrückt)
JDT	JDT {<JoystickID> <JoystickAxis> <uint>}	Bestimmt einen Standard-Lookup-Tabellentyp für eine Joystick-Achse.
JLT	JLT <JoystickID> <JoystickAxis> <Addr> <floatn>	Füllt die Lookup-Tabelle für eine Joystick-Achse mit individuellen Werten.
JLT?	JLT? [<StartPoint> <NumberOfPoints> [{<JoystickID> <JoystickAxis>}]]	Fragt die aktuell gültigen Werte der Lookup-Tabelle ab.

Parameter

Folgende Parameter stehen für die Verwendung des Joysticks zur Verfügung:

Parameter	Beschreibung und mögliche Werte
<i>Invert Direction Of Motion For Joystick-Controlled Axis?</i> 0x61	Bestimmt die Bewegungsrichtung für joystickgesteuerte Achsen. 0 = Bewegungsrichtung nicht invertiert (Standard-Einstellung) 1 = Bewegungsrichtung invertiert

8.6.3 Achsenbewegung steuern

Im Folgenden wird PIMikroMove® verwendet, um den Versteller über einen Joystick zu steuern. Sie müssen sich dabei nicht mit den entsprechenden GCS-Befehlen auseinandersetzen.

HINWEIS



Unbeabsichtigte Bewegungen bei Joystick-Aktivierung!

Wenn kein Joystick an den C-863 angeschlossen ist, kann die Aktivierung des Joysticks in der Software unbeabsichtigte Bewegungen der angeschlossenen Achse verursachen.

- Aktivieren Sie den Joystick in der Software nur, wenn tatsächlich ein Joystick an den C-863 angeschlossen ist.

INFORMATION

Der am C-863 angeschlossene Versteller kann nur durch eine Joystick-Achse gesteuert werden.

INFORMATION

Durch die Verwendung von Makros bieten sich vielfältige Einsatzmöglichkeiten für die Joystick-Steuerung. Insbesondere die Joystick-Taste kann in Makros für verschiedenste Zwecke genutzt werden. Details und Beispiele zu Makros finden Sie in "Controllermakros" (S. 131).

INFORMATION

Der C-863 unterstützt eine logische Achse und wird daher im Normalfall mit Verstellern eingesetzt, die nur eine Bewegungsachse haben. In diesem Fall ist die Bezeichnung "Achse" gleichbedeutend mit "Versteller". In den nachfolgenden Handlungsanleitungen wird daher nicht zwischen "Versteller" und "Achse" unterschieden.

Voraussetzung

- ✓ Sie haben den Versteller mit PIMikroMove® in Betrieb genommen und erste Bewegungen im geregelten Betrieb gestartet (S. 87).
- ✓ Sie haben einen Joystick an den C-863 angeschlossen (S. 68).

Achsenbewegung über einen Joystick steuern

1. Öffnen Sie im Hauptfenster von PIMikroMove® das Fenster **Configure Controller Joystick** über den Menüeintrag **C-863 > Configure controller joystick(s)...**



Abbildung 28: Configure Controller Joystick

Das Fenster **Configure Controller Joystick** listet die vom C-863 unterstützten Joystick-Geräte und deren Joystick-Achsen auf.

- Zeile 1 gehört zum Joystick-Gerät, das an Pin 4 der Buchse **Joystick** angeschlossen werden kann (z.B. Joystick C-819.20 oder C-819.30 von PI). Der C-863 unterstützt eine Joystick-Achse dieses Geräts.
 - Zeile 2 gehört zum Joystick-Gerät, das an Pin 2 der Buchse **Joystick** angeschlossen werden kann. Der C-863 unterstützt eine Joystick-Achse dieses Geräts.
2. Weisen Sie einer Joystick-Achse des angeschlossenen Joystick-Geräts die zu bewegend Achse zu:
 - a) Klicken Sie im Fenster **Configure Controller Joystick** auf die entsprechende Taste **Select**.
 - b) Markieren Sie im Fenster **Select controller axis** den korrekten Verstellernamen.
 - c) Klicken Sie im Fenster **Select controller axis** auf **OK**, um die Auswahl zu bestätigen und das Fenster zu schließen.
 3. Aktivieren Sie im Fenster **Configure Controller Joystick** das angeschlossene Joystick-Gerät, indem Sie das entsprechende Kontrollkästchen **Enable** markieren.

Wenn der Versteller sich bewegt, obwohl Sie den Joystick nicht betätigen:

- Prüfen Sie, ob der Joystick mechanisch arretiert ist.
 - Kalibrieren Sie den Joystick (S. 125).
4. Steuern Sie die Geschwindigkeit des Verstellers über den Joystick.

5. Wenn Sie die Joystick-Steuerung deaktivieren möchten, entfernen Sie im Fenster **Configure Controller Joystick** das Häkchen aus dem entsprechenden Kontrollkästchen **Enable**.

8.6.4 Joystick kalibrieren

Nach dem erstmaligen Anschließen eines Joysticks an den C-863 wird die Kalibration der einzelnen Joystick-Achsen empfohlen. Die Kalibration umfasst die folgenden Schritte:

- Wenn entsprechende Bedienelemente am Joystick-Gerät vorhanden sind: mechanisches Justieren der Joystick-Achsen.
- Kalibration der Joystick-Achsen in PIMikroMove®

INFORMATION

Bei der Joystick-Kalibration in PIMikroMove® wird der zu verwendende Lookup-Tabellentyp ausgewählt bzw. die Lookup-Tabelle mit individuellen Werten gefüllt. Dazu muss kein Versteller am C-863 angeschlossen sein.

In folgenden Fällen ist die Kalibration für eine ordnungsgemäße Joystick-Steuerung notwendig:

- Nach dem Aktivieren der Joystick-Steuerung bewegt sich der Versteller, obwohl Sie den Joystick nicht betätigen.
- Das Ansprechverhalten des Joysticks entspricht nicht Ihren Anforderungen.
- Sie verwenden die Z-Achse eines Joysticks C-819.30.

INFORMATION

Die Anzahl der Schreibzyklen im permanenten Speicher ist durch die begrenzte Lebensdauer des Speicherchips (EEPROM) beschränkt.

- Kalibrieren Sie die Joystick-Achsen in PIMikroMove® nur, wenn es notwendig ist.
- Wenden Sie sich an unseren Kundendienst (S. 309), wenn der C-863 unerwartetes Verhalten zeigt.

INFORMATION

Für die Z-Achse des Joysticks C-819.30 ist keine mechanische Justierung möglich, und sie kann nicht mit den Standard-Lookup-Tabellentypen (linear oder parabolisch) des C-863 betrieben werden.

- Kalibrieren Sie die Z-Achse des Joysticks nach dem Anschließen an den C-863 mit PIMikroMove®.
- Verwenden Sie zur Kalibration der Z-Achse die Methode ***Measure Joystick Parameters and Use Custom Lookup Table***.
- Wiederholen Sie die Kalibration der Z-Achse, wenn Sie die Z-Achse an einen anderen Controller anschließen.

INFORMATION

Der parabolische Lookup-Tabellentyp ermöglicht mehr Feinfühligkeit bei langsamer Fahrt.

Joystick-Achse mechanisch justieren

- Prüfen Sie bei aktivierter Joystick-Steuerung, ob sich der Versteller bereits bewegt, wenn Sie den Joystick nicht betätigen.

Wenn ja:

- Prüfen Sie, ob der Joystick mechanisch arretiert ist, und lösen Sie gegebenenfalls die Arretierung.
- Belassen Sie die betroffene Joystick-Achse (d. h. z. B. den Steuerhebel) in Mittelstellung und justieren Sie sie mit den entsprechenden Bedienelementen, bis sich der Versteller nicht mehr bewegt. Bei den Joysticks C-819.20 und C-819.30 drehen Sie dazu den entsprechenden Drehknopf für die Justierung (S. 129).

Wenn nein:

- Prüfen Sie, ob das Ansprechverhalten des Joysticks Ihren Anforderungen entspricht.

Wenn ja:

- Die Kalibration ist beendet.

Wenn nein:

- Kalibrieren Sie die Joystick-Achse in PIMikroMove®.

Joystick-Achse in PIMikroMove® kalibrieren

1. Öffnen Sie im Hauptfenster von PIMikroMove® das Fenster **Joystick Calibration** über den Menüeintrag **C-863 > Calibrate controller joystick... > Joystick n**, wobei *n* das angeschlossene Joystick-Gerät bezeichnet.

n = 1: Joystick-Gerät, das an Pin 4 der Buchse **Joystick** angeschlossen werden kann (z.B. Joystick C-819.20 oder C-819.30 von PI).

n = 2: Joystick-Gerät, das an Pin 2 der Buchse **Joystick** angeschlossen werden kann.

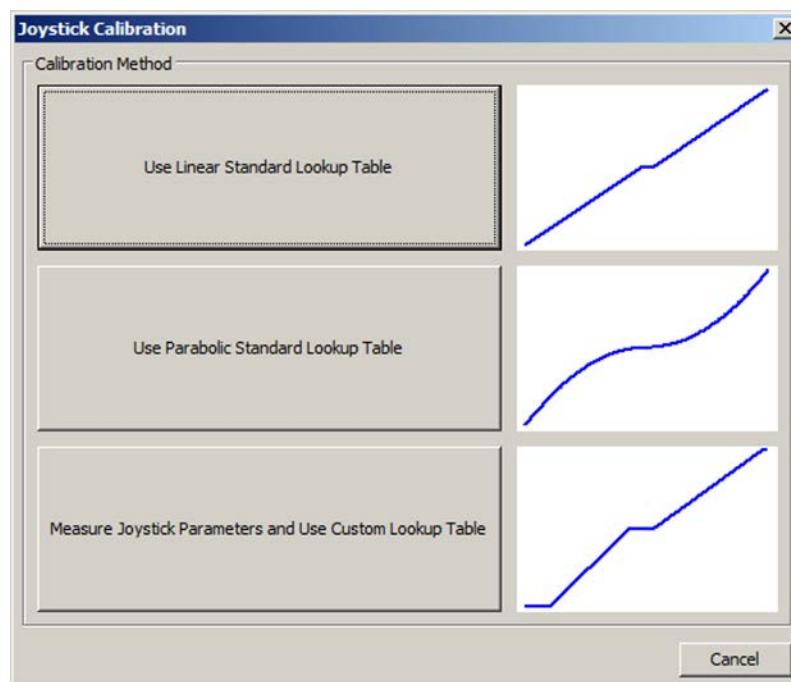


Abbildung 29: Fenster Joystick Calibration für die Kalibration einer Joystick-Achse

Das Fenster **Joystick Calibration** ermöglicht die Kalibration der unterstützten Joystick-Achsen des angeschlossenen Joystick-Geräts – mit C-863 jeweils eine Joystick-Achse.

2. Wählen Sie die bevorzugte Kalibrationsmethode aus, indem Sie auf die entsprechende Taste klicken:
 - Wenn Sie den linearen Lookup-Tabellentyp für die Joystick-Achse verwenden wollen, klicken Sie auf **Use Linear Standard Lookup Table**. Damit wird der entsprechende Lookup-Tabellentyp geladen, und die Kalibration ist beendet.

- Wenn Sie den parabolischen Lookup-Tabellentyp für die Joystick-Achse verwenden wollen, klicken Sie auf **Use Parabolic Standard Lookup Table**. Damit wird der entsprechende Lookup-Tabellentyp geladen, und die Kalibration ist beendet.
 - Wenn Sie die Z-Achse eines Joysticks C-819.30 angeschlossen haben oder generell das Verhalten des Joysticks in einer individuellen Lookup-Tabelle abbilden wollen, klicken Sie auf **Measure Joystick Parameters and Use Custom Lookup Table**. Damit öffnet sich das Fenster **Controller Joystick Calibration**.
3. Wenn sich das Fenster **Controller Joystick Calibration** geöffnet hat, folgen Sie den Anweisungen in diesem Fenster.

Auf diese Weise werden individuelle Lookup-Tabellenwerte ermittelt.

Durch Klicken auf **OK** laden Sie die ermittelten Lookup-Tabellenwerte in den permanenten Speicher des C-863. Die Kalibration ist damit beendet.

8.6.5 Verfügbare Joystick-Geräte

PI bietet die nachfolgend beschriebenen Joystick-Geräte als optionales Zubehör (S. 18) an.

Analoger Joystick C-819.20, 2 Achsen

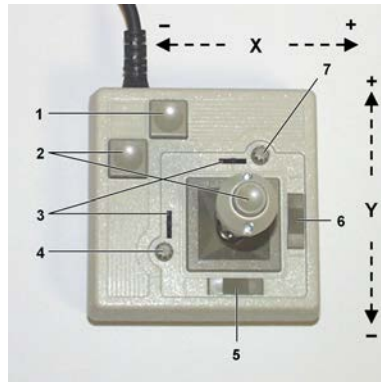


Abbildung 30: Joystick C-819.20

- 1 Drucktaste für die X-Achse
- 2 Drucktaste für die Y-Achse
- 3 Justageanzeiger
- 4 Drehknopf für Justierung der Y-Achse (Kalibrierung)
- 5 Arretierung der X-Achse
- 6 Arretierung der Y-Achse
- 7 Drehknopf für Justierung der X-Achse (Kalibrierung)

Analoger Joystick C-819.30, 3 Achsen

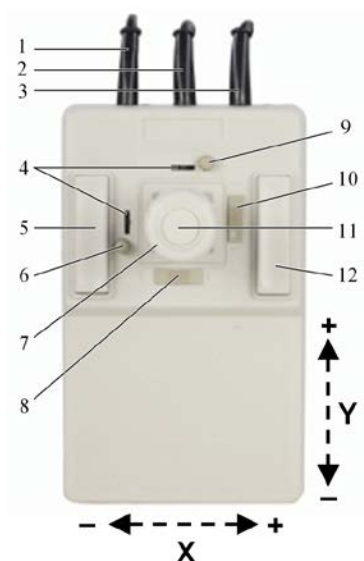


Abbildung 31: Joystick C-819.30

- 1 Kabel für die Z-Achse
- 2 Kabel für die Y-Achse
- 3 Kabel für die X-Achse
- 4 Justageanzeiger
- 5 Drucktaste für die Y-Achse
- 6 Drehknopf für Justierung der Y-Achse (Kalibrierung)
- 7 XY-Steuerhebel mit Drehknopf für Z-Achse
- 8 Arretierung der X-Achse
- 9 Drehknopf für Justierung der X-Achse (Kalibrierung)
- 10 Arretierung der Y-Achse
- 11 Drucktaste für die Z-Achse
- 12 Drucktaste für die X-Achse



Abbildung 32: Joystick C-819.30, Drehknopf für die Z-Achse

8.7 Controllermakros

8.7.1 Übersicht: Makrofunktionalitäten und Beispielmakros

Der C-863 kann Befehlsfolgen als Makros speichern und abarbeiten.

Die folgenden Funktionalitäten machen Makros zu einem wichtigen Werkzeug in vielen Anwendungsgebieten:

- Mehrere Makros können gleichzeitig gespeichert werden.
- Ein beliebiges Makro kann als Startup-Makro festgelegt werden. Das Startup-Makro wird bei jedem Einschalten oder Neustart des C-863 ausgeführt.
- Die Abarbeitung des Makros und das Stoppen der Makroausführung lassen sich an Bedingungen knüpfen. So können auch Schleifen realisiert werden.
- Makros können sich selbst oder andere Makros in mehreren Verschachtelungsebenen aufrufen.
- Variablen (S. 160) können für das Makro und im Makro selbst gesetzt und in verschiedenen Operationen verwendet werden.
- Eingangssignale können für Bedingungen und Variablen ausgewertet werden.

In diesem Handbuch finden Sie Beispielmakros für folgende Aufgaben:

- Achse hin und her bewegen (S. 136)
- Makro für Controller aufzeichnen, dessen Adresse verschieden von 1 ist (S. 137)
- Achse mit variablem Verfahrensweg hin und her bewegen (S. 139)
- Mehrfachaufruf eines Makros durch Schleife realisieren (S. 141)
- Achse durch Startup-Makro für den geregelten Betrieb vorbereiten (S. 142)
- Synchronisation zweier Controller (S. 144)
- Bewegung per Tastendruck stoppen (S. 145)
- Joystick-Steuerung mit Speicherung von Positionen (S. 147)
- Joystick-Steuerung mit Änderung der Geschwindigkeit (S. 152)

8.7.2 Befehle und Parameter für Makros

Befehle

Folgende Befehle stehen speziell für die Handhabung von Makros oder für die Verwendung in Makros zur Verfügung:

Befehl	Syntax	Funktion
ADD (S. 171)	ADD <Variable> <FLOAT1> <FLOAT2>	Addiert zwei Werte und speichert das Ergebnis als Variable (S. 160). Für lokale Variablen nur innerhalb von Makros verwendbar.
CPY (S. 175)	CPY <Variable> <CMD?>	Kopiert eine Antwort auf einen Befehl in eine Variable (S. 160). Für lokale Variablen nur innerhalb von Makros verwendbar.
DEL (S. 184)	DEL <uint>	Kann nur in Makros verwendet werden. Verzögert um <uint> Millisekunden.
JRC (S. 216)	JRC <Jump> <CMD?> <OP> <Value>	Kann nur in Makros verwendet werden. Löst einen relativen Sprung des Makroausführungszeigers in Abhängigkeit von einer Bedingung aus.
MAC (S. 219)	MAC BEG <macroname>	Startet die Aufzeichnung eines Makros mit dem Namen <i>macroname</i> auf dem Controller. <i>macroname</i> kann aus bis zu 8 Zeichen bestehen.
	MAC DEF <macroname>	Legt das angegebene Makro als Startup-Makro fest.
	MAC DEF?	Fragt das Startup-Makro ab.
	MAC DEL <macroname>	Löscht das angegebene Makro.
	MAC END	Stoppt die Makroaufzeichnung.
	MAC ERR?	Meldet den letzten Fehler, der während der Ausführung eines Makros auftrat.
	MAC NSTART <macroname> <uint> [<String1> [<String2>]]	Startet das angegebene Makro n-mal hintereinander (n = Anzahl der Ausführungen). Mit <String1> und <String2> können die Werte lokaler Variablen für das Makro gesetzt werden.
	MAC START <macroname> [<String1> [<String2>]]	Startet eine Ausführung des angegebenen Makros. Mit <String1> und <String2> können die Werte lokaler Variablen für das Makro gesetzt werden.

Befehl	Syntax	Funktion
MAC? (S. 222)	MAC? [<macroname>]	Listet alle Makros oder den Inhalt des angegebenen Makros auf.
MEX (S. 223)	MEX <CMD?> <OP> <Value>	Kann nur in Makros verwendet werden. Stoppt die Makroausführung in Abhängigkeit von einer Bedingung.
RMC? (S. 231)	RMC?	Listet die aktuell laufenden Makros auf.
VAR (S. 255)	VAR <Variable> <String>	Setzt eine Variable (S. 160) auf einen bestimmten Wert oder löscht sie. Für lokale Variablen nur innerhalb von Makros verwendbar.
VAR? (S. 256)	VAR? [{<Variable>}]	Gibt Variablenwerte zurück.
WAC (S. 259)	WAC <CMD?> <OP> <Value>	Kann nur in Makros verwendet werden. Wartet, bis eine Bedingung erfüllt ist.
#8 (S. 168)	-	Prüft, ob ein Makro auf dem Controller ausgeführt wird.

Parameter

Folgender Parameter steht für die Arbeit mit Makros zur Verfügung:

Parameter	Beschreibung und mögliche Werte
Ignore Macro Error? 0x72	Legt fest, ob das Controllermakro gestoppt wird, wenn bei dessen Ausführung ein Fehler auftritt. 0 = Bei Fehler Makro anhalten (Standard) 1 = Fehler ignorieren

8.7.3 Mit Makros arbeiten

Die Arbeit mit Makros umfasst Folgendes:

- Aufzeichnen von Makros (S. 134)
- Starten der Makroausführung (S. 138)
- Stoppen der Makroausführung (S. 141)
- Einrichten eines Startup-Makros (S. 142)
- Löschen von Makros (S. 143)

INFORMATION

Für die Arbeit mit Controllermakros wird die Verwendung der Registerkarte **Controller macros** in PIMikroMove® empfohlen. Dort können Sie Controllermakros komfortabel aufzeichnen, starten und verwalten. Details finden Sie im PIMikroMove® Handbuch.

Makro aufzeichnen**INFORMATION**

Der C-863 kann bis zu 32 Makros gleichzeitig speichern. Maximal 5 Verschachtelungsebenen sind in Makros möglich.

INFORMATION

Grundsätzlich können alle GCS-Befehle (S. 155) Bestandteil eines Makros werden. Ausnahmen:

- **RBT** für den Neustart des C-863
- **MAC BEG** und **MAC END** für die Makroaufzeichnung
- **MAC DEL** zum Löschen eines Makros

Abfragebefehle können in Makros in Kombination mit den Befehlen **CPY**, **JRC**, **MEX** und **WAC** verwendet werden. Andernfalls bleiben sie wirkungslos, da Makros keine Antworten an Schnittstellen senden.

INFORMATION

Wenn Sie ein Makro auf einem C-863 aufzeichnen, dessen Controlleradresse von 1 abweicht, beachten Sie Folgendes beim Eingeben der Befehle, die Bestandteil des Makros sein sollen:

- Wenn Sie mit PITerminal arbeiten und die Kommunikation mit der Schaltfläche **Connect...** hergestellt haben, muss die Empfängeradresse in jeder Befehlszeile eingetippt werden.
- Wenn Sie mit PIMikroMove® arbeiten oder mit PITerminal die Kommunikation mit der Schaltfläche **GCS DLL...** hergestellt haben, wird die Empfängeradresse automatisch mitgesendet und darf nicht eingetippt werden.

INFORMATION

Um die Anwendung von Makros flexibler zu gestalten, können Sie in Makros lokale und globale Variablen verwenden. Weitere Informationen siehe "Variablen" (S. 160).

INFORMATION

Die Anzahl der Schreibzyklen im permanenten Speicher ist durch die begrenzte Lebensdauer des Speicherchips beschränkt.

- Zeichnen Sie Makros nur auf, wenn es notwendig ist.
- Verwenden Sie Variablen (S. 160) in Makros, und geben Sie beim Starten der Makroausführung die entsprechenden Variablenwerte an.
- Wenden Sie sich an unseren Kundendienst (S. 309), wenn der C-863 unerwartetes Verhalten zeigt.

INFORMATION

Ein Makro wird überschrieben, wenn erneut ein Makro mit demselben Namen aufgezeichnet wird.

1. Starten Sie die Makroaufzeichnung.
 - Wenn Sie mit PITerminal oder im Fenster **Command entry** von PIMikroMove® arbeiten: Senden Sie den Befehl `MAC BEG macroname`, wobei *macroname* den Namen des Makros bezeichnet.
 - Wenn Sie in PIMikroMove® auf der Registerkarte **Controller macros** arbeiten: Klicken Sie auf das Symbol **Create new empty macro**, um eine Registerkarte zum Eingeben eines neuen Makros zu erzeugen. Geben Sie **nicht** den Befehl `MAC BEG macroname` ein.
2. Geben Sie unter Verwendung der normalen Befehlssyntax Zeile für Zeile die Befehle ein, die Bestandteil des Makros *macroname* sein sollen.

Makros können sich selbst oder andere Makros in mehreren Verschachtelungsebenen aufrufen.
3. Beenden Sie die Makroaufzeichnung.
 - Wenn Sie mit PITerminal oder im Fenster **Command entry** von PIMikroMove® arbeiten: Senden Sie den Befehl `MAC END`.

- Wenn Sie in PIMikroMove® auf der Registerkarte **Controller macros** arbeiten: Geben Sie **nicht** den Befehl `MAC END` ein. Klicken auf das Symbol **Send macro to controller** und geben Sie in einem separaten Dialogfenster den Namen des Makros ein.

Das Makro wurde im permanenten Speicher des C-863 abgelegt.

4. Wenn Sie prüfen wollen, ob das Makro korrekt aufgezeichnet wurde:

Wenn Sie mit PITerminal oder im Fenster **Command entry** von PIMikroMove® arbeiten:

- Fragen Sie ab, welche Makros im C-863 gespeichert sind, indem Sie den Befehl `MAC?` senden.
- Fragen Sie den Inhalt des Makros *macroname* ab, indem Sie den Befehl `MAC? macroname` senden.

Wenn Sie in PIMikroMove® auf der Registerkarte **Controller macros** arbeiten:

- Klicken Sie auf das Symbol **Read list of macros from controller**.
- Markieren Sie das zu prüfende Makro in der Liste auf der linken Seite, und klicken Sie auf das Symbol **Load selected macro from controller**.

Beispiel: Achse hin und her bewegen

INFORMATION

Bei der Aufzeichnung von Makros auf der Registerkarte **Controller macros** in PIMikroMove® müssen die Befehle `MAC BEG` und `MAC END` weggelassen werden.

Die Achse 1 soll sich hin und her bewegen. Dazu werden 3 Makros aufgezeichnet. Makro 1 startet die Bewegung in die positive Richtung und wartet, bis die Achse die Zielposition erreicht hat. Makro 2 erfüllt diese Aufgabe für die negative Bewegungsrichtung. Makro 3 ruft die Makros 1 und 2 auf.

- Zeichnen Sie die Makros auf, indem Sie senden:

```
MAC BEG macro1
```

```
MVR 1 12.5
```

```
WAC ONT? 1 = 1
```

```
MAC END
```



```

MAC BEG macro2
MVR 1 -12.5
WAC ONT? 1 = 1
MAC END
MAC BEG macro3
MAC START macro1
MAC START macro2
MAC END

```

Beispiel: Makro für Controller aufzeichnen, dessen Adresse verschieden von 1 ist

INFORMATION

Bei der Aufzeichnung von Makros auf der Registerkarte **Controller macros** in PIMikroMove® müssen die Befehle **MAC BEG** und **MAC END** weggelassen werden.

Über die DIP-Schalter ist die Controlleradresse auf 2 eingestellt. In diesem Beispiel erfolgt die Makroaufzeichnung mit PITerminal, wobei die Kommunikation mit der Taste **Connect...** hergestellt wurde (daher muss die Empfängeradresse in jeder Befehlszeile eingetippt werden).

Durch das Makro ref soll für Achse 1 der Servomodus eingeschaltet und eine Referenzfahrt zum Referenzschalter gestartet werden.

1. Zeichnen Sie das Makro auf, indem Sie senden:

```

2 MAC BEG ref
2 SV0 1 1
2 DEL 1000
2 FRF 1
2 MAC END

```

2. Prüfen Sie den Inhalt des Makros ref, indem Sie senden:

```

2 MAC? ref

```

Die Antwort lautet:

0 2 SVO 1 1

DEL 1000

FRF 1

Die erste Zeile der Antwort enthält die Empfänger- und Senderadresse gemäß der GCS-Syntax für mehrzeilige Antworten. Die Empfängeradresse ist jedoch nicht in das Makro aufgenommen worden.

Makroausführung starten

INFORMATION

Von der Befehlszeile können sämtliche Befehle gesendet werden, während auf dem Controller ein Makro läuft. Der Makroinhalt und Bewegungsbefehle, die von der Befehlszeile empfangen werden, können sich gegenseitig überschreiben.

INFORMATION

Zeitgleiche Ausführung mehrerer Makros ist nicht möglich. Es kann jeweils nur ein Makro ausgeführt werden.

INFORMATION

Sie können die Makroausführung mit den Befehlen **JRC** und **WAC** an Bedingungen knüpfen. Die Befehle müssen im Makro enthalten sein.

Im Folgenden wird PITerminal oder das Fenster **Command entry** von PIMikroMove® verwendet, um Befehle einzugeben. Details zur Arbeit mit der Registerkarte **Controller macros** in PIMikroMove® finden Sie im PIMikroMove® Handbuch.

1. Wenn die Makroausführung trotz Auftretens eines Fehlers fortgesetzt werden soll:
 - Stellen Sie den Parameter **Ignore Macro Error?** (ID 0x72) entsprechend ein: Senden Sie den Befehl **SPA 1 0x72 Status**, wobei *Status* die Werte 0 oder 1 annehmen kann (0 = Bei Fehler Makro anhalten (Standard); 1 = Makrofehler ignorieren).

Weitere Informationen zum Ändern von Parametern finden Sie in "Anpassen von Einstellungen" (S. 279).

2. Starten Sie die Makroausführung:

- Wenn das Makro einmal ausgeführt werden soll, senden Sie den Befehl `MAC START macroname string`, wobei *macroname* den Namen des Makros bezeichnet.
- Wenn das Makro n-mal ausgeführt werden soll, senden Sie den Befehl `MAC NSTART macroname n string`, wobei *macroname* den Namen des Makros bezeichnet und *n* die Anzahl der Ausführungen angibt.

string steht für die Werte lokaler Variablen. Die Werte sind nur dann anzugeben, wenn das Makro entsprechende lokale Variablen enthält. Die Reihenfolge der Werte bei der Eingabe muss der Nummerierung der zugehörigen lokalen Variablen entsprechen, beginnend mit dem Wert der lokalen Variablen 1. Die einzelnen Werte müssen durch Leerzeichen voneinander getrennt werden.

3. Wenn Sie die Makroausführung prüfen wollen:

- Fragen Sie ab, ob ein Makro auf dem Controller ausgeführt wird, indem Sie den Befehl `#8` senden.
- Fragen Sie den Namen des Makros ab, das gerade auf dem Controller ausgeführt wird, indem Sie den Befehl `RMC?` senden.

Beispiel: Achse mit variablem Verfahrensweg hin und her bewegen

INFORMATION

Bei der Aufzeichnung von Makros auf der Registerkarte **Controller macros** in PIMikroMove® müssen die Befehle `MAC BEG` und `MAC END` weggelassen werden.

Die Achse 1 soll sich hin und her bewegen. Der Verfahrensweg nach links und rechts soll variabel einstellbar sein, ohne dass dazu die verwendeten Makros geändert werden müssen. Deshalb werden lokale und globale Variablen verwendet.

1. Legen Sie die globalen Variablen LEFT und RIGHT an, indem Sie senden:

`VAR LEFT 5`

`VAR RIGHT 15`

LEFT hat damit den Wert 5, und RIGHT hat den Wert 15. Diese Werte können Sie jederzeit ändern, indem Sie z.B. den Befehl VAR erneut senden.

- Legen Sie die globalen Variablen nach jedem Einschalten oder Neustart des C-863 erneut an, da sie nur in den flüchtigen Speicher des C-863 geschrieben werden.

2. Zeichnen Sie das Makro MOVLR auf, indem Sie senden:

```
MAC BEG movlr
MAC START movwai ${LEFT}
MAC START movwai ${RIGHT}
MAC END
```

MOVLR startet das (noch aufzuzeichnende) Makro MOVWAI nacheinander für beide Bewegungsrichtungen. Die Werte der globalen Variablen LEFT und RIGHT werden beim Start von MOVWAI verwendet, um den Wert der in MOVWAI enthaltenen lokalen Variable 1 zu setzen (Dollarzeichen und geschweifte Klammern sind erforderlich, damit die lokale Variable 1 im Makro tatsächlich mit dem Wert der globalen Variable und nicht mit ihrem Namen ersetzt wird).

3. Zeichnen Sie das Makro MOVWAI auf, indem Sie senden:

```
MAC BEG movwai
MOV 1 $1
WAC ONT? 1 = 1
MAC END
```

MOVWAI bewegt die Achse 1 zur Zielposition, die durch den Wert der lokalen Variablen 1 vorgegeben ist, und wartet, bis die Achse die Zielposition erreicht hat.

4. Starten Sie die Ausführung des Makros MOVLR, indem Sie senden:

```
MAC NSTART movlr 5
```

Das Makro MOVLR wird fünfmal hintereinander ausgeführt, d.h. die Achse 1 bewegt sich fünfmal im Wechsel zu den Positionen 5 und 15. Für die Anzahl der Ausführungen können Sie auch einen beliebigen anderen Wert wählen.

Beispiel: Mehrfachaufruf eines Makros durch Schleife realisieren

INFORMATION

Bei der Aufzeichnung von Makros auf der Registerkarte **Controller macros** in PIMikroMove® müssen die Befehle `MAC BEG` und `MAC END` weggelassen werden.

Das Makro TESTDION prüft den Zustand der digitalen Eingangsleitungen auf der Buchse I/O. Es verwendet eine lokale Variable, um die digitale Eingangsleitung zu bezeichnen (1 bis 4). Damit das Makro TESTDION nicht für jede Eingangsleitung separat aufgerufen werden muss, wird ein weiteres Makro mit einer Schleife aufgezeichnet.

- Zeichnen Sie das Makro LOOPDION auf, indem Sie senden:

```
MAC BEG loopdion
```

```
VAR COUNTER 1
```

```
MAC START TESTDION ${COUNTER}
```

```
ADD COUNTER ${COUNTER} 1
```

```
JRC -2 VAR? COUNTER < 5
```

```
MAC END
```

Die Variable COUNTER wird mit dem Wert 1 angelegt. Anschließend wird das Makro TESTDION für die Eingangsleitung gestartet, deren Kennung durch die Variable COUNTER vorgegeben ist. Danach wird der Wert von COUNTER um 1 hochgesetzt. Solange der Wert von COUNTER kleiner als 5 ist, springt der Makroausführungszeiger anschließend 2 Zeilen zurück, so dass TESTDION nun für die nächste digitale Eingangsleitung gestartet wird.

Makroausführung stoppen

INFORMATION

Sie können das Stoppen der Makroausführung mit dem Befehl `MEX` an eine Bedingung knüpfen. Der Befehl muss im Makro enthalten sein.

Im Folgenden wird PITerminal oder das Fenster **Command entry** von PIMikroMove® verwendet, um Befehle einzugeben. Details zur Arbeit mit der Registerkarte **Controller macros** in PIMikroMove® finden Sie im PIMikroMove®-Handbuch.

- Stoppen Sie die Makroausführung mit den Befehlen `#24` oder `STP`.
- Wenn Sie prüfen wollen, ob während der Makroausführung ein Fehler aufgetreten ist, senden Sie den Befehl `MAC ERR?`. Die Antwort zeigt den letzten Fehler an, der aufgetreten ist.

Startup-Makro einrichten

Ein beliebiges Makro kann als Startup-Makro festgelegt werden. Das Startup-Makro wird bei jedem Einschalten oder Neustart des C-863 ausgeführt.

INFORMATION

Das Löschen eines Makros löscht nicht seine Auswahl als Startup-Makro.

Im Folgenden wird PITerminal oder das Fenster **Command entry** von PIMikroMove® verwendet, um Befehle einzugeben. Details zur Arbeit mit der Registerkarte **Controller macros** in PIMikroMove® finden Sie im PIMikroMove® Handbuch.

- Legen Sie mit dem Befehl `MAC DEF macroname` ein Makro als Startup-Makro fest, wobei *macroname* den Namen des Makros bezeichnet.
- Wenn Sie die Auswahl des Startup-Makros aufheben und kein anderes Makro als Startup-Makro festlegen wollen, senden Sie nur `MAC DEF`.
- Fragen Sie den Namen des aktuell festgelegten Startup-Makros ab, indem Sie den Befehl `MAC DEF?` senden.

Beispiel: Achse durch Startup-Makro für geregelten Betrieb vorbereiten

INFORMATION

Bei der Aufzeichnung von Makros auf der Registerkarte **Controller macros** in PIMikroMove® müssen die Befehle `MAC BEG` und `MAC END` weggelassen werden.

Das Makro STARTCL schaltet für Achse 1 die Joysticksteuerung aus und den Servomodus ein und startet eine Referenzfahrt zum negativen Endschalter. Indem STARTCL als Startup-Makro festgelegt wird, ist die Achse 1 sofort nach dem Einschalten für den geregelten Betrieb bereit.

➤ Senden Sie:

```
MAC BEG startcl
```

```
JON 1 0
```

```
SVO 1 1
```

```
DEL 1000
```

```
FNL 1
```

```
MAC END
```

```
MAC DEF startcl
```

INFORMATION

Bei Verwendung dieses Makros sollten die Parametereinstellungen des C-863 im permanenten Speicher an den angeschlossenen Verstärker angepasst sein. Alternativ können die Parametereinstellungen im flüchtigen Speicher auch durch das Startup-Makro gesetzt werden. Weitere Informationen siehe "Anpassen von Einstellungen" (S. 279).

Makro löschen

INFORMATION

Ein laufendes Makro kann nicht gelöscht werden.

Im Folgenden wird PITerminal oder das Fenster **Command entry** von PIMikroMove® verwendet, um Befehle einzugeben. Details zur Arbeit mit der Registerkarte **Controller macros** in PIMikroMove® finden Sie im PIMikroMove® Handbuch.

➤ Löschen Sie ein Makro mit dem Befehl `MAC DEL macroname`, wobei *macroname* den Namen des Makros bezeichnet.

8.7.4 Makrobeispiel: Synchronisation zweier Controller

INFORMATION

Bei der Aufzeichnung von Makros auf der Registerkarte **Controller macros** in PIMikroMove® müssen die Befehle **MAC BEG** und **MAC END** weggelassen werden.

Aktion	Befehl	Ergebnis
Digitale Ausgangsleitung 1 auf der Buchse I/O des Master-Controllers mit der digitalen Eingangsleitung 1 auf der Buchse I/O des Slave-Controllers verbinden.	- Geeignetes Kabel verwenden. Pinbelegung siehe "I/O" (S. 316).	Das digitale Ausgangssignal des Master-Controllers kann als Trigger für die Bewegung der Achse verwendet werden, die an den Slave-Controller angeschlossen ist.
Auf dem Master-Controller und auf dem Slave-Controller die Bewegung vorbereiten.	SV0 1 1 FRF 1 1 VEL 1 0 MOV 1 5.5	Für beide Controller: Der Servomodus ist eingeschaltet und die Achse hat eine Referenzfahrt ausgeführt – hier zum Referenzschalter. Die Geschwindigkeit ist auf null gesetzt. Somit bewegt sich die Achse vorerst nicht, obwohl das Bewegungskommando für die Fahrt zur absoluten Position 5,5 schon gesendet wurde.
Makro MASTER auf dem Master-Controller aufzeichnen.	MAC BEG master DIO 1 1 VEL 1 100 MAC END	Das Makro hat folgende Aufgaben: - Digitale Ausgangsleitung 1 des Mastercontrollers in den Zustand high schalten, um den Slave-Controller zu triggern - Geschwindigkeit auf 100 setzen, um die Bewegung zu starten
Makro SLAVE auf dem Slave-Controller aufzeichnen.	MAC BEG slave WAC DIO? 1 = 1 VEL 1 100 MAC END	Das Makro hat folgende Aufgaben: - Bedingung setzen: Das Makro wird erst dann weiter ausgeführt, wenn die digitale Eingangsleitung 1 den Zustand high hat (d.h. wenn der Mastercontroller das Triggersignal ausgibt). - Geschwindigkeit auf 100 setzen, um die Bewegung zu starten

Aktion	Befehl	Ergebnis
Makro SLAVE auf dem Slave-Controller starten.	MAC START slave	Die Achse am Slave-Controller bewegt sich noch nicht, da die Bedingung zur weiteren Makroausführung noch nicht erfüllt ist.
Makro MASTER auf dem Master-Controller starten.	MAC START master	Beide Achsen bewegen sich, da ihre Geschwindigkeit jetzt jeweils von Null verschieden ist. Die Bewegung erfolgt synchron.

8.7.5 Makrobeispiel: Bewegung per Tastendruck stoppen

INFORMATION

Um die digitalen Eingangssignale für die Verwendung in Makros zu erzeugen, können Sie die Pushbutton-Box C-170.PB von PI an die Buchse **I/O** anschließen. Sie zeigt über LEDs auch den Status der digitalen Ausgangsleitungen an.

INFORMATION

Bei der Aufzeichnung von Makros auf der Registerkarte **Controller macros** in PIMikroMove® müssen die Befehle **MAC BEG** und **MAC END** weggelassen werden.

Aktion	Befehl	Ergebnis
Digitale Eingangsleitung 1 auf der Buchse I/O mit einer geeigneten Signalquelle verbinden.	- Pinbelegung siehe "I/O" (S. 316).	Das digitale Eingangssignal kann z.B. für einen bedingten Sprung des Makroausführungs-Zeigers verwendet werden.

Aktion	Befehl	Ergebnis
Makro HALT auf dem Controller aufzeichnen.	<pre>MAC BEG halt MVR 1 5 JRC 2 DIO? 1 = 1 JRC -1 ONT? 1 = 0 HLT 1 MAC END</pre>	Das Makro hat folgende Aufgaben: ▪ Relative Bewegung der Achse 1 starten ▪ Bedingung setzen: Wenn die digitale Eingangsleitung 1 den Zustand high hat (bei Verwendung der Pushbutton-Box: Taste 1 ist gedrückt), springt der Makroausführungs-Zeiger zwei Zeilen nach vorn. Damit wird die Achse angehalten. Andernfalls wird die Makroausführung mit der nächsten Zeile fortgesetzt. ▪ Bedingung setzen: Solange die Achse 1 die Zielposition noch nicht erreicht hat, springt der Makroausführungs-Zeiger eine Zeile zurück. Damit wird eine Schleife eingerichtet.
Makro HALT auf dem Controller starten.	<pre>MAC START halt</pre>	Die Bewegung der Achse 1 startet. Sie wird durch Schalten der digitalen Eingangsleitung 1 in den Zustand high (z.B. durch Tastendruck) angehalten. Unabhängig davon, ob die Achse die Zielposition erreicht hat oder vorher angehalten wurde, wird der Fehlercode durch den Befehl <code>HLT</code> auf 10 gesetzt.
Wenn der Fehlercode 10 stört: alternatives Makro HALTVAR aufzeichnen, das eine Variable verwendet. Details siehe "Variablen" (S. 160).	<pre>MAC BEG haltvar MVR 1 5 JRC 2 DIO? 1 = 1 JRC -1 ONT? 1 = 0 CPY TARGET POS? 1 MOV 1 \${TARGET} VAR TARGET MAC END</pre>	Das Makro hat die selben Aufgaben wie das Makro HALT. Jedoch wird Achse 1 bei Tastendruck nicht durch den Befehl <code>HLT</code> angehalten, sondern das Ergebnis der Abfrage <code>POS? 1</code> wird in die Variable <code>TARGET</code> kopiert. Diese Variable wird dann als Zielposition für den Befehl <code>MOV</code> verwendet. Somit bleibt die Achse, wo sie gerade war. Zur Bereinigung wird <code>TARGET</code> mit dem Befehl <code>VAR</code> als leer definiert, wodurch die Variable gelöscht wird.

Aktion	Befehl	Ergebnis
Makro HALTVAR auf dem Controller starten.	MAC START haltvar	Die Bewegung der Achse 1 startet. Sie wird durch Schalten der digitalen Eingangsleitung 1 in den Zustand high (z.B. durch Tastendruck) angehalten. Fehlercode 10 wird nicht gesetzt, weil kein Halte- oder Stoppbefehl verwendet wird.

8.7.6 Makrobeispiel: Joysticksteuerung mit Speicherung von Positionen

Aufgabe:

Achse 1 soll mit einem Joystick gesteuert werden. Die Joysticksteuerung soll nur aktiv sein, wenn gleichzeitig der Joystick-Button gedrückt ist. Durch die Tasten einer angeschlossenen Pushbutton-Box sollen außerdem bis zu vier Positionen im Controller gespeichert bzw. von der Achse angefahren werden können. Die LEDs der Pushbutton-Box sollen anzeigen, ob der Controller zum Speichern der aktuellen Position bereit ist und ob die Speicherung erfolgt ist.

Lösungsansatz:

Die Makros STARTUP, MAINLOOP, TESTJOYB, TESTDION und MVAX2ST werden auf dem Controller aufgezeichnet. Sie verwenden die globalen Variablen STORE1, STORE2, STORE3, STORE4, COUNTER sowie die lokalen Variablen 1 und 2.

INFORMATION

Bei der Aufzeichnung von Makros auf der Registerkarte **Controller macros** in PIMikroMove® müssen die Befehle MAC BEG und MAC END weggelassen werden.

Aktion	Befehl	Ergebnis
Pushbutton-Box C-170.PB von PI an der Buchse I/O anschließen.	-	Die digitalen Eingangsleitungen 1 bis 4 sind in den Zustand high geschaltet, solange die entsprechende Taste gedrückt ist. Die Zustände der digitalen Ausgangsleitungen 1 bis 4 werden durch die LEDs dargestellt, die in den Tasten integriert sind.

Aktion	Befehl	Ergebnis
Joystick C-819.20 oder C-819.30 an der Buchse Joystick anschließen.	-	Für Befehle ist die angeschlossene Joystick-Achse zugänglich als Achse 1 von Joystick-Gerät 1, und die Joystick-Taste ist zugänglich als Taste 1 von Joystick-Gerät 1.
Servomodus für Achse 1 einschalten.	SV0 1 1	Der Servomodus muss eingeschaltet sein, damit Achse 1 über einen Joystick gesteuert werden kann.
Referenzfahrt für Achse 1 starten.	FRF 1	Die Achse startet eine Referenzfahrt – hier zum Referenzschalter. Danach können absolute Achsenpositionen kommandiert werden.
Bestimmen, welche Achse über die Joystick-Achse gesteuert werden soll.	JAX 1 1 1	Achse 1 wird der Joystick-Achse 1 des Joystickgeräts 1 zugewiesen. Die Joysticksteuerung ist noch nicht aktiv.
Makro STARTUP auf dem Controller aufzeichnen.	MAC BEG startup CPY STORE1 POS? 1 CPY STORE2 POS? 1 CPY STORE3 POS? 1 CPY STORE4 POS? 1 MAC START MAINLOOP MAC END	Das Makro hat folgende Aufgaben: ▪ Variablen zur Speicherung der Position initialisieren ▪ Makro MAINLOOP für die Hauptschleife starten
Makro MAINLOOP auf dem Controller aufzeichnen.	MAC BEG mainloop MAC START TESTJOYB VAR COUNTER 1 MAC START TESTDION \${COUNTER} ADD COUNTER \${COUNTER} 1 JRC -2 VAR? COUNTER < 5 MAC START MAINLOOP MAC END	Das Makro hat folgende Aufgaben: ▪ Makro TESTJOYB für Joysticksteuerung starten ▪ Makro TESTDION nacheinander für alle digitalen Eingänge (d.h. alle Tasten der Pushbutton-Box) starten, unter Verwendung einer Schleife ▪ Sich selbst aufrufen, um die Hauptschleife zu bilden

Aktion	Befehl	Ergebnis
Makro TESTJOYB auf dem Controller aufzeichnen.	<pre>MAC BEG testjoyb MEX JBS? 1 1 = 0 JON 1 1 DIO 0 15 JRC 6 JBS? 1 1 = 0 DEL 50 DIO 0 0 JRC 3 JBS? 1 1 = 0 DEL 50 JRC -6 JBS? 1 1 = 1 JON 1 0 DIO 0 0 MAC END</pre>	Das Makro hat folgende Aufgaben: ▪ Makroausführung stoppen, wenn Joystick-Taste 1 nicht gedrückt ist ▪ Joystick-Gerät 1 aktivieren ▪ Alle LEDs an der Pushbutton-Box einschalten ▪ 6 Zeilen nach vorn springen (zu JON 1 0), wenn Joystick-Taste 1 nicht mehr gedrückt ist ▪ 50 ms warten ▪ Alle LEDs an der Pushbutton-Box ausschalten ▪ 3 Zeilen nach vorn springen (zu JON 1 0), wenn Joystick-Taste 1 nicht mehr gedrückt ist ▪ Weitere 50 ms warten ▪ 6 Zeilen zurück springen (zu DIO 0 15), wenn Joystick-Taste 1 immer noch gedrückt ist ▪ Joystick-Gerät 1 deaktivieren ▪ Alle LEDs an der Pushbutton-Box ausschalten
Makro TESTDION auf dem Controller aufzeichnen.	<pre>MAC BEG testdion MEX VAR? 0 != 1</pre>	Das Makro hat folgende Aufgaben: ▪ Makroausführung stoppen, wenn die Anzahl der beim Start von TESTDION angegebenen lokalen Variablen nicht 1 ist

Aktion	Befehl	Ergebnis
	MEX DIO? \$1 = 0 DEL 300 JRC 3 DIO? \$1 = 1 MAC START MVAX2ST \$1 MEX DIO? \$1 = 0 DEL 400 MEX DIO? \$1 = 0 DIO \$1 1 WAC DIO? \$1 = 0 DIO \$1 0	▪ Makroausführung stoppen, wenn die durch die lokale Variable 1 angegebene Taste der Pushbutton-Box nicht mehr gedrückt ist (entsprechende Eingangsleitung hat den Zustand low) ▪ 300 ms warten ▪ Wenn die Taste immer noch gedrückt ist, 3 Zeilen nach vorn springen (zu DEL 400) ▪ Makro MVAX2ST starten, da die Taste nur kurz gedrückt wurde. Der Wert der lokalen Variablen 1 wird auch für die lokale Variable 1 in MVAX2ST verwendet. MVAX2ST bewegt Achse 1 zur Position, die für die Taste hinterlegt ist. ▪ Makroausführung stoppen, wenn die Taste nicht mehr gedrückt ist ▪ 400 ms warten ▪ Makroausführung stoppen, wenn die Taste nicht mehr gedrückt ist ▪ Die zur gedrückten Taste gehörende LED der Pushbutton-Box einschalten, um die Speicherung der aktuellen Position anzuzeigen ▪ Das Makro wird erst dann weiter ausgeführt, wenn die Taste nicht mehr gedrückt ist ▪ LED ausschalten

Aktion	Befehl	Ergebnis
	CPY STORE\$1 POS? 1 MAC END	Aktuelle Position der Achse 1 in der durch die lokale Variable 1 bezeichneten globalen Variable speichern
Makro MVAX2ST auf dem Controller aufzeichnen.	MAC BEG MVAX2ST CPY 2 VAR? STORE\$1 MOV 1 \$2 MAC END	Das Makro hat folgende Aufgaben: - Fragt die durch die lokale Variable 1 bezeichnete Speichervariable ab und kopiert ihren Wert in die lokale Variable 2 - Bewegung der Achse 1 zur Zielposition starten, die durch die lokale Variable 2 vorgegeben ist
Makro STARTUP auf dem Controller starten. Alternativ: Wenn die Variablen zur Speicherung von Positionen nicht initialisiert werden sollen, stattdessen das Makro MAINLOOP auf dem Controller starten.	MAC START startup	Die Joystick-Steuerung wird durch Drücken der Joystick-Taste aktiviert. Bei aktivierter Joystick-Steuerung blinken die LEDs der Pushbutton-Box schnell und zeigen damit an, dass die Tasten der Box nicht betätigt werden sollen. Nach dem Loslassen der Joystick-Taste ist die Joystick-Steuerung deaktiviert und die LEDs schalten sich ab. Die Pushbutton-Box kann jetzt zum Anfahren von gespeicherten Positionen oder zur Speicherung der aktuellen Position verwendet werden. Um den Versteller zu einer gespeicherten Position zu bewegen, wird die entsprechende Taste der Pushbutton-Box kurz gedrückt. Um die aktuelle Position des Verstellers zu speichern, wird eine Taste an der Pushbutton-Box so lange gedrückt, bis die LED der Taste aufleuchtet.

8.7.7 Makrobeispiel: Joysticksteuerung mit Änderung der Geschwindigkeit

INFORMATION

Bei der Aufzeichnung von Makros auf der Registerkarte **Controller macros** in PIMikroMove® müssen die Befehle **MAC BEG** und **MAC END** weggelassen werden.

Aktion	Befehl	Ergebnis
Joystick C-819.20 oder C-819.30 an der Buchse Joystick anschließen.	-	Für Befehle ist die angeschlossene Joystick-Achse zugänglich als Achse 1 von Joystick-Gerät 1, und die Joystick-Taste ist zugänglich als Taste 1 von Joystick-Gerät 1.
Servomodus für Achse 1 einschalten.	SV0 1 1	Der Servomodus muss eingeschaltet sein, damit Achse 1 über einen Joystick gesteuert werden kann.
Referenzfahrt für Achse 1 starten.	FRF 1	Die Achse startet eine Referenzfahrt – hier zum Referenzschalter. Danach können absolute Achsenpositionen kommandiert werden.
Bestimmen, welche Achse über die Joystick-Achse gesteuert werden soll.	JAX 1 1 1	Achse 1 wird der Joystick-Achse 1 des Joystickgeräts 1 zugewiesen. Die Joysticksteuerung ist noch nicht aktiv.
Makro JOYVEL auf dem Controller aufzeichnen.	MAC BEG joyvel JON 1 1 JRC 3 JBS? 1 1 = 1 VEL 1 0.5 JRC -2 JBS? 1 1 = 0 VEL 1 1	Das Makro hat folgende Aufgaben: ▪ Joystick-Gerät 1 aktivieren ▪ Wenn die Joystick-Taste 1 gedrückt ist, 3 Zeilen nach vorn springen (zu VEL 1 1). ▪ Die Maximalgeschwindigkeit während der Joystick-Steuerung beträgt 0,5. ▪ Wenn die Joystick-Taste 1 nicht gedrückt ist, 2 Zeilen zurück springen, um eine Schleife zu bilden. ▪ Die Maximalgeschwindigkeit während der Joystick-Steuerung beträgt 1.

Aktion	Befehl	Ergebnis
	JRC -4 JON? 1 = 1 MAC END	Wenn Joystick-Gerät 1 noch aktiv ist, 4 Zeilen zurück springen, um eine Schleife zu bilden.
Makro JOYVEL auf dem Controller starten.	MAC START joyvel	Langsame Fahrt: Bewegen Sie den Steuerhebel des Joysticks. Schnelle Fahrt: Halten Sie die Drucktaste 1 des Joysticks gedrückt und bewegen Sie den Steuerhebel.

9 GCS-Befehle

In diesem Kapitel

Schreibweise	155
GCS-Syntax für Syntaxversion 2.0.....	156
Empfänger- und Senderadresse	158
Variablen.....	160
Befehlsübersicht	162
Befehlsbeschreibungen für GCS 2.0	166
Fehlercodes	262

9.1 Schreibweise

Für die Festlegung der GCS-Syntax und die Beschreibung der Befehle wird folgende Schreibweise verwendet:

<...>	Spitze Klammern kennzeichnen ein Befehlsargument, das die Kennung eines Elements oder ein befehlspezifischer Parameter sein kann.
[...]	Eckige Klammern kennzeichnen eine optionale Angabe.
{...}	Geschweifte Klammern kennzeichnen die Wiederholung von Angaben, d. h. es kann auf mehr als ein Element (z. B. mehrere Achsen) in einer Befehlszeile zugegriffen werden.
LF	LineFeed (ASCII-Zeichen 10) ist das Standard-Abschlusszeichen (Zeichen am Ende einer Befehlszeile).
SP	Space (ASCII-Zeichen 32) steht für ein Leerzeichen.
"..."	Anführungszeichen zeigen an, dass die von ihnen eingeschlossenen Zeichen ausgegeben werden oder einzugeben sind.

9.2 GCS-Syntax für Syntaxversion 2.0

Ein GCS-Befehl besteht aus 3 Buchstaben, z. B. CMD. Dem dazugehörigen Abfragebefehl wird am Ende ein Fragezeichen hinzugefügt, z. B. CMD?.

Befehlskürzel:

CMD ::= Buchstabe1 Buchstabe2 Buchstabe3 [?]

Ausnahmen:

- Einzeichenbefehle, wie z. B. Befehle für schnelles Abfragen, bestehen aus nur einem ASCII-Zeichen. Geschrieben wird das ASCII-Zeichen als Kombination aus # und dem Code des Zeichens in Dezimalschreibweise, z. B. als #24.
- *IDN? (für GPIB-Kompatibilität).

Beim Befehlskürzel wird nicht zwischen Groß- und Kleinschreibung unterschieden. Das Befehlskürzel und alle Argumente (z. B. Achsen- und Kanalkennungen, Parameter etc.) müssen mit einem Leerzeichen (SP) voneinander getrennt werden. Die Befehlszeile endet mit dem Abschlusszeichen (LF).

CMD[{{SP}}<Argument>]]LF

CMD?{{SP}}<Argument>]]LF

Ausnahme:

- Auf Einzeichenbefehle folgt kein Abschlusszeichen. Die Antwort auf einen Einzeichenbefehl enthält hingegen ein Abschlusszeichen.

Das Argument <AxisID> wird für die logischen Achsen des Controllers verwendet. Je nach Controller kann die Achsenkennung aus bis zu 16 Zeichen bestehen. Alle alphanumerischen Zeichen und der Unterstrich sind erlaubt. Die vom C-863 unterstützten Kennungen sind im Abschnitt "Kommandierbare Elemente" (S. 20) beschrieben.

Beispiel 1:

Achse 1 soll zur Position 10.0 bewegt werden. Die Einheit hängt vom Controller ab (z. B. µm oder mm).

Senden: MOV SP1 SP10.0 LF

Mehr als ein Befehlskürzel pro Zeile ist nicht erlaubt. Mehrere Gruppen von Argumenten sind nach einem Befehlskürzel erlaubt.

Beispiel 2:

Zwei Achsen, die mit demselben Controller verbunden sind, sollen bewegt werden:

Senden: `MOV[SP]1[SP]17.3[SP]2[SP]2.05[LF]`

Wenn ein Teil der Befehlszeile nicht ausgeführt werden kann, wird die gesamte Zeile nicht ausgeführt.

Wenn alle Argumente optional sind und weggelassen werden, wird der Befehl für alle möglichen Werte der Argumente ausgeführt.

Beispiel 3:

Alle Parameter im flüchtigen Speicher sollen zurückgesetzt werden.

Senden: `RPA[LF]`

Beispiel 4:

Die Position aller Achsen soll abgefragt werden.

Senden: `POS?[LF]`

Die Antwort-Syntax lautet wie folgt:

`[<Argument>[{"[SP]<Argument>"}]=""]<Wert>[LF]`

In mehrzeiligen Antworten wird in der letzten Zeile das Leerzeichen von dem Abschlusszeichen weggelassen:

`[{"<Argument>[{"[SP]<Argument>"}]=""]<Wert>[SP] [LF]`

`[<Argument>[{"[SP]<Argument>"}]=""]<Wert>[LF]` für die letzte Zeile!

In der Antwort werden die Argumente in derselben Reihenfolge aufgelistet wie im Abfragebefehl.

Abfragebefehl:

`CMD?[SP]<Arg3>[SP]<Arg1>[SP]<Arg2>[LF]`

Antwort auf diesen Befehl:

`<Arg3>="<Wert3>[SP] [LF]`

`<Arg1>="<Wert1>[SP] [LF]`

`<Arg2>="<Wert2> [LF]`

Beispiel 5:

Senden: TSP?SP2SP1LF

Empfangen: 2=-1158.4405SP LF

1=+0000.0000LF

INFORMATION

Beim C-863 kann nur ein einzelnes Element pro Befehlszeile adressiert werden (z. B. Achse, Kanal oder Parameter).

Beispiel:

Durch Senden der Befehlszeile

SEP 100 1 0x32 0

wird ein neuer Wert des Parameters 0x32 für Achse 1 im permanenten Speicher gespeichert,

das Senden der Befehlszeile

SEP 100 1 0x32 0 1 0x14 1

ist jedoch nicht möglich, weil zwei Parameter geändert werden sollen.

Wenn der Befehl dies unterstützt, können auch alle Elemente durch Weglassen der Elementkennung adressiert werden.

Beispiel:

Durch Senden der Befehlszeile

SEP?

werden die Werte aller Parameter aus dem permanenten Speicher abgefragt.

9.3 Empfänger- und Senderadresse

Grundsätzlich sind die Adressen des anzusprechenden Controllers (Empfänger) und des Senders in jeder Befehlszeile erforderlich. Dies gilt selbst für Einzeichenbefehle (z. B. #4), oder für die Aufzeichnung von Makros. Weil aber nur der PC Befehlszeilen an die Controller senden darf, kann seine Adresse (0) weggelassen werden. Jedoch sind sowohl die Empfänger- als auch die Senderadresse in jeder Controllerantwort enthalten. Mehrzeilige Antworten enthalten die Empfänger- und Senderadresse nur in der ersten Zeile.

Beispiel:

In einem Terminal-Programm wie z. B. PITerminal wird mit dem Befehl *IDN? die Ident-Bezeichnung eines Controllers mit der Adresse 2 (hier: ein C-863.11) abgefragt.

Senden: `2 0 *IDN?`

oder

Senden: `2 *IDN?`

Die Antwort lautet in beiden Fällen:

```
0 2 (c)2011 Physik Instrumente(PI) Karlsruhe, C-
863.11,0,1.2.0.0
```

Ausnahme:

Die Empfängeradresse kann weggelassen werden, wenn der anzusprechende Controller die Adresse 1 hat, selbst wenn dieser Controller Bestandteil eines Daisy-Chain-Netzwerks ist. Wenn die Empfängeradresse beim Ansprechen des Controllers weggelassen wird, werden Empfänger- und Senderadresse auch in der Antwort des Controllers weggelassen.

Beispiel:

Senden: `*IDN?`

Der Controller mit der Adresse 1 (hier: ein C-863.11) antwortet:

```
@2011 Physik Instrumente(PI) Karlsruhe, C-863.11,0,1.2.0.0
```

Senden: `1 *IDN?`

Derselbe Controller antwortet:

```
0 1 (c)2011 Physik Instrumente(PI) Karlsruhe, C-
863.11,0,1.2.0.0
```

Informationen zur Einstellung der Controlleradresse finden Sie im Abschnitt "DIP-Schalter-Einstellungen anpassen" (S. 74). Die Controlleradresse kann im Bereich von 1 bis 16 liegen; die Standardadresse ist 1. Der PC hat immer die Adresse 0. Mit der Sammeladresse 255 können alle Controller in einem Daisy-Chain-Netzwerk gleichzeitig angesprochen werden, wobei dann jedoch keine Antworten an den PC gesendet werden.

9.4 Variablen

Für eine flexiblere Programmierung unterstützt der C-863 Variablen. Während globale Variablen immer verfügbar sind, gelten lokale Variablen immer nur für ein bestimmtes Makro. Typischerweise werden Variablen in Makros verwendet.

Variablen sind nur im flüchtigen Speicher (RAM) vorhanden. Die Variablenwerte haben den Datentyp STRING.

Für Variablennamen gelten folgende Konventionen:

- Variablennamen dürfen keine Sonderzeichen enthalten (insbesondere kein "\$").
- Höchstens 8 Zeichen sind erlaubt.
- Die Namen von globalen Variablen können aus den Zeichen A bis Z und 0 bis 9 bestehen. Sie müssen mit einem Buchstaben beginnen.
- Die Namen von lokalen Variablen dürfen keine Buchstaben enthalten. Mögliche Zeichen sind 0 bis 9.
- Der Variablenname kann auch über den Wert einer anderen Variablen angegeben werden.

Wenn der Wert einer Variablen verwendet werden soll, muss folgende Schreibweise angewandt werden:

- Dem Variablennamen muss ein "\$" vorangestellt werden.
- Variablennamen, die aus mehreren Zeichen bestehen, müssen in geschweifte Klammern gesetzt werden.

Wenn der Variablenname aus nur einem Zeichen besteht, können die geschweiften Klammern weggelassen werden.

Wenn die geschweiften Klammern bei Variablennamen weggelassen werden, die aus mehreren Zeichen bestehen, wird das erste Zeichen nach dem "\$" als der Variablenname interpretiert.

Lokale Variablen:

- Lokale Variablen können nur in Makros verwendet werden.
- Derzeit unterstützt die Controllerfirmware drei lokale Variablen: 0, 1 und 2.
- Die Werte der lokalen Variablen 1 und 2 werden als Argumente der Befehle `MAC START` oder `MAC NSTART` beim Start des Makros angegeben.

Befehlsformate:

```
MAC START <macroname> [<String1> [<String2>]]
```



```
MAC NSTART <macroname> <uint> [<String1> [<String2>]]
```

<STRING1> und <STRING2> geben die Werte für die im Makro verwendeten lokalen Variablen 1 und 2 an. <STRING1> und <STRING2> können direkt oder über Variablenwerte angegeben werden. <uint> bestimmt, wievielmals das Makro ausgeführt werden soll. Weitere Informationen finden Sie in der Beschreibung des Befehls **MAC** (S. 219).

- Die lokale Variable 0 kann nur gelesen werden. Ihr Wert gibt an, wieviele Argumente (d.h. Werte von lokalen Variablen) beim Start des Makros angegeben wurden.
- Innerhalb eines Makros können die Werte lokaler Variablen mit den Befehlen **ADD** (S. 171), **CPY** (S. 175) und **VAR** (S. 255) verändert und mit dem Befehl **VAR** gelöscht werden (Ausnahme: lokale Variable 0).
- Solange das Makro ausgeführt wird, können die Werte der lokalen Variablen abgefragt werden mit:

```
VAR? 0
```

```
VAR? 1
```

```
VAR? 2
```

Die Abfragen können innerhalb oder außerhalb des Makros gesendet werden.

Globale Variablen:

- Globale Variablen können innerhalb und außerhalb von Makros verwendet werden.
- Die maximale Anzahl globaler Variablen beträgt 10.
- Globale Variablen werden mit den Befehlen **ADD**, **CPY** oder **VAR** angelegt und verändert. Sie können mit dem Befehl **VAR** gelöscht werden.
- Die Variablenwerte können mit **VAR?** abgefragt werden.

9.5 Befehlsübersicht

Befehl	Format	Beschreibung
#4 (S. 166)	#4	Request Status Register
#5 (S. 167)	#5	Request Motion Status
#7 (S. 167)	#7	Request Controller Ready Status
#8 (S. 168)	#8	Query If Macro Is Running
#24 (S. 168)	#24	Stop All Axes
*IDN? (S. 169)	*IDN?	Get Device Identification
ACC (S. 169)	ACC {<AxisID> <Acceleration>}	Set Closed-Loop Acceleration
ACC? (S. 170)	ACC? [{<AxisID>}]	Get Closed-Loop Acceleration
ADD (S. 171)	ADD <Variable> <FLOAT1> <FLOAT2>	Add and Save To Variable
BRA (S. 174)	BRA {<AxisID> <BrakeState>}	Set Brake Activation State
BRA? (S. 174)	BRA? [{<AxisID>}]	Get Brake Activation State
CPY (S. 175)	CPY <Variable> <CMD?>	Copy Into Variable
CST? (S. 176)	CST? [{<AxisID>}]	Get Assignment Of Stages To Axes
CSV? (S. 177)	CSV?	Get Current Syntax Version
CTO (S. 177)	CTO {<TrigOutID> <CTOPam> <Value>}	Set Configuration Of Trigger Output
CTO? (S. 182)	CTO? [{<TrigOutID> <CTOPam>}]	Get Configuration Of Trigger Output
DEC (S. 183)	DEC {<AxisID> <Deceleration>}	Set Closed-Loop Deceleration
DEC? (S. 183)	DEC? [{<AxisID>}]	Get Closed-Loop Deceleration
DEL (S. 184)	DEL <uint>	Delay The Command Interpreter
DFH (S. 184)	DFH [{<AxisID>}]	Define Home Position
DFH? (S. 186)	DFH? [{<AxisID>}]	Get Home Position Definition
DIO (S. 187)	DIO {<DIOID> <OutputOn>}	Set Digital Output Lines
DIO? (S. 188)	DIO? [{<DIOID>}]	Get Digital Input Lines
DRC (S. 189)	DRC {<RecTableID> <Source> <RecOption>}	Set Data Recorder Configuration

Befehl	Format	Beschreibung
DRC? (S. 190)	DRC? [{<RecTableID>}]	Get Data Recorder Configuration
DRR? (S. 191)	DRR? [<StartPoint> <NumberOfPoints> [{<RecTableID>}]]	Get Recorded Data Values
DRT (S. 193)	DRT {<RecTableID> <TriggerSource> <Value>}	Set Data Recorder Trigger Source
DRT? (S. 194)	DRT? [{<RecTableID>}]	Get Data Recorder Trigger Source
ERR? (S. 195)	ERR?	Get Error Number
FED (S. 195)	FED {<AxisID> <EdgeID> <Param>}	Find Edge
FNL (S. 197)	FNL [{<AxisID>}]	Fast Reference Move To Negative Limit
FPL (S. 199)	FPL [{<AxisID>}]	Fast Reference Move To Positive Limit
FRF (S. 200)	FRF [{<AxisID>}]	Fast Reference Move To Reference Switch
FRF? (S. 201)	FRF? [{<AxisID>}]	Get Referencing Result
GOH (S. 202)	GOH [{<AxisID>}]	Go To Home Position
HDR? (S. 203)	HDR?	Get All Data Recorder Options
HLP? (S. 204)	HLP?	Get List of Available Commands
HLT (S. 205)	HLT [{<AxisID>}]	Halt Motion Smoothly
HPA? (S. 205)	HPA?	Get List Of Available Parameters
JAS? (S. 207)	JAS? [{<JoystickID> <JoystickAxis>}]	Query Joystick Axis Status
JAX (S. 208)	JAX <JoystickID> <JoystickAxis> <AxisID>	Set Axis Controlled By Joystick
JAX? (S. 209)	JAX? [{<JoystickID> <JoystickAxis>}]	Get Axis Controlled By Joystick
JBS? (S. 209)	JBS? [{<JoystickID> <JoystickButton>}]	Query Joystick Button Status
JDT (S. 210)	JDT {<JoystickID> <JoystickAxis> <uint>}	Set Joystick Default Lookup Table
JLT (S. 211)	JLT <JoystickID> <JoystickAxis> <Addr> <floatn>	Fill Joystick Lookup Table
JLT? (S. 213)	JLT? [<StartPoint> <NumberOfPoints> [{<JoystickID> <JoystickAxis>}]]	Get Joystick Lookup Table Values
JON (S. 214)	JON {<JoystickID> <uint>}	Set Joystick Activation Status

Befehl	Format	Beschreibung
JON? (S. 216)	JON? [{<JoystickID>}]	Get Joystick Activation Status
JRC (S. 216)	JRC <Jump> <CMD?> <OP> <Value>	Jump Relatively Depending On Condition
LIM? (S. 218)	LIM? [{<AxisID>}]	Indicate Limit Switches
MAC (S. 219)	MAC <keyword> {<parameter>} insbesondere: MAC BEG <macroname> MAC DEF <macroname> MAC DEF? MAC DEL <macroname> MAC END MAC ERR? MAC NSTART <macroname> <uint> [<String1> [<String2>]] MAC START <macroname> [<String1> [<String2>]]	Call Macro Function
MAC? (S. 222)	MAC? [<macroname>]	List Macros
MEX (S. 223)	MEX <CMD?> <OP> <Value>	Stop Macro Execution Due To Condition
MOV (S. 225)	MOV {<AxisID> <Position>}	Set Target Position
MOV? (S. 226)	MOV? [{<AxisID>}]	Get Target Position
MVR (S. 227)	MVR {<AxisID> <Distance>}	Set Target Relative To Current Position
ONT? (S. 228)	ONT? [{<AxisID>}]	Get On-Target State
POS (S. 229)	POS {<AxisID> <Position>}	Set Real Position
POS? (S. 230)	POS? [{<AxisID>}]	Get Real Position
RBT (S. 230)	RBT	Reboot System
RMC? (S. 231)	RMC?	List Running Macros
RON (S. 231)	RON {<AxisID> <ReferenceOn>}	Set Reference Mode
RON? (S. 232)	RON? [{<AxisID>}]	Get Reference Mode
RPA (S. 233)	RPA [{<ItemID> <PamID>}]	Reset Volatile Memory Parameters
RTR (S. 234)	RTR <RecordTableRate>	Set Record Table Rate
RTR? (S. 234)	RTR?	Get Record Table Rate

Befehl	Format	Beschreibung
SAI (S. 235)	SAI {<AxisID> <NewIdentifier>}	Set Current Axis Identifiers
SAI? (S. 235)	SAI? [ALL]	Get List Of Current Axis Identifiers
SEP (S. 236)	SEP <Pswd> {<ItemID> <PamID> <PamValue>}	Set Non-Volatile Memory Parameters
SEP? (S. 237)	SEP? [{<ItemID> <PamID>}]	Get Non-Volatile Memory Parameters
SMO (S. 238)	SMO {<AxisID> <ControlValue>}	Set Open-Loop Control Value
SMO? (S. 240)	SMO? [{<AxisID>}]	Get Control Value
SPA (S. 240)	SPA {<ItemID> <PamID> <PamValue>}	Set Volatile Memory Parameters
SPA? (S. 243)	SPA? [{<ItemID> <PamID>}]	Get Volatile Memory Parameters
SRG? (S. 244)	SRG? {<AxisID> <RegisterID>}	Query Status Register Value
STE (S. 246)	STE <AxisID> <Amplitude>	Start Step And Response Measurement
STP (S. 247)	STP	Stop All Axes
SVO (S. 247)	SVO {<AxisID> <ServoState>}	Set Servo Mode
SVO? (S. 249)	SVO? [{<AxisID>}]	Get Servo Mode
TAC? (S. 249)	TAC?	Tell Analog Channels
TAV? (S. 249)	TAV? [{<AnalogInputID>}]	Get Analog Input Voltage
TIO? (S. 250)	TIO?	Tell Digital I/O Lines
TMN? (S. 251)	TMN? [{<AxisID>}]	Get Minimum Commandable Position
TMX? (S. 251)	TMX? [{<AxisID>}]	Get Maximum Commandable Position
TNR? (S. 252)	TNR?	Get Number Of Record Tables
TRO (S. 252)	TRO {<TrigOutID> <TrigMode>}	Set Trigger Output State
TRO? (S. 253)	TRO? [{<TrigOutID>}]	Get Trigger Output State
TRS? (S. 253)	TRS? [{<AxisID>}]	Indicate Reference Switch
TVI? (S. 254)	TVI?	Tell Valid Character Set For Axis Identifiers
VAR (S. 255)	VAR <Variable> <String>	Set Variable Value
VAR? (S. 256)	VAR? [{<Variable>}]	Get Variable Value
VEL (S. 257)	VEL {<AxisID> <Velocity>}	Set Closed-Loop Velocity

Befehl	Format	Beschreibung
VEL? (S. 258)	VEL? [{<AxisID>}]	Get Closed-Loop Velocity
VER? (S. 258)	VER?	Get Versions Of Firmware And Drivers
WAC (S. 259)	WAC <CMD?> <OP> <Value>	Wait For Condition
WPA (S. 260)	WPA <Pswd> [{<ItemID> <PamID>}]	Save Parameters To Non-Volatile Memory

9.6 Befehlsbeschreibungen für GCS 2.0

#4 (Request Status Register)

Beschreibung: Fragt die Systemstatus-Information ab.

Format: #4

Argumente: Keine

Antwort: Die Antwort ist bit-codiert. Für die individuellen Codes siehe unten.

Hinweise: Dieser Befehl ist funktionsgleich mit SRG? (S. 244), aber es wird nur ein Zeichen über die Schnittstelle gesendet. Deshalb kann #4 auch verwendet werden, wenn der Controller zeitaufwändige Aufgaben ausführt.

Für den C-863 ist die Antwort die Summe der folgenden Codes in Hexadezimalformat:

Bit	15	14	13	12	11	10	9	8
Beschreibung	On-Target-Status	Bestimmt den Referenzwert	In Bewegung	Servo-modus Ein	-	-	-	Fehlerflag
Bit	7	6	5	4	3	2	1	0
Beschreibung	Digitale Eingangsleitung 4	Digitale Eingangsleitung 3	Digitale Eingangsleitung 2	Digitale Eingangsleitung 1	-	Pos. Endschalter	Referenzschalter	Neg. Endschalter

Beispiel: Senden: #4
 Empfangen: 0x9005

Hinweis: Die Antwort wird im Hexadezimalformat angegeben. Sie besagt: die Achse ist an der Zielposition (On-Target-Status = wahr), der Servomodus ist eingeschaltet, es ist kein Fehler aufgetreten, der Status der digitalen Eingangsleitungen 1 bis 4 ist low, und der Versteller befindet sich auf der positiven Seite des Referenzschalters (Endschalter sind nicht aktiv, beachten Sie, dass die Logik der Signale in diesem Beispiel invertiert ist).

#5 (Request Motion Status)

Beschreibung: Fragt den Bewegungsstatus der Achsen ab.

Format: #5

Argumente: Keine

Antwort: Die Antwort <uint> ist bit-codiert und wird als hexadezimale Summe der folgenden Codes zurückgegeben:

1 = erste Achse bewegt sich
 2 = zweite Achse bewegt sich
 4 = dritte Achse bewegt sich
 ...

Beispiele: 0 gibt an, dass die Bewegung aller Achsen abgeschlossen ist
 3 gibt an, dass sich die erste und die zweite Achse bewegen

#7 (Request Controller Ready Status)

Beschreibung: Fragt den Bereitschaftsstatus des Controllers ab (prüft, ob Controller zum Ausführen eines neuen Befehls bereit ist).

Hinweis: Verwenden Sie #5 (S. 167) anstelle von #7, um zu verifizieren, ob die Bewegung beendet ist.

Format: #7

Argumente:	Keine
Antwort:	B1h (ASCII Zeichen 177 = "±" in Windows) wenn Controller bereit ist B0h (ASCII Zeichen 176 = "°" in Windows) wenn Controller nicht bereit ist (z. B. führt eine Referenzfahrt aus)
Fehlersuche:	Die Antwortzeichen können in nicht-westeuropäischen Zeichensätzen oder anderen Betriebssystemen unterschiedlich angezeigt werden.

#8 (Query if Macro Is Running)

Beschreibung:	Prüft, ob ein Makro auf dem Controller ausgeführt wird.
Format:	#8
Argumente:	Keine
Antwort:	<uint>=0 wenn kein Makro ausgeführt wird <uint>=1 wenn ein Makro aktuell ausgeführt wird

#24 (Stop All Axes)

Beschreibung:	Stoppt alle Achsen abrupt. Nähere Angaben siehe Hinweise unten. Setzt den Fehlercode auf 10. Dieser Befehl ist funktionsgleich mit STP (S. 247), aber es wird nur ein Zeichen über die Schnittstelle gesendet.
Format:	#24
Argumente:	Keine
Antwort:	Keine
Hinweise:	#24 stoppt jede Bewegung, die durch Bewegungsbefehle (z. B. MOV (S. 225), MVR (S. 227), GOH (S. 202), STE (S. 246), SMO (S. 238)), Befehle zur Referenzwertbestimmung (FNL (S. 197), FPL (S. 199), FRF

(S. 200)) und Makros (MAC (S. 219)) verursacht wird.
Stoppt auch die Makroausführung.

Nachdem die Achsen gestoppt wurden, werden ihre
Zielpositionen auf ihre aktuellen Positionen gesetzt.

HLT (S. 205) stoppt im Gegensatz zu #24 die Bewegung
mit angegebener Abbremsung im Hinblick auf die
Systemträgheit.

***IDN? (Get Device Identification)**

Beschreibung:	Fragt die Ident-Bezeichnung des Geräts ab.
Format:	*IDN?
Argumente:	Keine
Antwort:	Mit dem Abschlusszeichen (line feed) beendeter einzeliger Text mit Controllernamen, Seriennummer und Firmwareversion
Hinweise:	Beim C-863 antwortet *IDN? etwa Folgendes:

```
(c)2013 Physik Instrumente(PI) Karlsruhe,  
C-863,112048255,1.2.0.0
```

ACC (Set Closed-Loop Acceleration)

Beschreibung:	Setzt die Beschleunigung für die angegebenen Achsen.
---------------	--

Die ACC-Einstellung wird nur wirksam, wenn sich die
angegebene Achse im geregelten Betrieb befindet
(Servomodus EIN).

ACC kann verändert werden, während die Achse sich
bewegt.

Format:	ACC {<AxisID> <Acceleration>}
---------	-------------------------------

Argumente:	<AxisID> ist eine Achse des Controllers <Acceleration> ist der Wert der Beschleunigung in physikalischen Einheiten pro s^2.
Antwort:	Keine
Fehlersuche:	Unzulässige Achsenkennungen
Hinweise:	Der kleinstmögliche Wert für <Acceleration> ist 0. ACC ändert den Wert des Parameters Closed-Loop Acceleration (Phys. Unit/s²) (ID 0xB) im flüchtigen Speicher (kann mit WPA (S. 260) als Standard gespeichert und mit SPA (S. 240) und SEP (S. 236) geändert werden). Der maximale Wert, der mit dem Befehl ACC gesetzt werden kann, wird durch den Parameter Maximum Closed-Loop Acceleration (Phys. Unit/s²) (ID 0x4A) angegeben (kann mit SPA (S. 240) und SEP (S. 236) geändert werden).

ACC? (Get Closed-Loop Acceleration)

Beschreibung:	Frägt den mit ACC gesetzten Wert der Beschleunigung für den geregelten Betrieb ab. Werden alle Argumente weggelassen, wird der mit ACC gesetzte Wert aller Achsen abgefragt.
Format:	ACC? [{<AxisID>}]
Argumente:	<AxisID> ist eine Achse des Controllers
Antwort:	{<AxisID>="<float> LF}
	wobei <float> der mit ACC gesetzte Beschleunigungswert für den geregelten Betrieb ist, in physikalischen Einheiten pro s^2.

ADD (Add And Save To Variable)

Beschreibung: Addiert zwei Werte und speichert das Ergebnis als Variable (S. 160).

Die Variable ist nur im flüchtigen Speicher (RAM) vorhanden.

Format: ADD <Variable> <FLOAT1> <FLOAT2>

Argumente: <Variable> ist der Name der Variable, in der das Ergebnis gespeichert werden soll.

<FLOAT1> ist der erste Summand.

<FLOAT2> ist der zweite Summand.

Für die Summanden werden Gleitkommazahlen erwartet. Sie können direkt angegeben werden oder über den Wert einer Variablen.

Antwort: Keine

Hinweise: Lokale Variablen können mit ADD nur in Makros gesetzt werden.

Beispiel 1: Wert \$B wird zu Wert \$A addiert und das Ergebnis wird als Variable C gespeichert:

```
ADD C $A $B
```

Beispiel 2: Der Name der Variablen, in die das Ergebnis kopiert werden soll, wird über den Wert einer anderen Variablen angegeben:

Senden: VAR?

Empfangen:

```
A=468
```

```
B=123
```

```
3Z=WORKS
```

Senden: ADD A\${3Z} \$A \$B

Senden: VAR?

Empfangen:

A=468

B=123

AWORKS=591

3Z=WORKS

Senden: ADD \${3Z} \$A \$B

Senden: VAR?

Empfangen:

A=468

B=123

AWORKS=591

WORKS=591

3Z=WORKS

Beispiel 3:

Unter Verwendung der nachstehenden Makros ist es möglich, mit LEDs, die an die digitalen Ausgangsleitungen des Controllers angeschlossen sind, ein "Blinklicht" zu erzeugen. \$1 und \$2 sind Werte lokaler Variablen und müssen als Argumente des Befehls MAC START oder MAC NSTART beim Start der Makros angegeben werden (siehe unten).

DIO 0 <bitmask>: Setzt die Ausgangskanäle gemäß <bitmask>. "DIO 0 5" aktiviert beispielsweise die Kanäle 1 und 3 und deaktiviert alle anderen Kanäle (5 ist 0000 0101 in Binärschreibweise).

Führen Sie folgende Schritte zur Implementierung des "Blinklichts" durch:

1. Schreiben Sie das Makro "STEPS":

```
MAC BEG STEPS
```

```
DIO 0 $1
```

```
ADD 1 $1 1
```

```
DEL $2
```

```
JRC -3 VAR? 1 <= 15
```

```
ADD 1 $1 -1
```

```
DIO 0 $1
```

```
DEL $2
```

```
JRC -3 VAR? 1 > 0
```

```
MAC END
```

2. Schreiben Sie das Makro "TEST":

```
MAC BEG TEST
```

```
MAC START STEPS 0 $1
```

```
ADD 1 $1 10
```

```
JRC -2 VAR? 1 < 110
```

```
VAR 1 10
```

```
ADD 2 $2 -1
```

```
JRC -5 VAR? 2 > 0
```

```
MAC END
```

3. Starten Sie das Makro TEST mit Argumenten, die die Variablenwerte \$1 und \$2 definieren:

```
MAC START Test 10 50
```

Bedeutung der Variablenwerte dabei:

\$1: Verzögerung in ms zwischen jedem Schritt im Makro STEPS. Der Wert wird durch das Makro TEST jeweils um 10 erhöht, bis er 110 erreicht hat.

\$2: Anzahl der Wiederholungen der gesamten "Blinklicht"-Prozedur.

BRA (Set Brake Activation State)

Beschreibung: Aktiviert/deaktiviert Bremse für angegebene Achsen.

Format: BRA {<AxisID> <BrakeState>}

Argumente: <AxisID> ist eine Achse des Controllers

<BrakeState> kann die folgenden Werte haben:

0 = Bremse deaktiviert

1 = Bremse aktiviert

Antwort: Keine

Fehlersuche: Unzulässige Achsenkennung

Hinweise: Die Bremse kann nur verwendet werden, wenn der Parameter 0x1A (**Has Brake?**) den Wert 1 ("ja") hat. Wenn der Parameter 0x1A (**Has Brake?**) den Wert 1 ("ja") hat, gilt Folgendes:

- Die Bremse kann mit BRA nur aktiviert oder deaktiviert werden, wenn der Servomodus ausgeschaltet ist. Sichern Sie den Versteller gegen unbeabsichtigte Bewegungen, bevor Sie die Bremse mit BRA deaktivieren!
- Das Setzen des Servomodus mit SVO (S. 247) beeinflusst den Aktivierungszustand der Bremse:
 - Einschalten des Servomodus deaktiviert die Bremse.
 - Ausschalten des Servomodus aktiviert die Bremse.
- Beim Auftreten eines Bewegungsfehlers (S. 99) wird der Servomodus ausgeschaltet, und die Bremse wird aktiviert.

BRA? (Get Brake Activation State)

Beschreibung: Fragt den Status der Bremsenaktivierung für die angegebenen Achsen ab.

Werden keine Argumente angegeben, wird der Status aller Achsen abgefragt.

Format:	BRA? [{<AxisID>}]
Argumente:	<AxisID> ist eine Achse des Controllers
Antwort:	{<AxisID>}"<BrakeState> LF}
	wobei
	<BrakeState> der aktuelle Status der Bremsenaktivierung der Achse ist: 0 = Bremse deaktiviert 1 = Bremse aktiviert
Fehlersuche:	Unzulässige Achsenkennung

CPY (Copy Into Variable)

Beschreibung:	Kopiert eine Antwort auf einen Befehl in eine Variable (S. 160). Die Variable ist nur im flüchtigen Speicher (RAM) vorhanden.
Format:	CPY <Variable> <CMD?>
Argumente:	<Variable> ist der Name der Variablen, in die die Befehlsantwort kopiert werden soll. <CMD?> ist ein Abfragebefehl in seiner üblichen Schreibweise. Die Antwort muss ein einzelner Wert sein (und nicht mehr).
Antwort:	Keine
Hinweise:	Lokale Variablen können mit CPY nur in Makros gesetzt werden.

Beispiel 1: Unter Verwendung des nachstehenden Makros ist es möglich, die digitalen Eingangs- und Ausgangsleitungen des Controllers durchzukontaktieren. 1 ist eine lokale Variable, deren Wert als Argument des Befehls MAC START oder MAC NSTART beim Start der Makros angegeben werden muss.

Schreiben Sie das Makro "connect":

```
MAC BEG connect
CPY 1 DIO? 0
DIO 0 $1
MAC START CONNECT
MAC END
```

Beispiel 2: Es ist möglich, den Wert einer Variable (z. B. SOURCE) in eine andere Variable (z. B. TARGET) zu kopieren:

```
CPY TARGET VAR? SOURCE
```

CST? (Get Assignment Of Stages To Axes)

Beschreibung: Fragt den Namen des Verstellertyps ab, der an die angegebene Achse angeschlossen ist.

Format: CST? [{<AxisID>}]

Argumente: <AxisID> ist eine Achse des Controllers

Antwort: {<AxisID>="<string> LF}

wobei

<string> der Name des Verstellertyps ist, der der Achse zugewiesen ist.

Hinweise: Der Verstellernamen wird vom Parameter **Stage Name** gelesen (ID 0x3C), dessen Werkseinstellungswert "DEFAULT_STAGE" ist. Sie können mit SPA (S. 240) oder SEP (S. 236) den Parameterwert auf den Namen Ihres Verstellers setzen. Einzelheiten finden Sie in der Parameterübersicht (S. 288).

CSV? (Get Current Syntax Version)

Beschreibung:	Fragt die GCS-Syntaxversion ab, die in der Firmware verwendet wird.
Format:	CSV?
Argumente:	Keine
Antwort:	Die aktuelle GCS-Syntaxversion
Hinweise:	1.0 (für GCS 1.0) oder 2.0 (für GCS 2.0) sind mögliche Antworten.

CTO (Set Configuration Of Trigger Output)

Beschreibung:	Konfiguriert die Bedingungen für die Triggerausgabe für die angegebene digitale Ausgangsleitung.
Format:	CTO {<TrigOutID> <CTOPam> <Value>}
Argumente:	<TrigOutID> ist eine digitale Ausgangsleitung des Controllers; weitere Angaben siehe unten. <CTOPam> ist die ID des CTO-Parameters im Dezimalformat; vorhandene IDs siehe unten. <Value> ist der Wert, auf den der CTO-Parameter gesetzt wird; siehe unten.
Antwort:	Keine
Hinweise:	Die Bedingungen für die Triggerausgabe werden aktiv, wenn sie mit TRO (S. 252) aktiviert werden. Verwenden Sie DIO (S. 187) nicht für digitale Ausgangsleitungen, bei denen die Triggerausgabe mit TRO aktiviert ist. Die CTO-Einstellungen gehen verloren, wenn C-863 ausgeschaltet oder neugestartet wird. Durch Speichern in einem Makro können sie auf einfache Weise beibehalten werden.

Vorhandene
Ausgangs-
leitungen und
Triggerbe-
dingungen:

<TrigOutID> entspricht den digitalen Ausgangsleitungen 1 bis 4, IDs = 1 bis 4; siehe "I/O" (S. 316).

<CTOPam> Parameter-IDs, verfügbar für C-863:

- 1 = TriggerStep
- 2 = Axis
- 3 = TriggerMode
- 7 = Polarity
- 8 = StartThreshold
- 9 = StopThreshold
- 10 = TriggerPosition

<Value> verfügbar für die entsprechende <CTOPam> ID:

für TriggerStep: Strecke

für Axis: die Kennung der Achse, die mit der digitalen Ausgangsleitung verbunden werden soll. Irrelevant für den Triggermodus MotionError.

für TriggerMode (Standardwert ist 0):

- 0 = PositionDistance;
es wird jedes Mal ein Triggerpuls geschrieben, wenn die Achse die Strecke TriggerStep zurückgelegt hat (<CTOPam> ID 1). Optional können Werte für StartThreshold und StopThreshold (<CTOPam> IDs 8 und 9) definiert werden, um die Triggerausgabe nur für einen begrenzten Positionsbereich und eine bestimmte Bewegungsrichtung zu aktivieren (negativ oder positiv; Hinweis: Falls sich die Bewegungsrichtung umkehrt, bevor die Achsenposition den Stop-Schwellenwert erreicht hat, werden weiterhin Triggerpulse erzeugt). Werden StartThreshold und StopThreshold auf den gleichen Wert gesetzt, werden sie nicht verwendet.
- 2 = OnTarget;
der On-Target-Status der gewählten Achse wird an die gewählte digitale Ausgangsleitung übertragen (dieser Status kann auch mit dem Befehl ONT? gelesen werden).

- 5 = MotionError;
die gewählte digitale Ausgangsleitung wird aktiv, wenn ein Bewegungsfehler auftritt. Die Leitung bleibt aktiv, bis der Fehlercode auf 0 zurückgesetzt wird (durch eine Abfrage mit ERR?).
- 6 = InMotion;
die gewählte digitale Ausgangsleitung ist solange aktiv, wie die gewählte Achse in Bewegung ist (der Bewegungszustand kann auch mit #4, #5 oder dem Befehl SRG? gelesen werden).
- 7 = Position+Offset;
der erste Triggerpuls wird geschrieben, wenn die Achse die durch TriggerPosition (<CTOPam> ID 10) angegebene Position erreicht hat. Die nächsten Triggerpulse werden jeweils geschrieben, wenn die Achsenposition gleich der Summe der letzten gültigen Triggerposition und der durch TriggerStep (<CTOPam> ID 1) angegebenen Strecke ist. Die Triggerausgabe wird beendet, wenn die Achsenposition den durch StopThreshold (<CTOPam> ID 9) angegebenen Wert übersteigt. Das Vorzeichen des Wertes TriggerStep bestimmt, für welche Bewegungsrichtung Triggerpulse ausgegeben werden sollen. Die Triggerverarbeitung erfolgt durch den DSP des C-863.

für Polarity (Standardwert ist 1): setzt die Signalarität für die digitale Ausgangsleitung

0 = low-aktiv

1 = high-aktiv

für StartThreshold/StopThreshold: Positionswert;
bei Verwendung für den Triggermodus PositionDistance müssen beide Schwellenwerte gesetzt werden, um den Positionsbereich und die Bewegungsrichtung für die Triggerausgabe zu bestimmen;
StopThreshold wird als Stopp-Bedingung für den Triggermodus Position+Offset verwendet

für TriggerPosition: Position, an der der erste Triggerpuls im Triggermodus Position+Offset auszugeben ist (Standardwert ist 0.0)

Anwendungsbeispiele und weitere Angaben siehe "Digitale Ausgangssignale" (S. 103) und nachstehende Zeilen.

Beispiel 1:

Ein Puls soll an der digitalen Ausgangsleitung 1 (ID 1) erzeugt werden, wenn Achse 1 eine Distanz von 0,05 µm zurückgelegt hat. Folgende Parameter müssen gesetzt werden:

TrigOutID = 1

Axis = 1

TriggerMode = 0

TriggerStep = 0.05

Senden: CT0 1 2 1

Senden: CT0 1 3 0

Senden: CT0 1 1 0.00005

Beispiel 2:

In diesem Beispiel soll die digitale Ausgangsleitung 1 von low auf high gesetzt werden, wenn Achse A ihre Bewegung beginnt. Folgende Parameter müssen gesetzt werden:

TrigOutID = 1

Axis = A (Achsenkennung wurde mit SAI geändert)

TriggerMode = 6

Polarity = high-aktiv

Sie müssen also Folgendes senden:

CT0 1 2 A

CT0 1 3 6

CT0 1 7 1

Beispiel 3: M-122.2DD wird mit Achse 1 verbunden. Die Referenzposition von M-122.2DD ist 18,5 mm. Von ihrer Referenzposition ausgehend soll die Achse abwechselnd vorwärts und rückwärts bewegt werden; Triggerpulse sollen für beide Bewegungsrichtungen in einem Bereich von 1 mm mit dem Triggermodus Position+Offset ausgegeben werden. Zu diesem Zweck werden zwei Makros in den Controller geschrieben. Das Makro TRIGREF initialisiert den Controller und kann auch als Startup-Makro festgelegt werden, während das Makro TRIGGER die Bewegung und somit die Triggerausgabe startet. Schreiben Sie die Makros wie nachstehend beschrieben. Nähere Angaben zu Makros siehe "Mit Makros arbeiten" (S. 133).

Stellen Sie sicher, dass die Geschwindigkeitseinstellung für die Achse zur CTO-Einstellung für *Stepsize* passt.

Empfohlener Wert:

Höchstgeschwindigkeit = $\text{Stepsize} * 20 \text{ kHz} / 2$

wobei 20 kHz die Frequenz des Regelzyklus des C-863 ist.

Bei einer Geschwindigkeit von 20 mm/s ergibt sich eine Frequenz des Triggersignals von 1 kHz.

- Zeichnen Sie ein Makro namens TRIGREF mit folgendem Inhalt auf:

```
CTO 1 3 7
```

```
SV0 1 1
```

```
FRF
```

```
TRO 1 1
```

```
MAC START TRIGGER
```

- Zeichnen Sie ein Makro namens TRIGGER mit folgendem Inhalt auf:

```
CTO 1 1 0.02
```

```
CTO 1 9 20.5
```

```
CTO 1 10 19.5
```

```
DEL 1000
```

```
DRT 0 2 0
```

```
MOV 1 20.51
```

```
WAC POS? 1 > 20.3
```

```

MEX CTO? 1 10 < 19.4
CTO 1 1 -0.02
CTO 1 9 19.5
CTO 1 10 20.5
DEL 1000
MOV 1 19.49
WAC POS? 1 < 19.5
MEX CTO? 1 10 > 19.6
MAC START TRIGGER

```

CTO? (Get Configuration Of Trigger Output)

Beschreibung: Fragt die Werte ab, die für die angegebenen Trigger-Ausgangsleitungen und Parameter gesetzt wurden.

Format: CTO? [{<TrigOutID> <CTOPam>}]

Argumente: <TrigOutID>: ist eine digitale Ausgangsleitung des Controllers; siehe CTO.

<CTOPam>: Parameter-ID; siehe CTO.

Werden alle Argumente weggelassen, enthält die Antwort die Werte für alle Parameter und alle Ausgangsleitungen.

Antwort: {<TrigOutID> <CTOPam>="<Value> LF}

Für <Value> siehe CTO.

DEC (Set Closed-Loop Deceleration)

Beschreibung: Setzt die Abbremsung für die angegebenen Achsen.

Die DEC-Einstellung wird nur wirksam, wenn sich die angegebene Achse im geregelten Betrieb befindet (Servomodus EIN).

DEC kann verändert werden, während die Achse sich bewegt.

Format: DEC {<AxisID> <Deceleration>}

Argumente: <AxisID> ist eine Achse des Controllers.

<Deceleration> ist der Wert der Abbremsung in physikalischen Einheiten pro s^2 .

Antwort: Keine

Fehlersuche: Unzulässige Achsenkennungen

Hinweise: Der kleinstmögliche Wert für <Deceleration> ist 0.

DEC ändert den Wert des Parameters **Closed Loop Deceleration (Phys. Unit/s²)** (ID 0xC) im flüchtigen Speicher (kann mit WPA (S. 260) als Standard gespeichert werden, kann auch mit SPA (S. 240) und SEP (S. 236) geändert werden).

Der maximale Wert, der mit dem Befehl DEC gesetzt werden kann, wird durch den Parameter **Maximum Closed-Loop Deceleration (Phys. Unit/s²)** (ID 0x4B) angegeben (kann mit SPA (S. 240) und SEP (S. 236) geändert werden).

DEC? (Get Closed-Loop Deceleration)

Beschreibung: Fragt den mit DEC gesetzten Wert der Abbremsung für den geregelten Betrieb ab.

Werden alle Argumente weggelassen, wird der mit DEC gesetzte Wert aller Achsen abgefragt.

Format: DEC? [{<AxisID>}]

Argumente: <AxisID> ist eine Achse des Controllers.

Antwort: {<AxisID>="<float> LF}

wobei

<float> der mit DEC gesetzte Wert der Abbremsung für den geregelten Betrieb ist, in physikalischen Einheiten pro s^2 .

DEL (Delay the Command Interpreter)

Beschreibung: Verzögert um <uint> Millisekunden.

Format: DEL <uint>

Argumente: <uint> ist der Verzögerungswert in Millisekunden.

Antwort: Keine

Hinweise: DEL kann nur in Makros verwendet werden. Verwechseln Sie nicht MAC DEL (löscht Makros) mit DEL (verzögert).

Weitere Informationen finden Sie in der Beschreibung des Befehls MAC (S. 219) und im Abschnitt "Controllermakros" (S. 131).

DFH (Define Home Position)

Beschreibung: Definiert die Nullposition für die angegebene Achse neu, indem der Positionswert an der aktuellen Position auf null gesetzt wird.

Werden alle Argumente weggelassen, definiert DHF die Nullposition aller Achsen neu.

Format: DFH [{<AxisID>}]

Argumente: <AxisID> ist eine Achse des Controllers

Antwort: keine

Fehlersuche:	Unzulässige Achsenkennung
Hinweise:	DFH setzt die aktuelle Position der Achse auf null und speichert den Positionswert, der beim Befehlsaufruf gültig war, als Offset im flüchtigen Speicher. Durch Addition dieses Offsets zur Antwort werden die Ausgabewerte folgender Befehle an die neue Nullposition angepasst: ▪ POS? (S. 230) (Abfrage der aktuellen Position) ▪ TMN? (S. 251) (Abfrage der kleinsten kommandierbaren Position) ▪ TMX? (S. 251) (Abfrage der größten kommandierbaren Position) DFH ändert nicht die Werte der Parameter für die Definition von Stellweg und Verfahrbereichsgrenzen (S. 35). Der Offset wird in folgenden Fällen auf null zurückgesetzt: ▪ Beim Einschalten und Neustart des C-863: für alle Achsen ▪ Bei der Referenzwertbestimmung: für die betroffene Achse
Beispiel:	Senden: MOV 1 9.87 Senden: POS? 1 Empfangen: 1=9.8700005 Senden: DFH? 1 Empfangen: 1=0.0000000 Senden: TMN? 1 Empfangen: 1=0.0000000 Senden: TMX? 1 Empfangen: 1=14.9999982 Hinweis: Achse 1 wird zur absoluten Position 9,87 mm bewegt. Anschließend werden die aktuelle Achsenposition (mit POS?), der aktuelle Offsetwert (mit DFH?) sowie die kleinste und größte kommandierbare Position (mit TMN? und TMX?) abgefragt.

```

Senden: DFH 1
Senden: POS? 1
Empfangen: 1=0.0000000
Senden: DFH? 1
Empfangen: 1=9.8700005
Senden: TMN? 1
Empfangen: 1=-9.8700005
Senden: TMX? 1
Empfangen: 1=5.1299978

```

Hinweis: Die Achse hat sich nicht bewegt. Mit DFH wurde die aktuelle Achsenposition als neue Nullposition festgelegt. Der Offsetwert für Achse 1 beträgt deshalb nun 9,87 mm. Die Werte für die kleinste und größte kommandierbare Position wurden durch Addition des Offsets an die neue Nullposition angepasst.

DFH? (Get Home Position Definition)

Beschreibung: Fragt ab, bei welcher Sensorposition die angegebene Achse ihre Nullposition hat.

Werden alle Argumente weggelassen, wird der Positionswert aller Achsen abgefragt.

Format: DFH? [{<AxisID>}]

Argumente: <AxisID> ist eine Achse des Controllers

Antwort: {<AxisID>="<SensorPosition> LF}

wobei

<SensorPosition> ist die Sensorposition, die zum Zeitpunkt der Verarbeitung des letzten DFH-Befehls gültig war. Dieser Sensorpositionswert wird intern als Offset für die Berechnung der aktuellen Achsenposition verwendet.

Fehlersuche: Unzulässige Achsenkennung

Hinweise: Die Sensorposition, die zum Zeitpunkt der Verarbeitung des letzten DFH-Befehls gültig war, ist als Offset im flüchtigen Speicher vorhanden. Der Offset wird in folgenden Fällen auf null zurückgesetzt:

- Beim Einschalten und Neustart des C-863: für alle Achsen
- Bei der Referenzwertbestimmung: für die betroffene Achse

Siehe DFH für ein Beispiel.

DIO (Set Digital Output Lines)

Beschreibung: Schaltet die angegebene(n) digitale(n) Ausgangsleitung(en) in den angegebenen Status.

Verwenden Sie TIO? (S. 250), um die Anzahl installierter digitaler I/O-Leitungen abzufragen.

Format: DIO {<DIOID> <OutputOn>}

Argumente: <DIOID> ist eine digitale Ausgangsleitung des Controllers; weitere Angaben siehe unten.

<OutputOn> ist der Status der digitalen Ausgangsleitung; nähere Angaben siehe unten.

Antwort: Keine

Hinweise: Mit dem Befehl DIO können die digitalen Ausgangsleitungen 1 bis 4, die sich auf der Buchse **I/O** (S. 316) befinden, aktiviert/deaktiviert werden. Mit dem C-863 können Sie pro Befehl DIO sowohl eine einzelne Leitung setzen als auch alle Leitungen auf einmal.

Die für die Leitungen zu verwendenden Kennungen <DIOID> sind 1 bis 4. Mit der Kennung 0 werden alle Leitungen gemäß einem durch <OutputOn> angegebenen Bitmuster gesetzt.

Wenn <OutputOn>=1 wird die Leitung auf HIGH/ON gesetzt, wenn <OutputOn>=0 wird sie auf LOW/OFF

gesetzt.

Verwenden Sie DIO nicht für Ausgangsleitungen, bei denen die Triggerausgabe mit TRO (S. 252) aktiviert ist.

DIO? (Get Digital Input Lines)

Beschreibung: Fragt den Status der angegebenen digitalen Eingangsleitungen ab.

Verwenden Sie TIO? (S. 250), um die Anzahl verfügbarer digitaler I/O-Leitungen abzufragen.

Format: DIO? [{<DIOID>}]

Argumente: <DIOID> ist die Kennung der digitalen Eingangsleitung; nähere Angaben siehe unten.

Antwort: {<DIOID>="<InputOn> LF}

wobei

<InputOn> den Status der digitalen Eingangsleitung angibt; nähere Angaben siehe unten.

Hinweise: Mit dem Befehl DIO? können die digitalen Eingangsleitungen 1 bis 4, die sich auf der Buchse **I/O** (S. 316) befinden, direkt gelesen werden.

Die für die Leitungen zu verwendenden Kennungen <DIOID> sind 1 bis 4. Wenn die Kennung weggelassen wird oder 0 ist, werden alle Leitungen abgefragt.

Wenn <InputOn>=0, ist das digitale Eingangssignal LOW/OFF, wenn <InputOn>=1, ist das digitale Eingangssignal HIGH/ON. Wenn <DIOID> 0 ist, ist <InputOn> ein Bitmuster, das den Status aller Leitungen im Hexadezimalformat angibt.

DRC (Set Data Recorder Configuration)

Beschreibung: Bestimmt für die angegebene Datenrekordertabelle die zu verwendende Datenquelle und die aufzunehmende Datenart (Aufzeichnungsoption).

Format: DRC {<RecTableID> <Source> <RecOption>}

Argumente: <RecTableID>: ist eine Datenrekordertabelle des Controllers, siehe unten.

<Source>: ist die Datenquelle, zum Beispiel eine Achse oder ein Kanal des Controllers. Die erforderliche Quelle ist von der ausgewählten Aufzeichnungsoption abhängig.

<RecOption>: bezeichnet die aufzuzeichnende Datenart (Aufzeichnungsoption).

Siehe unten für eine Liste der verfügbaren Aufzeichnungsoptionen und der entsprechenden Datenquellen.

Antwort: Keine

Hinweise: Der C-863 hat zwei Datenrekordertabellen mit 1024 Punkten pro Tabelle.

Mit HDR? (S. 203) erhalten Sie eine Liste aller verfügbaren Aufzeichnungs- und Triggeroptionen sowie zusätzliche Informationen über die Datenaufzeichnung. Die Anzahl verfügbarer Datenrekordertabellen kann mit TNR? (S. 252) gelesen werden.

Nähere Informationen siehe "Datenrekorder" (S. 101).

Verfügbare Aufzeichnungsoptionen mit den entsprechenden Datenquellen:

- 0=Nothing is recorded
- 1=Commanded position of axis
- 2=Actual position of axis
- 3=Position error of axis
- 70=Commanded velocity of axis
- 71=Commanded acceleration of axis
- 73=Motor output of axis
- 74=Kp of axis
- 75=Ki of axis

76=Kd of axis

80=Signal status register of axis

81=Analog input (channel = 1 - 9)

90=Active parameter set of axis

91=Motor current of axis

Hinweis: Die Eingangskanäle für die Aufzeichnungsoption 81 können die Leitungen Input 1 bis Input 4 der Buchse **I/O** (S. 316) sein (verwenden Sie für diese Datenquellen die Kennungen 1 bis 4).

Die Datenquellen-Kennungen 5 bis 7 bezeichnen die Eingänge für die Joystick-Achsen und den Joystick-Button:

5 = Achse 1 von Joystick-Gerät 1

6 = Taste 1 von Joystick-Gerät 1

7 = Achse 1 von Joystick-Gerät 2

Quellenkennungen 7 und höher sind für zusätzliche analoge Eingangskanäle reserviert.

DRC? (Get Data Recorder Configuration)

Beschreibung: Fragt die mit DRC (S. 189) vorgenommenen Einstellungen ab.

Format: DRC? [{<RecTableID>}]

Argumente: <RecTableID>: ist eine Datenrekordertabelle des Controllers; wird die Angabe weggelassen, enthält die Antwort die Einstellungen für alle Tabellen.

Antwort: Die aktuellen DRC-Einstellungen:

```
{<RecTableID>="<Source> <RecOption> LF}
```

wobei

<Source>: die Datenquelle ist, zum Beispiel, eine Achse oder ein Kanal des Controllers. Der Quelltyp ist von der Aufzeichnungsoption abhängig.

<RecOption>: bezeichnet die aufzuzeichnende Datenart (Aufzeichnungsoption).

Siehe DRC für eine Liste der verfügbaren Aufzeichnungsoptionen und der entsprechenden Datenquellen.

DRR? (Get Recorded Data Values)

Beschreibung: Fragt die zuletzt aufgezeichneten Daten ab.

In Abhängigkeit von der Anzahl der zu lesenden Punkte kann das Abfragen einige Zeit in Anspruch nehmen!

Es ist möglich, die Daten zu lesen, während die Aufzeichnung noch läuft.

Format: DRR? [<StartPoint> <NumberOfPoints> [{<RecTableID>}]]

Argumente: <StartPoint> ist der erste in der Datenrekordertabelle zu lesende Punkt, beginnt mit Index 1.

<NumberOfPoints> bezeichnet die Anzahl der je Tabelle zu lesenden Punkte.

<RecTableID> ist eine Datenrekordertabelle des Controllers.

Antwort: Die aufgezeichneten Daten im GCS-Array-Format siehe separates Handbuch für GCS Array, SM146E, und untenstehendes Beispiel.

Hinweise: Wenn <RecTableID> weggelassen wird, werden die Daten von allen Tabellen gelesen, deren Aufzeichnungsoption von Null verschieden ist.

Mit HDR? (S. 203) erhalten Sie eine Liste aller verfügbaren Aufzeichnungs- und Triggeroptionen sowie zusätzliche Informationen über die Datenaufzeichnung.

Weitere Informationen siehe die Beschreibung des Befehls DRC (S. 189) sowie "Datenrekorder" (S. 101).

Beispiel:

```
rtr?  
10  
drr? 1 20  
# REM C-863  
#  
# VERSION = 1  
# TYPE = 1  
# SEPARATOR = 32  
# DIM = 2  
# SAMPLE_TIME = 0.000500  
# NDATA = 20  
#  
# NAME0 = Actual Position of Axis  AXIS:1  
# NAME1 = Position Error of Axis  AXIS:1  
#  
# END_HEADER  
5.00000 0.00000  
4.99998 0.00002  
5.00000 0.00000  
5.00000 0.00000  
5.00000 0.00000  
5.00000 0.00000  
5.00000 0.00000  
4.99998 0.00002  
5.00000 0.00000  
4.99998 0.00002  
5.00000 0.00000  
5.00000 0.00000  
5.00000 0.00000  
5.00000 0.00000  
4.99998 0.00002  
5.00000 0.00000  
4.99998 0.00002  
4.99998 0.00002  
5.00000 0.00002  
4.99998 0.00004
```


DRT (Set Data Recorder Trigger Source)

Beschreibung:	Definiert eine Trigger-Quelle für die angegebene Datenrekordertabelle.
Format:	DRT <RecTableID> <TriggerSource> <Value>
Argumente:	<RecTableID> ist eine Datenrekordertabelle des Controllers. Nähere Angaben siehe unten. <TriggerSource> ID der Trigger-Quelle, Liste verfügbarer Optionen siehe unten. <Value> ist abhängig von der Trigger-Quelle, kann ein Dummy sein; siehe unten.
Antwort:	Keine
Hinweise:	Zurzeit ist nur 0 für <RecTableID> gültig; dies bedeutet, dass die angegebene Triggerquelle für alle Datenrekordertabellen gesetzt wird. Unabhängig von der eingestellten Triggeroption wird die Datenaufzeichnung immer ausgelöst, wenn eine Sprungantwortmessung mit STE (S. 246) durchgeführt wird. Eine mit DRT gesetzte Triggeroption gilt für alle Datenrekordertabellen, deren Aufzeichnungsoption von Null verschieden ist. Mit HDR? (S. 203) erhalten Sie eine Liste aller verfügbaren Aufzeichnungs- und Triggeroptionen sowie zusätzliche Informationen über die Datenaufzeichnung. Weitere Informationen siehe die Beschreibung des Befehls DRC (S. 189) sowie "Datenrekorder" (S. 101).
Verfügbare Triggeroptionen:	0 = default setting; Datenaufzeichnung wird mit STE ausgelöst; <Value> muss ein Dummy sein. 1 = any command changing target position (z. B. MVR (S. 227), MOV (S. 225)); <Value> muss ein Dummy sein.

2 = next command, setzt Trigger nach Ausführung zurück;
<Value> muss ein Dummy sein.

6 = any command changing target position (z. B. MVR,
MOV); setzt Trigger nach Durchführung zurück; <Value>
muss ein Dummy sein.

7 = SMO command, setzt Trigger nach Durchführung
zurück; <Value> muss ein Dummy sein.

DRT? (Get Data Recorder Trigger Source)

Beschreibung: Fragt die Triggerquelle für die Datenrekordertabellen ab.

Format: DRT? [{<RecTableID>}]

Argumente: <RecTableID> ist eine Datenrekordertabelle des
Controllers.

Antwort: {<RecTableID>="<TriggerSource> <Value> LF}

wobei

<TriggerSource> die ID der Triggerquelle ist.

<Value> ist abhängig von der Triggerquelle; 0 ist ein
Dummy.

Weitere Informationen finden Sie in der Beschreibung des
Befehls DRT (S. 193).

Hinweise: Da alle Datenrekordertabellen des C-863 dieselbe
Triggerquelle haben, wird die Antwort auf DRT? als
einzelne Zeile wie folgt angegeben

0=<TriggerSource> <Value>

ERR? (Get Error Number)

Beschreibung:	Fragt den Fehlercode <int> des zuletzt aufgetretenen Fehlers ab und setzt den Fehler auf 0 zurück. Es wird nur der letzte Fehler zwischengespeichert. Deshalb sollten Sie ERR? nach jedem Befehl aufrufen. Eine Auflistung der Fehlercodes und ihrer Beschreibungen ist unter "Fehlercodes" (S. 262) zu finden.
Format:	ERR?
Argumente:	Keine
Antwort:	Der Fehlercode des zuletzt aufgetretenen Fehlers (Integer).
Fehlersuche:	Kommunikationsstörung

FED (Find Edge)

Beschreibung:	Bewegt die angegebene Achse zu einer angegebenen Signalfanke. FED setzt keinen bestimmten Positionswert an der ausgewählten Flanke (im Gegensatz zu den Befehlen FNL (S. 197), FPL (S. 199) und FRF (S. 200) zur Referenzwertbestimmung), d. h. die Achse ist nach Verwendung von FED nicht "referenziert". Enthält der Befehl mehrere Achsen, werden sie synchron bewegt.
Format:	FED {<AxisID> <EdgeID> <Param>}
Argumente:	<AxisID> ist eine Achse des Controllers. <EdgeID> ist der Flankentyp, zu dem sich die Achse bewegen soll. Nähere Angaben zu vorhandenen Flankentypen siehe unten. <Param> ist von der ausgewählten Flanke abhängig und bestimmt sie näher. Nähere Angaben siehe unten.

Antwort: Keine

Fehlersuche: Unzulässige Achsenkennung; Endschalter und/oder Referenzschalter sind deaktiviert (siehe unten); Servomodus ist ausgeschaltet.

Hinweise: Der Servomodus muss mit SVO (S. 247) für die kommandierte Achse vor dem Einsatz dieses Befehls eingeschaltet sein (geregelter Betrieb).

Anhand von Parametern (ID 0x14 für Referenzschalter; ID 0x32 für Endschalter) ermittelt die Firmware des C-863 das Vorhandensein oder Fehlen von Referenz- und Endschaltern. Gemäß den Werten dieser Parameter aktiviert oder deaktiviert der C-863 FED-Bewegungen zu den entsprechenden Signalfanken. Passen Sie die Parameterwerte mit SPA (S. 240) oder SEP (S. 236) Ihrer Hardware entsprechend an. Weitere Informationen finden Sie in der Parameterübersicht (S. 288).

Sie können die digitalen Eingangsleitungen anstelle der Schalter als Quellen der Schaltersignale für FED verwenden. Weitere Informationen siehe "Digitale Eingangssignale" (S. 113).

FED kann verwendet werden, um den physikalischen Stellweg einer neuen Mechanik zu messen und so die Werte für die entsprechenden Parameter zu ermitteln: Den Abstand vom negativen zum positiven Endschalter, den Abstand zwischen dem negativen Endschalter und dem Referenzschalter (Parameter ID 0x17) und den Abstand zwischen Referenzschalter und positivem Endschalter (Parameter ID 0x2F). Weitere Informationen siehe "Stellweg und Verfahrbereichsgrenzen" (S. 35).

Die Bewegung kann durch #24 (S. 168), STP (S. 247) und HLT (S. 205) gestoppt werden.

Bewegungsbefehle wie FED sind nicht zulässig, wenn der Joystick für die Achse aktiv ist. Weitere Informationen siehe "Joystick-Steuerung" (S. 121).

Vorhandene Flankenarten und Parameter: Die folgenden Flankenarten und deren Parametereinstellungen sind vorhanden:

1 = negativer Endschalter, <Param> muss 0 sein
 2 = positiver Endschalter, <Param> muss 0 sein
 3 = Referenzschalter, <Param> muss 0 sein

FNL (Fast Reference Move To Negative Limit)

Beschreibung: Startet eine Referenzfahrt.

Bewegt die angegebene Achse zur negativen physikalischen Grenze ihres Stellwegs und setzt die aktuelle Position auf einen definierten Wert. Nähere Angaben siehe unten.

Enthält der Befehl mehrere Achsen, werden sie synchron bewegt.

Format: FNL [{<AxisID>}]

Argumente: <AxisID> ist eine Achse des Controllers; wenn die Angabe weggelassen wird, sind alle Achsen betroffen.

Antwort: Keine

Fehlersuche: Unzulässige Achsenkennung

Hinweise: Der Servomodus muss mit SVO (S. 247) für die kommandierte Achse vor dem Einsatz dieses Befehls eingeschaltet sein (geregelter Betrieb).
 Wenn die Referenzfahrt erfolgreich war, ist anschließend eine absolute Bewegung im geregelten Betrieb möglich.

Beim C-863 wird der negative Endschanter der Mechanik verwendet, um die negative physikalische Grenze des Stellwegs festzulegen. Die Differenz der Werte der Parameter 0x16 und 0x17 wird als aktuelle Position gesetzt, wenn die Achse am negativen Endschanter ist (Wert kann negativ sein).

Sie können ein digitales Eingangssignal anstelle des negativen Endschanter als Quelle des negativen Endschanter signals für FNL verwenden. Weitere Informationen siehe "Digitale Eingangssignale" (S. 113).

Die Bewegung kann durch #24 (S. 168), STP (S. 247) und HLT (S. 205) gestoppt werden.

Verwenden Sie FRF? (S. 201), um zu prüfen, ob die Referenzfahrt erfolgreich war.

Verwenden Sie FRF (S. 200) anstelle von FNL, um eine Referenzfahrt für eine Achse auszuführen, die keine Endschanter, sondern einen Referenzschanter hat.

Um die beste Wiederholgenauigkeit zu erreichen, führen Sie die Referenzwertbestimmung immer auf die gleiche Weise durch.

Wenn Verfahrbereichsgrenzen (Parameter 0x15 und 0x30) für die Verringerung des Stellwegs verwendet werden, können die Endschanter nicht für Referenzfahrten verwendet werden. Die Befehle FNL und FPL verursachen eine Fehlermeldung; nur der Referenzschanter kann für eine Referenzfahrt (FRF) verwendet werden.

Weitere Informationen siehe "Stellweg und Verfahrbereichsgrenzen" (S. 35).

FPL (Fast Reference Move To Positive Limit)

Beschreibung: Startet eine Referenzfahrt.

Bewegt die angegebene Achse zur positiven physikalischen Grenze ihres Stellwegs und setzt die aktuelle Position auf einen definierten Wert. Nähere Angaben siehe unten.

Enthält der Befehl mehrere Achsen, werden sie synchron bewegt.

Format: FPL [{<AxisID>}]

Argumente: <AxisID> ist eine Achse des Controllers; wenn die Angabe weggelassen wird, sind alle Achsen betroffen.

Antwort: Keine

Fehlersuche: Unzulässige Achsenkennung

Hinweise: Der Servomodus muss mit SVO (S. 247) für die kommandierte Achse vor dem Einsatz dieses Befehls eingeschaltet sein (geregelter Betrieb).
Wenn die Referenzfahrt erfolgreich war, ist anschließend eine absolute Bewegung im geregelten Betrieb möglich.
Beim C-863 wird der positive Endschalter der Mechanik verwendet, um die positive physikalische Grenze des Stellwegs zu bestimmen. Die Summe der Werte der Parameter 0x16 und 0x2F wird als aktuelle Position gesetzt, wenn die Achse am positiven Endschalter ist.

Sie können ein digitales Eingangssignal anstelle des positiven Endschalters als Quelle des positiven Endschaltersignals für FPL verwenden. Weitere Informationen siehe "Digitale Eingangssignale" (S. 113).

Die Bewegung kann durch #24 (S. 168), STP (S. 247) und HLT (S. 205) gestoppt werden.

Verwenden Sie FRF? (S. 201), um zu prüfen, ob die Referenzfahrt erfolgreich war.

Verwenden Sie FRF (S. 200) anstelle von FPL, um eine Referenzfahrt für eine Achse auszuführen, die keine

Endschalter, sondern einen Referenzschalter hat.

Um die beste Wiederholgenauigkeit zu erreichen, führen Sie die Referenzwertbestimmung immer auf die gleiche Weise durch.

Wenn Verfahrbereichsgrenzen (Parameter 0x15 und 0x30) für die Verringerung des Stellwegs verwendet werden, können die Endschalter nicht für Referenzfahrten verwendet werden. Die Befehle FNL und FPL verursachen eine Fehlermeldung; nur der Referenzschalter kann für eine Referenzfahrt (FRF) verwendet werden.

Weitere Informationen siehe "Stellweg und Verfahrbereichsgrenzen" (S. 35).

FRF (Fast Reference Move To Reference Switch)

Beschreibung:	Startet eine Referenzfahrt.
	Bewegt die angegebene Achse zum Referenzschalter und setzt die aktuelle Position auf einen definierten Wert. Nähere Angaben siehe unten.
	Enthält der Befehl mehrere Achsen, werden sie synchron bewegt.
Format:	FRF [{<AxisID>}]
Argumente:	<AxisID> ist eine Achse des Controllers; wenn die Angabe weggelassen wird, sind alle Achsen betroffen.
Antwort:	Keine
Fehlersuche:	Unzulässige Achsenkennung
Hinweise:	Der Servomodus muss mit SVO (S. 247) für die kommandierte Achse vor dem Einsatz dieses Befehls eingeschaltet sein (geregelter Betrieb). Wenn die Referenzfahrt erfolgreich war, ist anschließend eine absolute Bewegung im geregelten Betrieb möglich.

Der Wert des Parameters 0x16 wird als die aktuelle Position gesetzt, wenn die Achse am Referenzschalter ist.

Sie können ein digitales Eingangssignal anstelle des Referenzschalters als Quelle des Referenzsignals für den Befehl FRF verwenden. Weitere Informationen siehe "Digitale Eingangssignale" (S. 113).

Die Bewegung kann durch #24 (S. 168), STP (S. 247) und HLT (S. 205) gestoppt werden.

Verwenden Sie FRF? (S. 201), um zu prüfen, ob die Referenzfahrt erfolgreich war.

Verwenden Sie FNL (S. 197) oder FPL (S. 199) anstelle von FRF (S. 200), um eine Referenzfahrt für eine Achse auszuführen, die keinen Referenzschalter, sondern Endschalter hat.

Um die beste Wiederholgenauigkeit zu erreichen, führen Sie die Referenzwertbestimmung immer auf die gleiche Weise durch. Der Befehl FRF fährt den Referenzschalter immer von der gleichen Seite an, unabhängig davon, wo sich die Achse beim Befehlsaufruf befindet.

Weitere Informationen siehe "Stellweg und Verfahrbereichsgrenzen" (S. 35).

FRF? (Get Referencing Result)

Beschreibung:	Fragt ab, ob die angegebene Achse referenziert ist oder nicht.
Format:	FRF? [{<AxisID>}]
Argumente:	<AxisID>: ist eine Achse des Controllers.

Antwort:	{<AxisID>="<uint> LF}
	wobei
	<uint> angibt, ob die Achse erfolgreich referenziert wurde (=1) oder nicht (=0).
Fehlersuche:	Unzulässige Achsenkennung
Hinweise:	Eine Achse gilt als "referenziert", wenn der aktuelle Positionswert auf eine bekannte Position gesetzt ist. Dies ist der Fall, wenn eine Referenzfahrt erfolgreich durchgeführt wurde mit FNL (S. 197), FPL (S. 199) bzw. FRF (S. 200) oder wenn die Position direkt mit POS (S. 229) eingestellt wurde (abhängig von dem mit RON (S. 231) eingestellten Modus der Referenzwertbestimmung).

GOH (Go To Home Position)

Beschreibung:	Bewegt die angegebene Achse zur Nullposition.
	GOH [{<AxisID>}] ist identisch mit MOV {<AxisID> 0}
	Der Servomodus muss für die kommandierte Achse vor dem Einsatz dieses Befehls eingeschaltet sein (geregelter Betrieb).
	Die Bewegung kann durch #24 (S. 168), STP (S. 247) und HLT (S. 205) gestoppt werden.
Format:	GOH [{<AxisID>}]
Argumente:	<AxisID>: ist eine Achse des Controllers; wenn die Angabe weggelassen wird, sind alle Achsen betroffen.
Antwort:	Keine
Fehlersuche:	Unzulässige Achsenkennung

HDR? (Get All Data Recorder Options)

Beschreibung: Zeigt einen Hilfetext an, der alle verfügbaren Informationen zur Datenaufzeichnung enthält (Aufzeichnungsoptionen und Triggeroptionen, Information über zusätzliche Parameter und Befehle für die Datenaufzeichnung).

Format: HDR?

Argumente: Keine

Antwort #RecordOptions
{<RecOption>=" "<DescriptionString>[of <Channel>]}

#TriggerOptions
[<TriggerOption>=" "<DescriptionString>]

#Parameters to be set with SPA
[<ParameterID>=" "<DescriptionString>]

#Additional information
[<Command description>(" "<Command>")]

#Sources for Record Options
[<RecOption>=" "<Source>]

end of help

Beispiel: Für den C-863 lautet die Antwort auf HDR? wie folgt:

```
#RecordOptions
0=Nothing is recorded
1=Commanded Position of Axis
2=Actual Position of Axis
3=Position Error of Axis
70=Commanded Velocity of Axis
71=Commanded Acceleration of Axis
73=Motor Output of Axis
74=Kp of Axis
75=Ki of Axis
76=Kd of Axis
```

```

80=Signal Status Register of Axis
81=Analog input (Channel = 1 - 9)
90=active parameterset
91=Motorcurrent
#TriggerOptions
0=default setting
1=any command changing position (e.g. MOV)
2=next command
6=any command changing position (e.g. MOV),
reset trigger after execution
7=with SMO command, reset trigger after
execution
#Additional information
2 record tables
1024 datapoints per table
end of help

```

Hinweis: TriggerOptions = 0 (default) heißt, dass die Aufzeichnung durch den Befehl STE (S. 246) ausgelöst wird.

HLP? (Get List Of Available Commands)

Beschreibung:	Zeigt einen Hilfetext an, der alle verfügbaren Befehle enthält.
Format:	HLP?
Argumente:	Keine
Antwort:	Liste der verfügbaren Befehle
Fehlersuche:	Kommunikationsstörung

HLT (Halt Motion Smoothly)

Beschreibung:	Stoppt die Bewegung der angegebenen Achsen sanft. Nähere Angaben siehe Hinweise unten.
	Fehlercode 10 wird gesetzt.
	#24 (S. 168) und STP (S. 247) stoppen die aktuelle Bewegung hingegen so schnell wie für den Controller möglich, ohne Berücksichtigung von maximaler Geschwindigkeit und Beschleunigung.
Format:	HLT [{<AxisID>}]
Argumente:	<AxisID>: ist eine Achse des Controllers; wenn die Angabe weggelassen wird, werden alle Achsen angehalten.
Antwort:	Keine
Fehlersuche:	Unzulässige Achsenkennung
Hinweise:	HLT stoppt die Bewegung mit angegebener Systemabbremung im Hinblick auf die Systemträgheit.
	HLT stoppt jede Bewegung, die durch Bewegungsbefehle (z. B. MOV (S. 225), MVR (S. 227), GOH (S. 202), STE (S. 246), SMO (S. 238)), Befehle zur Referenzwertbestimmung (FNL (S. 197), FPL (S. 199), FRF (S. 200)) und Makros (MAC (S. 219)) verursacht wird.
	Nachdem die Achsen gestoppt wurden, werden ihre Zielpositionen auf ihre aktuellen Positionen gesetzt.

HPA? (Get List Of Available Parameters)

Beschreibung:	Antwortet mit einem Hilfe-String, der alle verfügbaren Parameter mit Kurzbeschreibungen enthält. Weitere Informationen siehe "Parameterübersicht" (S. 288).
Format:	HPA?
Argumente:	Keine

Antwort {<PamID>="<string> LF}

wobei

<PamID> die ID eines Parameters im Hexadezimalformat ist

<string> ein String ist, der den entsprechenden Parameter beschreibt.

Der String hat folgendes Format:

<CmdLevel>TAB<MaxItem>TAB<DataType>TAB<FunctionGroupDescription>TAB<ParameterDescription>[[TAB<PossibleValue>="<ValueDescription>]]

wobei

<CmdLevel> die Befehlsebene ist, die Schreibzugriff auf den Parameterwert erlaubt.

<MaxItem> ist die maximale Anzahl von Elementen des gleichen Typs, die von dem Parameter betroffen sind. Bei C-863 ist ein "Element" eine Achse.

<DataType> ist der Datentyp des Parameterwertes, er kann INT, FLOAT oder CHAR sein.

<FunctionGroupDescription> ist der Name der Funktionsgruppe, zu der der Parameter gehört.

<ParameterDescription> ist der Name des Parameters.

<PossibleValue> ist ein Wert aus dem zulässigen Datenbereich.

<ValueDescription> ist die Bedeutung des entsprechenden Wertes.

Die mit HPA? aufgelisteten Parameter können anhand der folgenden Befehle geändert und/oder gespeichert werden:

SPA (S. 240) beeinflusst die Parametereinstellungen im flüchtigen Speicher (RAM).

WPA (S. 260) kopiert Parametereinstellungen vom flüchtigen in den permanenten Speicher.

SEP (S. 236) schreibt die Parametereinstellungen direkt in den permanenten Speicher (ohne die Einstellungen im flüchtigen Speicher zu ändern).

RPA (S. 233) setzt den flüchtigen Speicher auf die Werte aus dem permanenten Speicher zurück.

JAS? (Query Joystick Axis Status)

Beschreibung: Fragt den aktuellen Status der angegebenen Achse des angegebenen Joystick-Gerätes ab, das an den Controller angeschlossen ist.

Format: JAS? [{<JoystickID> <JoystickAxis>}]

Argumente: <JoystickID> ist ein Joystick-Gerät, das an den Controller angeschlossen ist; nähere Angaben siehe unten.

<JoystickAxis> ist eine der Achsen des Joystick-Gerätes; nähere Angaben siehe unten.

Antwort: {<JoystickID> <JoystickAxis>="<Amplitude>}

wobei

<Amplitude> der Faktor ist, der gerade auf die aktuell gültige Geschwindigkeitseinstellung der angesteuerten Bewegungsachse angewendet wird; er entspricht der aktuellen Auslenkung der Joystick-Achse. Nähere Angaben siehe unten.

Hinweise: Zwei Joystick-Geräte können an die Buchse **Joystick** (S. 317) des C-863 angeschlossen werden, die Kennungen sind 1 und 2. Der C-863 unterstützt jeweils eine Achse pro Joystick-Gerät, die Kennung der Joystick-Achse ist jeweils 1. Weitere Informationen siehe "Kommandierbare Elemente" (S. 20).

Der Faktor <Amplitude> wird auf die mit VEL (S. 257) gesetzte Geschwindigkeit angewendet, der Bereich reicht von -1.0 bis 1.0. Beispiele: Bei einem Faktor von 0 ist die Joystick-Achse in Mittelstellung; bei Faktor -0.7 ist die Joystick-Achse um ca. 2/3 in negative Richtung ausgelenkt, vorausgesetzt, dass eine lineare Lookup-Tabelle aktuell gültig ist (ein Beispiel finden Sie in der Beschreibung des Befehls JLT (S. 211)).

JAX (Set Axis Controlled By Joystick)

Beschreibung: Bestimmt die Achse, die durch einen an den Controller angeschlossenen Joystick gesteuert wird.

Jede Achse des Controllers kann nur durch eine Joystick-Achse gesteuert werden.

Format: JAX <JoystickID> <JoystickAxis> <AxisID>

Argumente: <JoystickID> ist ein Joystick-Gerät, das an den Controller angeschlossen ist; nähere Angaben siehe unten.

<JoystickAxis> ist eine der Achsen des Joystick-Gerätes; nähere Angaben siehe unten.

<AxisID> ist eine Achse des Controllers.

Antwort: Keine

Hinweise: Zwei Joystick-Geräte können an die Buchse **Joystick** (S. 317) des C-863 angeschlossen werden, die Kennungen sind 1 und 2. Der C-863 unterstützt jeweils eine Achse pro Joystick-Gerät, die Kennung der Joystick-Achse ist jeweils 1. Weitere Informationen siehe "Kommandierbare Elemente" (S. 20).

JAX? (Get Axis Controlled By Joystick)

Beschreibung: Fragt die Achse ab, die durch einen an den Controller angeschlossenen Joystick gesteuert wird.

Format: JAX? [{<JoystickID> <JoystickAxis>}]

Argumente: <JoystickID> ist ein Joystick-Gerät, das an den Controller angeschlossen ist; nähere Angaben siehe unten.

<JoystickAxis> ist eine der Achsen des Joystick-Gerätes; nähere Angaben siehe unten.

Antwort: {<JoystickID> <JoystickAxis>="{<AxisID> }LF}

wobei

<AxisID> eine Achse des Controllers ist.

Hinweise: Zwei Joystick-Geräte können an die Buchse **Joystick** (S. 317) des C-863 angeschlossen werden, die Kennungen sind 1 und 2. Der C-863 unterstützt jeweils eine Achse pro Joystick-Gerät, die Kennung der Joystick-Achse ist jeweils 1. Weitere Informationen siehe "Kommandierbare Elemente" (S. 20).

JBS? (Query Joystick Button Status)

Beschreibung: Fragt den aktuellen Status der angegebenen Taste des angegebenen Joystick-Gerätes ab, das an den Controller angeschlossen ist.

Format: JBS? [{<JoystickID> <JoystickButton>}]

Argumente:	<JoystickID> ist ein Joystick-Gerät, das an den Controller angeschlossen ist; nähere Angaben siehe unten. <JoystickButton> ist eine der Tasten des Joystick-Gerätes; nähere Angaben siehe unten.
Antwort:	{<JoystickID> <JoystickButton> "="<State>} wobei <State> angibt, ob die Joystick-Taste gedrückt ist; 0 = nicht gedrückt, 1 = gedrückt.
Hinweise:	Zwei Joystick-Geräte können an die Buchse Joystick (S. 317) des C-863 angeschlossen werden, die Kennungen sind 1 und 2. Der C-863 unterstützt eine Taste des Joystick-Geräts 1, die Kennung der Joystick-Taste ist 1. Weitere Informationen siehe "Kommandierbare Elemente" (S. 20).

JDТ (Set Joystick Default Lookup Table)

Beschreibung:	Setzt den Typ der Lookup-Tabelle für die angegebene Achse des angegebenen Joystick-Gerätes, das an den Controller angeschlossen ist. Der aktuell gültige Inhalt der Lookup-Tabelle für die angegebene Joystick-Achse wird durch die mit JDТ getroffene Auswahl überschrieben.
Format:	JDТ {<JoystickID> <JoystickAxis> <uint>}
Argumente:	<JoystickID> ist ein Joystick-Gerät, das an den Controller angeschlossen ist; nähere Angaben siehe unten. <JoystickAxis> ist eine der Achsen des Joystick-Gerätes; nähere Angaben siehe unten. <uint> definiert den zu verwendenden Typ des Lookup-Tabellen-Profiles; nähere Angaben siehe unten.
Antwort:	Keine

Hinweise: Zwei Joystick-Geräte können an die Buchse **Joystick** (S. 317) des C-863 angeschlossen werden, die Kennungen sind 1 und 2. Der C-863 unterstützt jeweils eine Achse pro Joystick-Gerät, die Kennung der Joystick-Achse ist jeweils 1. Weitere Informationen siehe "Kommandierbare Elemente" (S. 20).

Beachten Sie, dass die Anzahl von Schreibzyklen im permanenten Speicher begrenzt ist. Ändern Sie den Typ der Lookup-Tabelle nur, wenn dies notwendig ist.

Vorhandene Lookup-Tabellen: Der C-863 stellt folgende Typen der Lookup-Tabellen-Profile zur Verfügung:

- 1 = linear (Standard)
- 2 = parabolisch

JLT (Fill Joystick Lookup Table)

Beschreibung: Füllt die Lookup-Tabelle für die angegebene Achse des angegebenen Joystick-Gerätes, das an den Controller angeschlossen ist.

Die Amplituden der Joystick-Achsen (d. h. deren Auslenkungen) werden auf die aktuell gültigen Geschwindigkeitseinstellungen der Controllerachsen abgebildet. Es gibt für jede Joystick-Achse eine Lookup-Tabelle, die diese Zuordnung festlegt. Diese Tabelle kann mit JLT geschrieben werden, oder es kann ein Standardtabellenprofil, das vom Controller zur Verfügung gestellt wird, mit dem Befehl JDT (S. 210) geladen werden.

Jede Lookup-Tabelle besteht aus 256 Punkten. Der erste Punkt entspricht standardmäßig der maximalen Auslenkung der Joystick-Achse in negativer Richtung, der 256. Punkt entspricht der maximalen Auslenkung in positiver Richtung.

Format: JLT <JoystickID> <JoystickAxis> <Addr> <floatn>

Argumente: <JoystickID> ist ein Joystick-Gerät, das an den Controller angeschlossen ist; nähere Angaben siehe unten.

<JoystickAxis> ist eine der Achsen des Joystick-Gerätes; nähere Angaben siehe unten.

<Addr> ist der Index eines Punktes in der Lookup-Tabelle, beginnt mit 1.

<floatn> ist der Wert von Punkt n.

Antwort: Keine

Hinweise: Zwei Joystick-Geräte können an die Buchse **Joystick** (S. 317) des C-863 angeschlossen werden, die Kennungen sind 1 und 2. Der C-863 unterstützt jeweils eine Achse pro Joystick-Gerät, die Kennung der Joystick-Achse ist jeweils 1. Weitere Informationen siehe "Kommandierbare Elemente" (S. 20).

Die Werte <floatn> sind Faktoren, die während der Joysticksteuerung auf die mit VEL (S. 257) gesetzte Geschwindigkeit angewendet werden, der Bereich reicht von -1.0000 bis 1.0000.

Die Werte <floatn> werden automatisch im permanenten Speicher des C-863 abgelegt.

Beispiel: In der aktuellen Lookup-Tabelle hat Punkt 1 den Wert -1, d. h. die angesteuerte Achse bewegt sich mit voller Geschwindigkeit in negativer Richtung bei maximaler negativer Auslenkung des Joysticks. Die Punkte 124 bis 133 haben den Wert 0, d. h. in Mittelstellung des Joysticks und in einem kleinen Bereich um den Mittelpunkt herum ist die Geschwindigkeit 0 und die angesteuerte Achse bewegt sich nicht. Punkt 236 hat den Wert 0.8369, d. h. wenn die Auslenkung der Joystick-Achse ca. 2/3 in positiver Richtung beträgt, bewegt sich die angesteuerte Achse in positiver Richtung mit ca. 4/5 der vollen Geschwindigkeit. Punkt 256 hat den Wert 1, d. h. die angesteuerte Achse bewegt sich mit voller Geschwindigkeit in positiver Richtung bei maximaler positiver Auslenkung des Joysticks.

Beachten Sie, dass die Anzahl von Schreibzyklen im permanenten Speicher begrenzt ist. Schreiben Sie Werte in die Lookup-Tabelle nur, wenn dies notwendig ist.

Weitere Informationen siehe "Joystick-Steuerung" (S. 121).

JLT? (Get Joystick Lookup Table Values)

Beschreibung: Fragt die aktuell gültigen Werte der Lookup-Tabelle ab.

Format: JLT? [<StartPoint> <NumberOfPoints> [{<JoystickID> <JoystickAxis>}]]

Argumente: <StartPoint> ist der Startpunkt in der Lookup-Tabelle, beginnt mit 1.

<NumberOfPoints> bezeichnet die Anzahl der je Joystick-Achse zu lesenden Punkte; die maximale Anzahl ist 256.

<JoystickID> ist ein Joystick-Gerät, das an den Controller angeschlossen ist; nähere Angaben siehe unten.

<JoystickAxis> ist eine der Achsen des Joystick-Gerätes; nähere Angaben siehe unten.

Antwort: Die Inhalte der Lookup-Tabellen im GCS-Array-Format, siehe separates Handbuch für GCS Array, SM 146E, und untenstehendes Beispiel.

Hinweise: Bei C-863 müssen <JoystickID> und <JoystickAxis> im Befehl JLT? weggelassen werden, während <StartPoint> und <NumberOfPoints> immer erforderlich sind.

Die Werte <floatn> in der Lookup-Tabelle sind Faktoren, die bei der Joystick-Steuerung auf die mit VEL (S. 257) gesetzte Geschwindigkeit angewendet werden, der Bereich reicht von -1.0000 bis 1.0000.

Beispiel:

```
jlt? 1 20
# TYPE = 1
#
# SEPARATOR = 32
# DIM = 1
# NDATA = 20
# NAME0 = Joysticktable 1
# END HEADER
-1.0000
-0.9922
-0.9834
-0.9756
-0.9678
-0.9590
-0.9512
-0.9434
-0.9346
-0.9268
-0.9189
-0.9102
-0.9023
-0.8945
-0.8857
-0.8779
-0.8701
-0.8613
-0.8535
-0.8457
```

JON (Set Joystick Activation Status)

Beschreibung: Aktiviert oder deaktiviert ein Joystick-Gerät, das an den Controller angeschlossen ist.

Format:	JON {<JoystickID> <uint>}
Argumente:	<JoystickID> ist ein Joystick-Gerät, das an den Controller angeschlossen ist; nähere Angaben siehe unten. <uint> 1 aktiviert das Joystick-Gerät, 0 deaktiviert das Joystick-Gerät.
Antwort:	Keine
Hinweise:	Zwei Joystick-Geräte können an die Buchse Joystick (S. 317) des C-863 angeschlossen werden, die Kennungen sind 1 und 2. Weitere Informationen siehe "Analogen Joystick anschließen" (S. 68). Bevor ein Joystick-Gerät mit JON aktiviert werden kann, müssen seine Achsen den Controllerachsen mit JAX (S. 208) zugewiesen werden. Wenn ein an den C-863 angeschlossener Joystick mit dem Befehl JON aktiviert ist, steuert dieser Joystick die Achsengeschwindigkeit (die vom Profilgenerator ausgegebene "kommandierte Geschwindigkeit"). Während der Joystick-Steuerung wird die Zielposition auf die Verfahrbereichsgrenze eingestellt, die durch den Parameter 0x15 bzw. 0x30 vorgegeben ist. Einzelheiten zu den Parametern finden Sie in "Stellweg und Verfahrbereichsgrenzen" (S. 35). Im unregelmäßigen Betrieb (Servomodus ausgeschaltet) ist keine Joysticksteuerung möglich. Beim Deaktivieren des Joysticks wird die Zielposition auf die aktuelle Position der joystickgesteuerten Achse eingestellt. Bewegungsbefehle wie MOV (S. 225) sind nicht zulässig, wenn ein Joystick für die Achse aktiv ist. Weitere Informationen siehe "Joystick-Steuerung" (S. 121).

JON? (Get Joystick Activation Status)

Beschreibung: Fragt den Status der Aktivierung des angegebenen Joystick-Geräts ab, das an den Controller angeschlossen ist.

Format: JON? [{<JoystickID>}]

Argumente: <JoystickID> ist ein Joystick-Gerät, das an den Controller angeschlossen ist; nähere Angaben siehe unten.

Antwort: {<JoystickID>="<uint>}

wobei

<uint> ist der Status der Joystickaktivierung: 1 = Joystick-Gerät aktiv, 0 = Joystick-Gerät deaktiviert.

Hinweise: Zwei Joystick-Geräte können an die Buchse **Joystick** (S. 317) des C-863 angeschlossen werden, die Kennungen sind 1 und 2. Weitere Informationen siehe "Analogen Joystick anschließen" (S. 68).

JRC (Jump Relatively Depending On Condition)

Beschreibung: Springt relativ, abhängig von einer angegebenen Bedingung des folgenden Typs: ein angegebener Wert wird mit einem abgefragten Wert gemäß einer angegebenen Regel verglichen.

Kann nur in Makros verwendet werden.

Format: JRC <Jump> <CMD?> <OP> <Value>

Argumente: <Jump> ist die Größe des relativen Sprungs. -1 bedeutet, dass der Makroausführungs-Zeiger zurück zur vorherigen Zeile springt, 0 bedeutet, dass der Befehl erneut ausgeführt wird, was dem Verhalten von WAC (S. 259) entspricht. 1 springt zur nächsten Zeile, was den Befehl überflüssig macht, 2 überspringt den nächsten Befehl. Es sind nur Sprünge innerhalb des aktuellen Makros zulässig.

<CMD?> ist ein Abfragebefehl in seiner üblichen Schreibweise. Die Antwort muss ein einzelner Wert sein (und nicht mehr). Beispiel siehe unten.

<OP> ist der zu verwendende Operator. Folgende Operatoren sind möglich:

= <= < > >= !=

Wichtig: Vor und nach dem Operator muss ein Leerzeichen stehen!

<Value> ist der Wert, der mit der Antwort auf <CMD?> zu vergleichen ist.

Antwort: Keine

Fehlersuche: Korrektes Sprungziel prüfen

Beispiel: Mit dem folgenden Makro können Sie die Bewegung der Achse 1 durch einen Stoppknopf anhalten, der an einem digitalen Eingang angeschlossen ist. Die Überprüfung des Stoppknopfes erfolgt so lange, bis die Achse die Zielposition erreicht hat (Abfrage ONT?). Wenn der Stoppknopf gedrückt wird, solange die Zielposition noch nicht erreicht ist: Das Ergebnis der Abfrage POS? 1 wird in die Variable TARGET kopiert. Diese Variable wird dann als zweites Argument für den Befehl MOV verwendet. Somit bleibt der Versteller, wo er gerade war. Zur Bereinigung wird TARGET mit dem Befehl VAR als leer definiert, wodurch die Variable gelöscht wird.

Schreiben Sie das Makro "stop":

```
MAC BEG stop
MOV 1 20
JRC 2 DI0? 1 = 1
JRC -1 ONT? 1 = 0
```

```

CPY TARGET POS? 1
MOV 1 ${TARGET}
VAR TARGET
MAC END

```

LIM? (Indicate Limit Switches)

Beschreibung: Fragt ab, ob die Achsen Endschalter haben.

Format: LIM? [{<AxisID>}]

Argumente: <AxisID>: ist eine Achse des Controllers.

Antwort: {<AxisID>="<uint> LF}

wobei

<uint> angibt, ob die Achse Endschalter hat (=1) oder nicht (=0).

Fehlersuche: Unzulässige Achsenkennung

Hinweise: Anhand eines Parameters (ID 0x32) ermittelt die Firmware des C-863 das Vorhandensein oder Fehlen von Endschaltern. Entsprechend dem Wert dieses Parameters aktiviert oder deaktiviert der C-863 das Stoppen der Bewegung an den Endschaltern und Referenzfahrten unter Verwendung von FNL (S. 197) oder FPL (S. 199).

Passen Sie den Parameterwert Ihrer Hardware entsprechend mit SPA (S. 240) oder SEP (S. 236) an. Weitere Informationen finden Sie in "Endschaltererkennung" (S. 34).

Sie können die digitalen Eingangsleitungen anstelle der Endschalter als Quelle der negativen oder positiven Endschaltersignale verwenden. Weitere Informationen finden Sie in "Digitale Eingangssignale" (S. 113).

MAC (Call Macro Function)

Beschreibung: Ruft eine Makrofunktion auf. Erlaubt das Aufzeichnen, Löschen und Ausführen von Makros auf dem Controller.

Format: MAC <keyword> {<parameter>}

insbesondere:

MAC BEG <macroname>

MAC DEF <macroname>

MAC DEF?

MAC DEL <macroname>

MAC END

MAC ERR?

MAC NSTART <macroname> <uint> [<String1>
[<String2>]]

MAC START <macroname> [<String1> [<String2>]]

Argumente: <keyword> legt fest, welche Makrofunktion aufgerufen wird. Die folgenden Schlüsselworte und Parameter werden verwendet:

MAC BEG <macroname>

Startet die Aufzeichnung eines Makros mit dem Namen *macroname* auf dem Controller; darf nicht in einem Makro verwendet werden; die Befehle, die folgen, bilden das Makro. Die Aufzeichnung wird mit MAC END beendet. Beachten Sie, dass fehlerhafter Makroinhalt nicht durch Senden des Befehls ERR? ermittelt werden kann.

MAC END

Stoppt die Makroaufzeichnung (kann nicht Bestandteil eines Makros werden).

MAC ERR?

Meldet den letzten Fehler, der während der Ausführung eines Makros auftrat.

Antwort: <macroname> <uint1>="<uint2> <"<"CMD">">

wobei <macroname> der Name des Makros ist, <uint1> ist die Zeile im Makro, <uint2> ist der Fehlercode, und <"<"CMD">"> ist der fehlerhafte Befehl, der an den Parser gesendet wurde.

MAC DEF <macroname>

Legt das angegebene Makro als Startup-Makro fest. Dieses Makro wird automatisch nach dem nächsten Einschalten oder Neustart des Controllers ausgeführt. Wird <macroname> weggelassen, wird die Auswahl des aktuellen Startup-Makros annulliert.

MAC DEF?

Fragt das Startup-Makro ab.

Antwort: <macroname>

Ist kein Startup-Makro festgelegt, ist die Antwort ein leerer String mit dem Abschlusszeichen.

MAC DEL <macroname>

Löscht das angegebene Makro.

MAC NSTART <macroname> <uint> [<String1> [<String2>]]

Wiederholt das angegebene Makro <uint> Mal. Eine neue Ausführung wird gestartet, wenn die letzte beendet ist.

<String1> und <String2> sind optionale Argumente, die die Werte der lokalen Variablen 1 und 2 angeben, die in dem angegebenen Makro verwendet werden. <String1> und <String2> können direkt oder über Variablenwerte angegeben werden. Die Makroausführung schlägt fehl, wenn das Makro lokale Variablen enthält, aber <String1> und <String2> im Befehl MAC NSTART weggelassen werden. Nähere Angaben siehe "Variablen" (S. 160).

MAC START <macroname> [<String1> [<String2>]]

Startet eine Ausführung des angegebenen Makros.

<String1> und <String2> haben die gleiche Funktion wie bei MAC NSTART.

Antwort: Keine

Fehlersuche: Makroaufzeichnung ist aktiv (Schlüsselworte BEG, DEL) oder inaktiv (END)

Makro enthält unzulässigerweise den Befehl MAC

Hinweise: Während einer Makroaufzeichnung ist keine Makroausführung erlaubt.

Wird ein Makro für einen Controller aufgezeichnet, dessen Adresse von 1 abweichend ist, muss die Empfängeradresse Bestandteil jeder Befehlszeile sein, sie wird jedoch nicht zum Bestandteil des Makroinhalts. PIMikroMove® sendet die Empfängeradresse bei der Makroaufzeichnung automatisch mit, so dass sie dort nicht eingegeben werden darf. Weitere Informationen siehe "Mit Makros arbeiten" (S. 133) und "Empfänger- und Senderadresse" (S. 158).

Bei der Aufzeichnung von Makros auf der Registerkarte **Controller macros** in PIMikroMove® müssen die Befehle **MAC BEG** und **MAC END** weggelassen werden.

Ein Makro kann von einem Makro mit demselben Namen überschrieben werden.

Makros können lokale und globale Variablen beinhalten. Weitere Informationen siehe "Variablen" (S. 160).

Ein laufendes Makro sendet keine Antworten an eine Schnittstelle.

In Abhängigkeit vom Wert des Parameters 0x72 (**Ignore Macro Error?**) bestehen die folgenden Möglichkeiten, wenn ein Fehler durch ein laufendes Makro verursacht wird:
0 = Die Makroausführung wird gestoppt (Standard).
1 = Der Fehler wird ignoriert und die Makroausführung wird fortgesetzt.

Unabhängig von der Parametereinstellung meldet MAC ERR? stets den letzten Fehler, der während einer Makroausführung aufgetreten ist.

Die folgenden vom C-863 zur Verfügung gestellten Befehle können nur in Makros verwendet werden:
DEL (S. 184), JRC (S. 216), MEX (S. 223) und WAC (S. 259).

Ein Makro kann ein anderes Makro starten. Die Höchstzahl der Verschachtelungsebenen beträgt 5. Ein Makro kann sich selbst aufrufen, um eine Endlosschleife zu bilden.

Von der Befehlszeile können sämtliche Befehle gesendet werden, während ein Makro läuft. Der Makroinhalt und Bewegungsbefehle, die von der Befehlszeile empfangen werden, können sich gegenseitig überschreiben.

Die Makroausführung kann durch #24 (S. 168) und STP (S. 247) gestoppt werden.

Zeitgleiche Ausführung mehrerer Makros ist nicht möglich. Es kann jeweils nur ein Makro ausgeführt werden.

Ein laufendes Makro kann nicht gelöscht werden.

Mit #8 (S. 168) können Sie abfragen, ob ein Makro aktuell auf dem Controller ausgeführt wird.

Beachten Sie, dass die Anzahl von Schreibzyklen im permanenten Speicher begrenzt ist. Zeichnen Sie Makros nur auf, wenn dies notwendig ist.

MAC? (List Macros)

Beschreibung:	Listet Makros oder den Inhalt eines angegebenen Makros auf.
Format:	MAC? [<macroname>]
Argumente	<macroname>: Name des Makros, dessen Inhalt aufgelistet werden soll; wird diese Angabe weggelassen, werden die Namen aller gespeicherten Makros aufgelistet.

Antwort: <string>

Wenn <macroname> angegeben wurde, ist <string> der Inhalt dieses Makros.

Wenn <macroname> weggelassen wurde, ist <string> eine Liste der Namen aller gespeicherten Makros.

Fehlersuche: Makro <macroname> nicht gefunden

MEX (Stop Macro Execution Due To Condition)

Beschreibung: Stoppt Makroausführung aufgrund einer angegebenen Bedingung des folgenden Typs: Ein angegebener Wert wird mit einem abgefragten Wert gemäß einer angegebenen Regel verglichen.

Kann nur in Makros verwendet werden.

Wenn der Makro-Interpreter auf diesen Befehl zugreift, wird die Bedingung geprüft. Ist sie erfüllt, wird das aktuelle Makro gestoppt, andernfalls wird die Makroausführung in der nächsten Zeile fortgesetzt. Sollte die Bedingung später erfüllt sein, wird der Interpreter sie ignorieren.

Siehe auch den Befehl WAC (S. 259).

Format: MEX <CMD?> <OP> <Value>

Argumente <CMD?> ist ein Abfragebefehl in seiner üblichen Schreibweise. Die Antwort muss ein einzelner Wert sein (und nicht mehr). Beispiel siehe unten.

<OP> ist der zu verwendende Operator. Folgende Operatoren sind möglich:

= <= < > >= !=

Wichtig: Vor und nach dem Operator muss ein Leerzeichen stehen!

<Value> ist der Wert, der mit der Antwort auf <CMD?> zu vergleichen ist.

Antwort: Keine

Beispiel: Senden: `MAC START LOOP`

Hinweis:

Makro LOOP beinhaltet Folgendes:

```
MAC START KEY1
```

```
MAC START KEY2
```

```
MEX DIO? 4 = 1
```

```
MAC START LOOP
```

Makro KEY1 beinhaltet Folgendes:

```
MEX DIO? 4 = 1
```

```
MEX DIO? 1 = 0
```

```
MVR 1 1.0
```

```
DEL 100
```

Makro KEY2 beinhaltet Folgendes:

```
MEX DIO? 4 = 1
```

```
MEX DIO? 2 = 0
```

```
MVR 1 -1.0
```

```
DEL 100
```

Makro LOOP bildet eine Endlosschleife, indem es permanent KEY1, KEY2 und sich selbst aufruft.

KEY1 prüft den Status des digitalen Eingangskanals 1 (befindet sich auf der Buchse **I/O** (S. 316)). Wird er nicht gesetzt (0), wird das Makro gestoppt, andernfalls bewegt das Makro Achse 1 um 1.0 in positive Richtung (relative Bewegung).

KEY2 prüft den Status des digitalen Eingangskanals 2 und bewegt Achse 1 entsprechend in negative Richtung.

Durch das Verbinden der digitalen Eingangsleitungen 1, 2 und 4 mit Drucktasten, z. B. mit der Pushbutton-Box C-170.PB, ist es möglich, die interaktive Ansteuerung einer

Achse ohne jegliche Softwareunterstützung zu realisieren. Die Verzögerung (DEL 100) ist erforderlich, um die Erzeugung mehrfacher Befehle MVR zu verhindern, wenn die Drucktaste kurz gedrückt wird.

Kanal 4 wird als globaler Ausstieg verwendet. Da MEX nur die Ausführung des aktuellen Makros stoppt, muss es auch in dem aufrufenden Makro enthalten sein, das andernfalls fortgesetzt würde.

MOV (Set Target Position)

Beschreibung: Setzt eine neue absolute Zielposition für die angegebene Achse.

Der Servomodus muss für die kommandierte Achse vor dem Einsatz dieses Befehls eingeschaltet werden (geregelter Betrieb).

Format: MOV {<AxisID> <Position>}

Argumente: <AxisID> ist eine Achse des Controllers.

<Position> ist die neue absolute Zielposition in physikalischen Einheiten.

Antwort: Keine

Hinweise: Die Zielposition muss sich innerhalb der Verfahrbereichsgrenzen befinden. Verwenden Sie TMN? (S. 251) und TMX? (S. 251), um die aktuell gültigen Verfahrbereichsgrenzen abzufragen.

Die Bewegung kann durch #24 (S. 168), STP (S. 247) und HLT (S. 205) gestoppt werden.

Während einer Bewegung setzt ein neuer Bewegungsbefehl das Ziel auf einen neuen Wert; der alte Wert wird eventuell niemals erreicht. Dies gilt auch für Makros: Bewegungsbefehle können von der Befehlszeile gesendet werden, wenn ein Makro ausgeführt wird. Der Makroinhalt und Bewegungsbefehle, die von der Befehlszeile empfangen werden, können sich gegenseitig

überschreiben.

Bewegungsbefehle wie MOV sind nicht zulässig, wenn ein Joystick für die Achse aktiv ist. Weitere Informationen siehe "Joystick-Steuerung" (S. 121).

- Beispiel 1: Senden: `MOV 1 10`
 Hinweis: Achse 1 bewegt sich nach 10 (Zielposition in mm)
- Beispiel 2: Senden: `MOV 1 243`
 Senden: `ERR?`
 Empfangen: `7`
 Hinweis: Die Achse bewegt sich nicht. Der Fehlercode "7" in der Antwort auf den Befehl ERR? (S. 195) gibt an, dass die in den Bewegungsbefehlen angegebene Zielposition außerhalb der Grenzwerte ist.

MOV? (Get Target Position)

Beschreibung: Fragt die letzte gültige kommandierte Zielposition ab.

Format: `MOV? [{<AxisID>}]`

Argumente: <AxisID> ist eine Achse des Controllers.

Antwort: `{<AxisID>="<float> LF}`

wobei

<float> die letzte kommandierte Zielposition in physikalischen Einheiten ist.

Fehlersuche: Unzulässige Achsenkennung

Hinweise: Die Zielposition kann durch Befehle, die Bewegung verursachen, geändert werden (z. B. MOV (S. 225), MVR (S. 227), GOH (S. 202), STE (S. 246)) oder durch den Joystick (bei Deaktivierung eines Joysticks wird für joystickgesteuerte Achsen im geregelten Betrieb die Zielposition auf die aktuelle Position gesetzt).

MOV? fragt die kommandierten Positionen ab. Verwenden Sie POS? (S. 230), um die aktuellen Positionen abzufragen.

MVR (Set Target Relative To Current Position)

Beschreibung: Bewegt die angegebene Achse relativ zur letzten kommandierten Zielposition.

Der Servomodus muss für die kommandierte Achse vor dem Einsatz dieses Befehls eingeschaltet sein (geregelter Betrieb).

Format: MVR {<AxisID> <Distance>}

Argumente: <AxisID> ist eine Achse des Controllers.

<Distance> gibt die Strecke an, um die sich die Achse bewegen soll; die Summe der Strecke und der letzten kommandierten Zielposition wird als neue Zielposition gesetzt (in physikalischen Einheiten).

Antwort: Keine

Hinweise: Die Zielposition muss sich innerhalb der Verfahrbereichsgrenzen befinden. Verwenden Sie TMN? (S. 251) und TMX? (S. 251), um die aktuell gültigen Verfahrbereichsgrenzen abzufragen, und MOV? (S. 226) für die Abfrage des aktuellen Ziels.

Die Bewegung kann durch #24 (S. 168), STP (S. 247) und HLT (S. 205) gestoppt werden.

Während einer Bewegung setzt ein neuer Bewegungsbefehl das Ziel auf einen neuen Wert; der alte Wert wird eventuell niemals erreicht. Dies gilt auch für

Makros: Von der Befehlszeile können Bewegungsbefehle gesendet werden, wenn ein Makro läuft. Der Makroinhalt und Bewegungsbefehle, die von der Befehlszeile empfangen werden, können sich gegenseitig überschreiben.

Bewegungsbefehle wie MVR sind nicht zulässig, wenn ein Joystick für die Achse aktiv ist. Weitere Informationen siehe "Joystick-Steuerung" (S. 121).

Beispiel:

Senden: `M0V 1 0.5`

Hinweis: Dies ist eine absolute Bewegung.

Senden: `POS? 1`

Empfangen: `1=0.500000`

Senden: `M0V? 1`

Empfangen: `1=0.500000`

Senden: `MVR 1 2`

Hinweis: Dies ist eine relative Bewegung.

Senden: `POS? 1`

Empfangen: `1=2.500000`

Senden: `MVR 1 2000`

Hinweis: Neue Zielposition von Achse 1 würde den Bewegungsbereich überschreiten. Befehl wird ignoriert, d. h. die Zielposition bleibt unverändert und die Achse bewegt sich nicht.

Senden: `M0V? 1`

Empfangen: `1=2.500000`

Senden: `POS? 1`

Empfangen: `1=2.500000`

ONT? (Get On-Target State)

Beschreibung: Fragt den On-Target-Status der angegebenen Achse ab.

Werden alle Argumente weggelassen, wird der Status aller Achsen abgefragt.

Format: `ONT? [{<AxisID>}]`

Argumente:	<AxisID> ist eine Achse des Controllers.
Antwort:	{<AxisID>="<uint> LF}
	wobei
	<uint> = "1" wenn die angegebene Achse an der Zielposition ist, anderenfalls "0".
Fehlersuche:	Unzulässige Achsenkennung
Hinweise:	Die Ermittlung des On-Target-Status ist nur im geregelten Betrieb möglich (Servomodus EIN).
	Der On-Target-Status wird von den Einstellungen für das Einschwingfenster (Parameter 0x36) und die Verzögerungszeit (Parameter 0x3F) beeinflusst. Details siehe "On-Target-Status" (S. 32).

POS (Set Real Position)

Beschreibung:	Setzt die aktuelle Position (löst keine Bewegung aus).
Format:	POS {<AxisID> <Position>}
Argumente:	<AxisID> ist eine Achse des Controllers.
	<Position> ist die neue aktuelle Position in physikalischen Einheiten.
Antwort:	Keine
Fehlersuche:	Unzulässige Achsenkennung
Hinweise:	Das Setzen der aktuellen Position mit POS ist nur möglich, wenn der Modus der Referenzwertbestimmung auf "0" gesetzt ist; siehe RON (S. 231).
	Eine Achse wird als "referenziert" bezeichnet, wenn die Position mit POS gesetzt wurde (weitere Informationen siehe "Referenzwertbestimmung" (S. 40)).

Die kleinsten und größten kommandierbaren Positionen (TMN? (S. 251), TMX? (S. 251)) werden nicht angepasst, wenn eine Position mit POS gesetzt wurde. Dies kann zu Zielpositionen führen, die vom C-863 zugelassen sind, aber von der Hardware nicht angefahren werden können. Ebenso sind Zielpositionen möglich, die von der Hardware angefahren werden können, aber vom C-863 verweigert werden. Darüber hinaus kann nach der Verwendung von POS die Nullposition außerhalb des physikalischen Stellwegs liegen.

POS? (Get Real Position)

Beschreibung: Fragt die aktuelle Achsenposition ab.

Werden alle Argumente weggelassen, wird die aktuelle Position aller Achsen abgefragt.

Format: POS? [{<AxisID>}]

Argumente: <AxisID> ist eine Achse des Controllers.

Antwort: {<AxisID>="<float> LF}

wobei

<float> die aktuelle Achsenposition in physikalischen Einheiten ist.

Fehlersuche: Unzulässige Achsenkennung

RBT (Reboot System)

Beschreibung: Startet das System neu. Der Controller verhält sich wie nach dem Einschalten.

Format: RBT

Argumente: Keine

Antwort: Keine

Hinweise: RBT kann nicht in Makros verwendet werden. Dadurch werden Probleme bei der Ausführung des Startup-Makros vermieden.

RMC? (List Running Macros)

Beschreibung: Listet die aktuell laufenden Makros auf.

Format: RMC?

Argumente: Keine

Antwort: {<macroname> LF}

wobei

<macroname> der Name eines Makros ist, das auf dem Controller gespeichert und aktuell ausgeführt wird. Die Antwort ist eine leere Zeile, wenn kein Makro ausgeführt wird.

RON (Set Reference Mode)

Beschreibung: Bestimmt den Modus der Referenzwertbestimmung der angegebenen Achsen.

Format: RON {<AxisID> <ReferenceOn>}

Argumente: <AxisID> ist eine Achse des Controllers.

<ReferenceOn> kann 0 oder 1 sein. 1 ist Standard. Details siehe unten.

Antwort: Keine

Fehlersuche: Unzulässige Achsenkennung

- Hinweise:
- <ReferenceOn> = 0: Für die Referenzwertbestimmung der Achse kann ein absoluter Positionswert mit POS (S. 229) zugewiesen werden, oder eine Referenzfahrt kann mit FRF (S. 200), FNL (S. 197) oder FPL (S. 199) gestartet werden. Relative Bewegungen mit MVR sind möglich, auch wenn der Referenzwert für die Achse noch nicht bestimmt wurde.
 - <ReferenceOn> = 1: Für die Referenzwertbestimmung der Achse muss eine Referenzfahrt mit FRF, FNL oder FPL gestartet werden. Die Verwendung von POS ist nicht zulässig. Bewegungen im geregelten Betrieb sind erst möglich, wenn der Referenzwert für die Achse bestimmt wurde.
 - Weitere Informationen siehe "Referenzwertbestimmung" (S. 40) und "Stellweg und Verfahrbereichsgrenzen" (S. 35).

RON? (Get Reference Mode)

- Beschreibung: Fragt den Modus der Referenzwertbestimmung der angegebenen Achsen ab.
- Format: RON? [{<AxisID>}]
- Argumente: <AxisID> ist eine Achse des Controllers.
- Antwort: {<AxisID>="<ReferenceOn> LF}
- wobei
- <ReferenceOn> der aktuell für die Achse gesetzte Modus der Referenzwertbestimmung ist
- Fehlersuche: Unzulässige Achsenkennung
- Hinweis: Weitere Informationen finden Sie in der Beschreibung des Befehls RON (S. 231).

RPA (Reset Volatile Memory Parameters)

Beschreibung: Setzt den angegebenen Parameter des angegebenen Elements zurück. Der Wert aus dem permanenten Speicher wird in den flüchtigen Speicher geschrieben.

Verwandte Befehle:

Mit HPA? (S. 205) erhalten Sie eine Liste der verfügbaren Parameter. SPA (S. 240) beeinflusst die Parametereinstellungen im flüchtigen Speicher, WPA (S. 260) schreibt Parametereinstellungen aus dem flüchtigen in den permanenten Speicher und SEP (S. 236) schreibt Parametereinstellungen direkt in den permanenten Speicher (ohne die Einstellungen im flüchtigen Speicher zu ändern).

Beispiel siehe SPA.

Format: RPA [{<ItemID> <PamID>}]

Argumente: <ItemID> ist das Element, für das ein Parameter zurückzusetzen ist. Nähere Angaben siehe unten.

<PamID> ist die Parameterkennung, kann im Hexadezimal- oder Dezimalformat geschrieben werden. Nähere Angaben siehe unten.

Antwort: Keine

Fehlersuche: Unzulässige Elementkennung, falsche Parameter-ID

Hinweise: Mit dem C-863 können Sie entweder alle Parameter oder einen einzelnen Parameter mit RPA zurücksetzen.

Verfügbare Element-IDs und Parameter-IDs: Ein Element ist eine Achse, die Kennung kann mit SAI (S. 235) geändert werden. Weitere Informationen siehe "Kommandierbare Elemente" (S. 20).

Gültige Parameter-IDs finden Sie in der Parameterübersicht (S. 288).

RTR (Set Record Table Rate)

Beschreibung: Setzt die Aufzeichnungsrate des Datenrekorders, d. h. die Anzahl der Zyklen, die für Datenaufzeichnungsvorgänge verwendet werden. Einstellungen größer als 1 ermöglichen es, längere Zeitspannen abzudecken.

Format: RTR <RecordTableRate>

Argumente: <RecordTableRate> ist die Aufzeichnungsrate des Datenrekorders, die für die Aufzeichnungsvorgänge zu verwenden ist (Einheit: Anzahl der Zyklen), muss ein ganzzahliger Wert größer als Null sein.

Antwort: Keine

Hinweise: Die Dauer der Aufzeichnung kann wie folgt berechnet werden:

$$\text{Aufz.Dauer} = \text{Zykluszeit des Regelkreises} * \text{RTR Wert} * \text{Anzahl der Punkte}$$

wobei

die Zykluszeit des Regelkreises für den C-863 50 µs beträgt

die Anzahl der Punkte für den C-863 1024 beträgt (Länge der Datenrekordertabelle)

Weitere Informationen siehe "Datenrekorder" (S. 101).

Die mit RTR gesetzte Aufzeichnungsrate des Datenrekorders wird nur im flüchtigen Speicher (RAM) gespeichert.

RTR? (Get Record Table Rate)

Beschreibung: Fragt die aktuelle Aufzeichnungsrate für die Datenrekordertabellen ab, d. h. die Anzahl der Zyklen, die für Datenaufzeichnungsvorgänge verwendet werden.

Format: RTR?

Argumente: Keine

Antwort: <RecordTableRate> ist die Rate, die für die Aufzeichnungsvorgänge verwendet wird (Einheit: Anzahl der Zyklen).

SAI (Set Current Axis Identifiers)

Beschreibung: Setzt die Achsenkennung für die angegebenen Achsen.

Nach dem Setzen der neuen Achsenkennung mit SAI ist sie als <AxisID> in allen achsenbezogenen Befehlen zu verwenden.

Format: SAI {<AxisID> <NewIdentifier>}

Argumente: <AxisID> ist eine Achse des Controllers.

<NewIdentifier> ist die neue für die Achse zu verwendende Kennung, nähere Angaben siehe unten.

Antwort: Keine

Hinweise: Eine Achse kann mit bis zu 8 Zeichen gekennzeichnet werden. Verwenden Sie TVI? (S. 254), um gültige Zeichen abzufragen.

Die neue Achsenkennung wird automatisch gespeichert und ist daher nach dem Neustart oder nächsten Einschalten wieder vorhanden.

SAI? (Get List Of Current Axis Identifiers)

Beschreibung: Fragt die Achsenkennung ab.

Siehe auch "Kommandierbare Elemente" (S. 20).

Format: SAI? [ALL]

Argumente: [ALL] ist optional. Bei Controllern, die Achsen-Deaktivierung zulassen, stellt [ALL] sicher, dass die Antwort auch Achsen enthält, die "deaktiviert" sind.

Antwort: {<AxisID> LF}

<AxisID> ist eine Achse des Controllers.

SEP (Set Non-Volatile Memory Parameters)

Beschreibung: Setzt einen Parameter des angegebenen Elements auf einen anderen Wert im permanenten Speicher, womit er zum neuen Standard wird.

Nach dem Setzen der Parameter mit SEP können Sie RPA (S. 233) verwenden, um sie ohne Neustart des Controllers zu aktivieren (in den flüchtigen Speicher zu schreiben).

Beachten Sie, dass dieser Befehl für die Einstellung hardwarespezifischer Parameter gilt. Falsche Werte können eventuell zu fehlerhaftem Betrieb oder zur Beschädigung Ihrer Hardware führen!

Verwandte Befehle:

HPA? (S. 205) gibt eine Liste verfügbarer Parameter zurück.

SPA (S. 240) schreibt Parametereinstellungen in den flüchtigen Speicher (ohne die Einstellungen im permanenten Speicher zu ändern).

WPA (S. 260) schreibt Parametereinstellungen vom flüchtigen in den permanenten Speicher.

Beispiel siehe SPA.

Format: SEP <Pswd> {<ItemID> <PamID> <PamValue>}

Argumente	<Pswd> ist das Passwort zum Schreiben in den permanenten Speicher, Standardwert ist "100". <ItemID> ist das Element, für das ein Parameter im permanenten Speicher zu ändern ist. Nähere Angaben siehe unten. <PamID> ist die Parameterkennung, kann im Hexadezimal- oder Dezimalformat geschrieben werden. Nähere Angaben siehe unten. <PamValue> ist der Wert, auf den der angegebene Parameter des angegebenen Elements gesetzt wird.
Antwort:	Keine
Fehlersuche:	Unzulässige Elementkennung, falsche Parameter-ID, ungültiges Kennwort
Hinweise:	Beachten Sie, dass die Anzahl von Schreibzyklen im permanenten Speicher begrenzt ist. Schreiben Sie Standardeinstellungen nur, wenn dies notwendig ist. Beim C-863 können Sie nur einen Parameter pro Befehl SEP schreiben.
Verfügbare Element-IDs und Parameter-IDs:	Ein Element ist eine Achse, die Kennung kann mit SAI (S. 235) geändert werden. Weitere Informationen siehe "Kommandierbare Elemente" (S. 20). Gültige Parameter-IDs finden Sie in der Parameterübersicht (S. 288).

SEP? (Get Non-Volatile Memory Parameters)

Beschreibung:	Frägt den Wert eines Parameters für ein angegebenes Element aus dem permanenten Speicher ab. Mit HPA? (S. 205) erhalten Sie eine Liste der verfügbaren Parameter und deren Kennungen (IDs).
Format:	SEP? [{<ItemID> <PamID>}]

Argumente:	<ItemID> ist das Element, für das ein Parameterwert im permanenten Speicher abgefragt werden soll. Nähere Angaben siehe unten. <PamID> ist die Parameterkennung, kann im Hexadezimal- oder Dezimalformat geschrieben werden. Nähere Angaben siehe unten.
Antwort:	{<ItemID> <PamID>="<PamValue> LF} wobei <PamValue> der Wert des angegebenen Parameters für das angegebene Element ist.
Fehlersuche:	Unzulässige Elementkennung, falsche Parameter-ID
Hinweise:	Mit dem C-863 können Sie entweder alle Parameter oder einen einzelnen Parameter pro Befehl SEP? abfragen.
Verfügbare Element-IDs und Parameter-IDs:	Ein Element ist eine Achse, die Kennung kann mit SAI (S. 235) geändert werden. Weitere Informationen siehe "Kommandierbare Elemente" (S. 20). Gültige Parameter-IDs finden Sie in der Parameterübersicht (S. 288).

SMO (Set Open-Loop Control Value)

Beschreibung:	Setzt den Stellwert direkt, um die Achse zu bewegen. Profilgenerator (wenn vorhanden), Sensor-Rückmeldung und Regelalgorithmus werden nicht berücksichtigt. Der Servomodus muss bei Verwendung dieses Befehls ausgeschaltet sein (ungeregelter Betrieb).
Format:	SMO {<AxisID> <ControlValue>}

Argumente	<AxisID> ist eine Achse des Controllers. <ControlValue> ist der neue Stellwert (dimensionslos). Nähere Angaben siehe unten.
Antwort:	Keine
Fehlersuche:	Unzulässige Achsenkennung
Hinweise:	Der Servomodus ist für eine der angegebenen Achsen eingeschaltet. HINWEIS: Bei großen Stellwerten kann der Versteller trotz Endschaltefunktion am mechanischen Anschlag aufprallen. Dies kann die Ausrüstung beschädigen. Der Stellwert ohne Vorzeichen darf nicht größer als der Wert des Parameters Maximum Motor Output (0x9) sein. Wird dieser Parameter auf seinen Höchstwert gesetzt (32767), reicht <ControlValue> von -32766 bis 32766 (dimensionslos). <ControlValue> steuert das PWM-Signal für die Achse an. Das Vorzeichen des Wertes bestimmt die Bewegungsrichtung: -32766 entspricht der maximalen Ausgangsspannung in negativer Bewegungsrichtung, und 32766 entspricht der maximalen Ausgangsspannung in positiver Bewegungsrichtung. Die Parameter Range Limit Min (0x07000000) und Range Limit Max (0x07000001) können als Verfahrbereichsgrenzen für Bewegungen im unregelmäßigen Betrieb mit SMO verwendet werden: Wenn die aktuelle Position diese Werte erreicht, wird der Stellwert auf null gesetzt und damit die Bewegung gestoppt. Sobald der Wert für die Verfahrbereichsgrenze verkleinert bzw. vergrößert wurde, kann die Achse wieder bewegt werden.
Beispiel:	Senden: <code>SMO 1 -16000</code> Hinweis: Der Stellwert beträgt ungefähr die Hälfte des maximalen Stellwerts. Die Achse bewegt sich in negativer Richtung.

SMO? (Get Control Value)

Beschreibung: Fragt den letzten gültigen Stellwert der angegebenen Achse ab.

Format: SMO? [{<AxisID>}]

Argumente <AxisID> ist eine Achse des Controllers.

Antwort: {<AxisID>="<float> LF}

wobei

<float> der letzte gültige Stellwert (dimensionslos) ist.
Nähere Angaben siehe unten.

Fehlersuche: Unzulässige Achsenkennung

Hinweise: Der Stellwert, der durch SMO? zurückgegeben wird, kann das Ergebnis des Regelalgorithmus und weiterer Korrekturen sein bzw. der Wert, der durch einen Befehl SMO (S. 238) im ungeregelten Betrieb gesetzt wird. Siehe das Blockdiagramm (S. 19) für weitere Informationen.

Der Stellwert reicht von -32766 bis 32766 (dimensionslos) und steuert das PWM-Signal für die Achse an. Das Vorzeichen des Wertes bestimmt die Bewegungsrichtung: -32766 entspricht der maximalen Ausgangsspannung in negativer Bewegungsrichtung und 32766 entspricht der maximalen Ausgangsspannung in positiver Bewegungsrichtung.

SPA (Set Volatile Memory Parameters)

Beschreibung: Setzt einen Parameter des angegebenen Elements im flüchtigen Speicher (RAM) auf einen bestimmten Wert. Parameteränderungen gehen verloren, wenn der Controller abgeschaltet oder neugestartet wird.

Format: SPA {<ItemID> <PamID> <PamValue>}

Argumente:	<ItemID> ist das Element, für das ein Parameter im flüchtigen Speicher geändert wird. Nähere Angaben siehe unten. <PamID> ist die Parameterkennung, kann im Hexadezimal- oder Dezimalformat geschrieben werden. Nähere Angaben siehe unten. <PamValue> ist der Wert, auf den der Parameter des angegebenen Elements gesetzt wird.
Antwort:	Keine Parameteränderungen gehen auch verloren, wenn die Parameter mit RPA (S. 233) wiederhergestellt werden. Beachten Sie, dass dieser Befehl für die Einstellung hardwarespezifischer Parameter gilt. Falsche Werte können eventuell zu fehlerhaftem Betrieb oder zur Beschädigung Ihrer Hardware führen!
	Verwandte Befehle: HPA? (S. 205) gibt eine Liste verfügbarer Parameter zurück. SEP (S. 236) schreibt Parametereinstellungen direkt in den permanenten Speicher (ohne die Einstellungen im flüchtigen Speicher zu ändern). WPA (S. 260) schreibt Parametereinstellungen vom flüchtigen in den permanenten Speicher. RPA setzt den flüchtigen Speicher auf den Wert aus dem permanenten Speicher zurück.
Fehlersuche:	Unzulässige Elementkennung, falsche Parameter-ID, Wert im unzulässigen Bereich
Hinweise:	Mit dem C-863 können Sie nur einen Parameter je Befehl SPA schreiben.

Verfügbare Element-IDs und Parameter-IDs:	Ein Element ist eine Achse, die Kennung kann mit SAI (S. 235) geändert werden. Weitere Informationen siehe "Kommandierbare Elemente" (S. 20). Gültige Parameter-IDs sind in der Parameterübersicht (S. 288) angegeben.
Beispiel 1:	Senden: <code>SPA 1 0x1 100</code> Hinweis: Setzt den P-Term des Regelalgorithmus für Achse 1 auf 100, die Parameter-ID wird im Hexadezimalformat geschrieben Senden: <code>SPA 1 1 150</code> Hinweis: Setzt den P-Term des Regelalgorithmus für Achse 1 auf 150, die Parameter-ID wird im Dezimalformat geschrieben
Beispiel 2:	Die P-, I- und D-Parameter des Regelalgorithmus müssen an eine neue Last, die auf die angeschlossene Mechanik einwirkt, angepasst werden. Senden: <code>SPA 1 0x1 150</code> Hinweis: Der P-Term wird für Achse 1 auf 150 gesetzt. Die Einstellung wird nur im flüchtigen Speicher vorgenommen. Setzen Sie mit SPA nun die I- und D-Terme im flüchtigen Speicher und prüfen anschließend die Funktion des Systems. Stellt sich die Leistung des geregelten Systems als zufriedenstellend heraus und möchten Sie die Systemkonfiguration als Standard verwenden, speichern Sie die Parametereinstellungen aus dem flüchtigen Speicher im permanenten Speicher. Senden: <code>WPA 100</code> Hinweis: Siehe die Befehlsbeschreibung für WPA (S. 260) für Details zum Umfang der gespeicherten Einstellungen.

Beispiel 3: Senden: SEP 100 LEFT 0xA 20

Hinweis: Die maximale geregelte Geschwindigkeit ist auf 20 mm/s für die Achse LEFT zu setzen (Achse wurde mit SAI umbenannt). Die Einstellung wird im permanenten Speicher vorgenommen und ist somit der neue Standard, der aber noch nicht aktiv ist. Um die neuen Einstellungen sofort zu verwenden, müssen Sie sie in den flüchtigen Speicher laden (anderenfalls würden sie aktiv werden, wenn der Controller das nächste Mal angeschaltet oder neugestartet wird).

Senden: RPA

Hinweis: Die neue Konfiguration ist nun aktiv.

Senden: SPA? LEFT 0xA

Empfangen: LEFT 0xA=20.00000

Hinweis: Prüfen Sie die Parametereinstellungen im flüchtigen Speicher.

SPA? (Get Volatile Memory Parameters)

Beschreibung: Fragt den Wert eines Parameters für ein angegebenes Element aus dem flüchtigen Speicher (RAM) ab.

Mit HPA? (S. 205) erhalten Sie eine Liste der verfügbaren Parameter.

Format: SPA? [{<ItemID> <PamID>}]

Argumente: <ItemID> ist das Element, für das ein Parameter im flüchtigen Speicher abgefragt werden soll. Nähere Angaben siehe unten.

<PamID> ist die Parameterkennung, kann im Hexadezimal- oder Dezimalformat geschrieben werden. Nähere Angaben siehe unten.

Antwort: {<ItemID> <PamID>="<PamValue> LF}

wobei

<PamValue> der Wert des angegebenen Parameters für das angegebene Element ist.

Fehlersuche: Unzulässige Elementkennung, falsche Parameterkennung

Hinweise: Mit dem C-863 können Sie entweder alle Parameter oder einen einzelnen Parameter je Befehl SPA? abfragen.

Verfügbare Element-IDs und Parameter-IDs: Ein Element ist eine Achse, die Kennung kann mit SAI (S. 235) geändert werden. Weitere Informationen siehe "Kommandierbare Elemente" (S. 20).

Gültige Parameter-IDs sind in der Parameterübersicht (S. 288) angegeben.

SRG? (Query Status Register Value)

Beschreibung: Gibt Registerwerte für die abgefragten Achsen und Register zurück.

Format: SRG? {<AxisID> <RegisterID>}

Argumente: <AxisID> ist eine Achse des Controllers.

<RegisterID> ist die ID des angegebenen Registers, verfügbare Register siehe unten.

Antwort: {<AxisID><RegisterID>="<Value> LF}

wobei

<Value> der Wert des Registers ist, nähere Angaben siehe unten.

Hinweis: Dieser Befehl ist funktionsgleich mit #4 (S. 166), der bevorzugt werden sollte, wenn der Controller zeitaufwändige Aufgaben ausführt.

Mögliche <RegisterID> kann 1 sein.

Register-IDs und

Antwortwerte: <Value> ist die bit-codierte Antwort und wird als Summe der folgenden einzelnen Codes in Hexadezimalformat zurückgegeben:

Bit	15	14	13	12	11	10	9	8
Beschreibung	On-Target-Status	Bestimmt den Referenzwert	In Bewegung	Servomodus Ein	-	-	-	Fehlerflag
Bit	7	6	5	4	3	2	1	0
Beschreibung	Digitale Eingangsleitung 4	Digitale Eingangsleitung 3	Digitale Eingangsleitung 2	Digitale Eingangsleitung 1	-	Pos. Endschalter	Referenzschalter	Neg. Endschalter

Beispiel: Senden: SRG? 1 1

Empfangen: 1 1=0x9002

Hinweis: Die Antwort wird im Hexadezimalformat angegeben. Sie besagt: Achse 1 ist an der Zielposition (On-Target-Status = wahr), der Servomodus für diese Achse ist eingeschaltet, es ist kein Fehler aufgetreten, der Status der digitalen Eingangsleitungen 1 bis 4 ist low, und Achse 1 befindet sich auf der positiven Seite des Referenzschalters.

STE (Start Step And Response Measurement)

Beschreibung:	Startet einen Sprung und die Aufzeichnung der Sprungantwort für die angegebene Achse. Die Datenrekorderkonfiguration, d. h. die Zuweisung der Datenquellen und der Aufzeichnungsoptionen zu den Rekordertabellen, kann mit DRC (S. 189) gesetzt werden. Die aufgezeichneten Daten können mit dem Befehl DRR? (S. 191) gelesen werden.
Format:	STE <AxisID> <Amplitude>
Argumente:	<AxisID> ist eine Achse des Controllers <Amplitude> ist die Größe des Sprungs. Nähere Angaben siehe unten.
Antwort:	Keine
Fehlersuche:	Der Servomodus muss für die kommandierte Achse vor dem Einsatz dieses Befehls eingeschaltet sein (geregelter Betrieb). Die Zielposition muss sich innerhalb der Verfahrbereichsgrenzen befinden. Verwenden Sie TMN? (S. 251) und TMX? (S. 251), um die aktuell gültigen Verfahrbereichsgrenzen abzufragen und MOV? (S. 226) für die Abfrage des aktuellen Ziels. Bewegungsbefehle wie STE sind nicht zulässig, wenn der Joystick für die Achse aktiv ist. Weitere Informationen siehe "Joystick-Steuerung" (S. 121).
Hinweise:	Ein "Sprung" besteht aus einer relativen Bewegung mit der angegebenen Amplitude, die relativ zur aktuellen Position ausgeführt wird.

STP (Stop All Axes)

Beschreibung: Stoppt alle Achsen abrupt. Nähere Angaben siehe Hinweise unten.

Setzt den Fehlercode auf 10.

Dieser Befehl ist funktionsgleich mit dem Befehl #24 (S. 168).

Format: STP

Argumente: Keine

Antwort: Keine

Fehlersuche: Kommunikationsstörung

Hinweise: STP stoppt jede Bewegung, die durch Bewegungsbefehle (z. B. MOV (S. 225), MVR (S. 227), GOH (S. 202), STE (S. 246), SMO (S. 238)), Befehle zur Referenzwertbestimmung (FNL (S. 197), FPL (S. 199), FRF (S. 200)) und Makros (MAC (S. 219)) verursacht wird. Stoppt auch die Makroausführung.

Nachdem die Achsen gestoppt wurden, werden ihre Zielpositionen auf ihre aktuellen Positionen gesetzt.

HLT (S. 205) stoppt im Gegensatz zu STP die Bewegung mit vorgegebener Abbremsung im Hinblick auf die Systemträgheit.

SVO (Set Servo Mode)

Beschreibung: Setzt den Servomodus für die angegebenen Achsen (ungeregelter oder geregelter Betrieb).

Format: SVO {<AxisID> <ServoState>}

Argumente: <AxisID> ist eine Achse des Controllers

<ServoState> kann folgende Werte haben:
 0 = Servomodus aus (ungeregelter Betrieb)
 1 = Servomodus ein (geregelter Betrieb)

Antwort:	Keine
Fehlersuche:	Unzulässige Achsenkennung
Hinweise:	Beim Wechsel vom unregelmäßigen Betrieb in den geregelten Betrieb wird die Zielposition auf die aktuelle Position gesetzt, um Sprünge der Mechanik zu vermeiden.

Der aktuelle Zustand des Servomodus bestimmt die anwendbaren Bewegungsbefehle:

Servomodus an: Verwenden Sie die Befehle MOV (S. 225), MVR (S. 227), GOH (S. 202) oder die Joystick-Steuerung (S. 121).

Servomodus aus: Verwenden Sie SMO (S. 238).

Der Servomodus muss eingeschaltet sein, bevor Referenzfahrten mit FRF (S. 200), FNL (S. 197) oder FPL (S. 199) gestartet werden können.

Wenn der Servomodus ausgeschaltet wird, während sich die Achse bewegt, stoppt die Achse.

Mit einem Startup-Makro kann der Controller so konfiguriert werden, dass der Servomodus nach dem Anschalten oder Neustart automatisch eingeschaltet ist. Nähere Angaben siehe "Startup-Makro einrichten" (S. 142).

Wenn die Achse über eine Bremse verfügt, beeinflusst das Setzen des Servomodus mit SVO den Aktivierungszustand der Bremse:

- Einschalten des Servomodus deaktiviert die Bremse.
- Ausschalten des Servomodus aktiviert die Bremse. Wenn der Servomodus ausgeschaltet ist, kann die Bremse mit BRA (S. 174) aktiviert oder deaktiviert werden. Sichern Sie den Versteller gegen unbeabsichtigte Bewegungen, bevor Sie die Bremse mit BRA deaktivieren!

Beim Auftreten eines Bewegungsfehlers wird der Servomodus ausgeschaltet, und die Bremse wird aktiviert. Mehr Informationen siehe "Bewegungsfehler" (S. 99).

SVO? (Get Servo Mode)

Beschreibung: Fragt den Servomodus für die angegebenen Achsen ab.

Werden keine Argumente angegeben, wird der Servomodus aller Achsen abgefragt.

Format: SVO? [{<AxisID>}]

Argumente: <AxisID> ist eine Achse des Controllers.

Antwort: {<AxisID>="<ServoState> LF}

wobei

<ServoState> der aktuelle Servomodus der Achse ist:

0 = Servomodus aus (ungeregelter Betrieb)

1 = Servomodus an (geregelter Betrieb)

Fehlersuche: Unzulässige Achsenkennung

TAC? (Tell Analog Channels)

Beschreibung: Fragt die Anzahl installierter Analogleitungen ab.

Format: TAC?

Argumente: Keine

Antwort: <uint> gibt die Gesamtanzahl der Analogleitungen (Ein- und Ausgänge) an.

Hinweise: Fragt die Anzahl der analogen Eingangsleitungen auf der Buchse **I/O** (S. 316) des C-863 ab (Input 1 bis Input 4). Beachten Sie, dass diese Leitungen auch für digitale Eingangssignale verwendet werden können. Weitere Informationen siehe "Kommandierbare Elemente" (S. 20).

TAV? (Get Analog Input Voltage)

Beschreibung: Fragt die Spannung am Analogeingang ab.

Format: TAV? [{<AnalogInputID>}]

Argumente:	<AnalogInputID> ist die Kennung des analogen Eingangskanals; nähere Angaben siehe unten.
Antwort:	{<AnalogInputID>="<float> LF}
	wobei
	<float> die aktuelle Spannung am Analogeingang ist, in Volt
Hinweise:	Mit dem Befehl TAV? können die Leitungen Input 1 bis Input 4 auf der Buchse I/O (S. 316) des C-863 direkt gelesen werden. Die Kennungen der Leitungen sind 1 bis 4. Weitere Informationen siehe "Kommandierbare Elemente" (S. 20).
	Die Werte der analogen Eingangsleitungen können mit DRC-Aufzeichnungsoption 81 (S. 189) aufgezeichnet werden.

TIO? (Tell Digital I/O Lines)

Beschreibung:	Gibt die Anzahl der installierten digitalen I/O-Leitungen an.
Format:	TIO?
Argumente:	Keine
Antwort:	I=<uint1> O=<uint2>
	wobei
	<uint1> die Anzahl der digitalen Eingangsleitungen ist. <uint2> die Anzahl der digitalen Ausgangsleitungen ist.

Hinweise: Die durch TIO? gemeldeten digitalen Ausgangsleitungen sind Output 1 bis Output 4. Der Status der Leitungen Output 1 bis Output 4 kann durch Verwendung des Befehls DIO (S. 187) gesetzt werden. Darüber hinaus können Sie die Leitungen Output 1 bis Output 4 mit dem Befehl CTO (S. 177) (Triggerkonfiguration) und dem Befehl TRO (S. 252) (Triggeraktivierung/-deaktivierung) programmieren.

Die durch TIO? gemeldeten digitalen Eingangsleitungen sind Input 1 bis Input 4. Sie können mit DIO? (S. 188), #4 (S. 166) und SRG? (S. 244) gelesen werden.

Alle Leitungen befinden sich auf der Buchse I/O (S. 316) des C-863.

TMN? (Get Minimum Commandable Position)

Beschreibung: Fragt die kleinste kommandierbare Position in physikalischen Einheiten ab.

Format: TMN? [{<AxisID>}]

Argumente: <AxisID> ist eine Achse des Controllers.

Antwort {<AxisID>="<float> LF}

wobei

<float> die kleinste kommandierbare Position in physikalischen Einheiten ist.

Hinweis: Die kleinste kommandierbare Position wird durch den Parameter 0x30 bestimmt.

TMX? (Get Maximum Commandable Position)

Beschreibung: Fragt die größte kommandierbare Position in physikalischen Einheiten ab.

Format: TMX? [{<AxisID>}]

Argumente:	<AxisID> ist eine Achse des Controllers.
Antwort	{<AxisID>="<float> LF}
	wobei
	<float> die größte kommandierbare Position in physikalischen Einheiten ist.
Hinweis:	Die größte kommandierbare Position wird durch den Parameter 0x15 bestimmt.

TNR? (Get Number of Record Tables)

Beschreibung:	Fragt die Anzahl der aktuell auf dem Controller verfügbaren Datenrekordertabellen ab.
Format:	TNR?
Argumente:	Keine
Antwort	<uint> ist die Anzahl der Datenrekordertabellen, die aktuell auf dem Controller verfügbar sind.
Hinweise:	Der C-863 hat zwei Datenrekordertabellen mit 1024 Datenpunkten pro Tabelle.

Weitere Informationen siehe "Datenrekorder" (S. 101).

TRO (Set Trigger Output State)

Beschreibung:	Aktiviert oder deaktiviert die mit CTO (S. 177) gesetzten Bedingungen der Triggerausgabe für die angegebene digitale Ausgangsleitung.
Format:	TRO {<TrigOutID> <TrigMode>}
Argumente:	<TrigOutID> ist eine digitale Ausgangsleitung des Controllers; weitere Angaben siehe unten.
	<TrigMode> kann folgende Werte annehmen: 0 = Triggerausgabe deaktiviert 1 = Triggerausgabe aktiviert

Antwort: Keine

Fehlersuche: Unzulässige Kennung der digitalen Ausgangsleitung

Hinweise: <TrigOutID> entspricht den digitalen Ausgangsleitungen Output 1 bis Output 4, IDs = 1 bis 4; weitere Informationen siehe "I/O" (S. 316).

Verwenden Sie DIO (S. 187) nicht für digitale Ausgangsleitungen, bei denen die Triggerausgabe mit TRO aktiviert ist.

TRO? (Get Trigger Output State)

Beschreibung: Fragt für die angegebene digitale Ausgangsleitung den Aktivierungsstatus der mit CTO (S. 177) vorgenommenen Triggerausgabe-Konfiguration ab.

Werden alle Argumente weggelassen, wird der Status aller digitalen Ausgangsleitungen abgefragt.

Format: TRO? [{<TrigOutID>}]

Argumente: <TrigOutID> ist eine digitale Ausgangsleitung des Controllers; weitere Einzelheiten finden Sie in der Beschreibung des Befehls TRO (S. 252).

Antwort: {<TrigOutID>="<TrigMode> LF}

wobei

<TrigMode> der aktuelle Status der digitalen Ausgangsleitung ist:
 0 = Triggerausgabe deaktiviert
 1 = Triggerausgabe aktiviert

Fehlersuche: Unzulässige Kennung der digitalen Ausgangsleitung

TRS? (Indicate Reference Switch)

Beschreibung: Zeigt an, ob die Achsen einen Referenzschalter mit Richtungserkennung haben.

Format:	TRS? [{<AxisID>}]
Argumente:	<AxisID> ist eine Achse des Controllers
Antwort:	{<AxisID>="<uint> LF}
	wobei
	<uint> angibt, ob die Achse einen richtungserkennenden Referenzschalter hat (=1) oder nicht (=0).
Fehlersuche:	Unzulässige Achsenkennung
Hinweise:	Anhand eines Parameters (ID 0x14) ermittelt die Firmware des C-863 das Vorhandensein oder Fehlen eines Referenzschalters. Gemäß dem Wert dieses Parameters aktiviert oder deaktiviert der C-863 Referenzfahrten zum Referenzschalter (Befehl FRF (S. 200)). Passen Sie den Parameterwert entsprechend Ihrer Hardware mit SPA (S. 240) oder SEP (S. 236) an. Weitere Informationen siehe "Referenzschalterkennung" (S. 33). Sie können eine digitale Eingangsleitung anstelle des Referenzschalters als Quelle des Referenzsignals für den Befehl FRF verwenden. Weitere Informationen siehe "Digitale Eingangssignale" (S. 113).

TVI? (Tell Valid Character Set For Axis Identifiers)

Beschreibung:	Gibt einen String mit den für Achsenkennungen verwendbaren Zeichen zurück. Verwenden Sie SAI (S. 235), um die Achsenkennungen zu ändern und SAI? (S. 235), um die aktuell gültigen Achsenkennungen abzufragen.
Format:	TVI?
Argumente:	Keine
Antwort:	<string> ist eine Liste von Zeichen.

Hinweise: Beim C-863 besteht der String aus
1234567890ABCDEFGHIJKLMNOPQRSTUVWXYZ_

VAR (Set Variable Value)

Beschreibung: Setzt eine Variable auf einen bestimmten Wert.

Lokale Variablen können mit VAR nur in Makros gesetzt werden. Nähere Angaben zu lokalen und globalen Variablen siehe "Variablen" (S. 160).

Die Variable ist nur im RAM vorhanden.

Format: VAR <Variable> <String>

Argumente: <Variable> ist der Name der Variablen, deren Wert gesetzt werden soll.

<String> ist der Wert, auf den die Variable zu setzen ist. Wird er weggelassen, wird die Variable gelöscht.

Der Wert kann direkt oder über den Wert einer Variablen angegeben werden.

Nähere Angaben zu Konventionen bezüglich Namen und Werten von Variablen siehe "Variablen" (S. 160).

Antwort: Keine

Beispiel: Es ist möglich, den Wert einer Variablen (z. B. TARGET) auf den einer anderen Variablen (z. B. SOURCE) zu setzen:

```
VAR TARGET ${SOURCE}
```

Verwenden Sie geschweifte Klammern, wenn der Name der Variablen länger als ein Zeichen ist:

```
VAR A ONE
```

```
VAR VARB TWO
```

```
VAR $A 1
```

```

VAR ${VARB} 2
VAR $VARB 2 // dies führt zu unerwünschtem Verhalten
VAR?
A=ONE
VARB=TWO
ONE=1
TWO=2 // ${VARB}: wird durch ihren Wert "TWO"
ersetzt.
ARB=2 // $VARB: $V wird durch ihren (leeren) Wert
ersetzt.

```

Ein weiteres Beispiel finden Sie in der Beschreibung des Befehls ADD (S. 171).

VAR? (Get Variable Values)

Beschreibung: Gibt Variablenwerte zurück.

Wird VAR? mit CPY (S. 175), JRC (S. 216), MEX (S. 223) oder WAC (S. 259) kombiniert, muss die Antwort auf VAR? ein einzelner Wert sein (und nicht mehr).

Nähere Angaben zu lokalen und globalen Variablen siehe "Variablen" (S. 160).

Format: VAR? [{<Variable>}]

Argumente: <Variable> ist der Name der abzufragenden Variablen. Nähere Angaben zu Namenskonventionen siehe "Variablen" (S. 160).

Wird <Variable> weggelassen, werden alle im RAM vorhandenen globalen Variablen aufgelistet.

Antwort: {<Variable>="<String>LF}

wobei

<String> den Wert angibt, auf den die Variable gesetzt ist.

- Hinweise: Lokale Variablen können mit VAR? nur abgefragt werden, wenn ein Makro mit lokalen Variablen läuft. Nähere Angaben zu lokalen und globalen Variablen siehe "Variablen" (S. 160).
- Beispiel: Beispiel siehe ADD (S. 171).

VEL (Set Closed-Loop Velocity)

- Beschreibung: Setzt die Geschwindigkeit für die angegebenen Achsen.
- Die Geschwindigkeit kann mit VEL verändert werden, während die Achse sich bewegt.
- Format: VEL {<AxisID> <Velocity>}
- Argumente: <AxisID> ist eine Achse des Controllers.
- <Velocity> ist der Geschwindigkeitswert in physikalischen Einheiten pro Sekunde.
- Antwort: Keine
- Fehlersuche: Unzulässige Achsenkennungen
- Hinweise: Die VEL-Einstellung wird nur wirksam, wenn sich die angegebene Achse im geregelten Betrieb befindet (Servomodus ein).
- Der kleinstmögliche Wert für <Velocity> ist 0.
- VEL ändert den Wert des Parameters **Closed-Loop Velocity (Phys. Unit/s)** (ID 0x49) im flüchtigen Speicher (kann mit WPA als Standard gespeichert werden, kann auch mit SPA (S. 240) und SEP (S. 236) geändert werden).
- Der maximale Wert, der mit dem Befehl VEL gesetzt werden kann, wird durch den Parameter **Maximum Closed-Loop Velocity (Phys. Unit/s)**, ID 0xA, vorgegeben (kann mit SPA (S. 240) und SEP (S. 236) geändert werden).

VEL? (Get Closed-Loop Velocity)

Beschreibung:	Fragt den mit VEL (S. 257) kommandierten Geschwindigkeitswert ab.
	Werden keine Argumente angegeben, wird der mit VEL kommandierte Wert aller Achsen abgefragt.
Format:	VEL? [{<AxisID>}]
Argumente:	<AxisID> ist eine Achse des Controllers.
Antwort:	{<AxisID>="<float> LF}
	wobei
	<float> der mit VEL kommandierte Geschwindigkeitswert in physikalischen Einheiten pro Sekunde ist.
Hinweise:	VEL? fragt den Wert der Geschwindigkeit für den geregelten Betrieb ab.

VER? (Get Versions Of Firmware And Drivers)

Beschreibung:	Fragt die Versionen der Firmware des C-863 und weiterer Komponenten wie z.B. Treiber und Bibliotheken ab.
Format:	VER?
Argumente:	Keine
Antwort	{<string1>":"<string2> [<string3>]LF}
	wobei
	<string1> der Name der Komponente ist;
	<string2> die Versionsinformation der Komponente
	<string1>ist;
	<string3> eine optionale Angabe ist.

WAC (Wait For Condition)

Beschreibung:	Wartet, bis eine angegebene Bedingung des folgenden Typs auftritt: ein angegebener Wert wird mit einem abgefragten Wert gemäß einer angegebenen Regel verglichen. Kann nur in Makros verwendet werden. Siehe auch den Befehl MEX (S. 223).
Format:	WAC <CMD?> <OP> <Value>
Argumente	<CMD?> ist ein Abfragebefehl in seiner üblichen Schreibweise. Die Antwort muss ein einzelner Wert sein (und nicht mehr). Beispiel siehe unten. <OP> ist der zu verwendende Operator. Folgende Operatoren sind möglich: = <= < > >= != Wichtig: Vor und nach dem Operator muss ein Leerzeichen stehen! <Value> ist der Wert, der mit der Antwort auf <CMD?> zu vergleichen ist.
Antwort:	Keine
Beispiel:	Senden: <pre>MAC BEG LPMOTION MVR 1 1 WAC ONT? 1 = 1 MVR 1 -1 WAC ONT? 1 = 1 MAC START LPMOTION MAC END MAC START LPMOTION</pre> Hinweis: Makro LPMOTION wird erst aufgezeichnet, dann gestartet. WAC ONT? 1 = 1 wartet, bis die Antwort auf ONT? 1 1=1 ist. Das Makro ruft sich selbst auf, um eine Endlosschleife zu bilden.

WPA (Save Parameters To Non-Volatile Memory)

Beschreibung: Schreibt den aktuell gültigen Wert eines Parameters für ein angegebenes Element aus dem flüchtigen Speicher (RAM) in den permanenten Speicher. Die auf diese Art gespeicherten Werte werden die Standardwerte.

Hinweis: Sind die aktuellen Parameterwerte falsch, kann dies zu einer Störung des Systems führen. Vergewissern Sie sich, dass die Parametereinstellungen korrekt sind, bevor Sie den Befehl WPA ausführen.

RAM-Einstellungen, die nicht mit WPA gespeichert wurden, gehen verloren, wenn der Controller ausgeschaltet oder neugestartet wird bzw. wenn die Parameter mit RPA (S. 233) wiederhergestellt werden.

Mit HPA? (S. 205) erhalten Sie eine Liste aller verfügbaren Parameter.

Prüfen Sie die aktuellen Parametereinstellungen im flüchtigen Speicher mit SPA? (S. 240).

Ein Beispiel finden Sie in der Beschreibung des Befehls SPA (S. 240).

Format: WPA <Pswd> [{<ItemID> <PamID>}]

Argumente: <Pswd> ist das Kennwort zum Schreiben in den permanenten Speicher. Nähere Angaben siehe unten.

<ItemID> ist das Element, für das ein Parameter aus dem flüchtigen Speicher im permanenten Speicher gespeichert werden soll. Nähere Angaben siehe unten.

<PamID> ist die Parameterkennung, kann im Hexadezimal- oder Dezimalformat geschrieben werden. Nähere Angaben siehe unten.

Antwort: Keine

Fehlersuche:	Unzulässige Elementkennung, falsche Parameter-ID, ungültiges Kennwort
Hinweise:	Die Parameter können im flüchtigen Speicher mit SPA (S. 240), ACC (S. 169), DEC (S. 183) und VEL (S. 257) geändert werden. Wird WPA ohne Spezifizierung eines Arguments verwendet, mit Ausnahme des Kennworts, werden die aktuell gültigen Werte aller Parameter, die von dem angegebenen Kennwort betroffen sind, gespeichert. Anderenfalls kann nur ein einzelner Parameter pro Befehl WPA gespeichert werden. Beachten Sie, dass die Anzahl von Schreibzyklen im permanenten Speicher begrenzt ist. Schreiben Sie Standardeinstellungen nur, wenn dies notwendig ist.
Gültige Kennwörter:	Das Kennwort zum Schreiben in den permanenten Speicher ist "100".
Verfügbare Element-IDs und Parameter-IDs:	Ein Element ist eine Achse, die Kennung kann mit SAI (S. 235) geändert werden. Weitere Informationen siehe "Kommandierbare Elemente" (S. 20). Gültige Parameter-IDs finden Sie in der Parameterübersicht (S. 288).

9.7 Fehlercodes

Die hier aufgelisteten Fehlercodes sind Bestandteil des PI General Command Set. Einige der Fehlercodes sind für Ihren Controller möglicherweise nicht relevant und werden daher nie ausgegeben.

Controllerfehler

0	PI_CNTR_NO_ERROR	No error
1	PI_CNTR_PARAM_SYNTAX	Parameter syntax error
2	PI_CNTR_UNKNOWN_COMMAND	Unknown command
3	PI_CNTR_COMMAND_TOO_LONG	Command length out of limits or command buffer overrun
4	PI_CNTR_SCAN_ERROR	Error while scanning
5	PI_CNTR_MOVE_WITHOUT_REF_OR_NO_SERVO	Unallowable move attempted on unreferenced axis, or move attempted with servo off
6	PI_CNTR_INVALID_SGA_PARAM	Parameter for SGA not valid
7	PI_CNTR_POS_OUT_OF_LIMITS	Position out of limits
8	PI_CNTR_VEL_OUT_OF_LIMITS	Velocity out of limits
9	PI_CNTR_SET_PIVOT_NOT_POSSIBLE	Attempt to set pivot point while U,V and W not all 0
10	PI_CNTR_STOP	Controller was stopped by command
11	PI_CNTR_SST_OR_SCAN_RANGE	Parameter for SST or for one of the embedded scan algorithms out of range
12	PI_CNTR_INVALID_SCAN_AXES	Invalid axis combination for fast scan
13	PI_CNTR_INVALID_NAV_PARAM	Parameter for NAV out of range
14	PI_CNTR_INVALID_ANALOG_INPUT	Invalid analog channel
15	PI_CNTR_INVALID_AXIS_IDENTIFIER	Invalid axis identifier
16	PI_CNTR_INVALID_STAGE_NAME	Unknown stage name
17	PI_CNTR_PARAM_OUT_OF_RANGE	Parameter out of range
18	PI_CNTR_INVALID_MACRO_NAME	Invalid macro name
19	PI_CNTR_MACRO_RECORD	Error while recording macro
20	PI_CNTR_MACRO_NOT_FOUND	Macro not found
21	PI_CNTR_AXIS_HAS_NO_BRAKE	Axis has no brake
22	PI_CNTR_DOUBLE_AXIS	Axis identifier specified more than once

23	PI_CNTR_ILLEGAL_AXIS	Illegal axis
24	PI_CNTR_PARAM_NR	Incorrect number of parameters
25	PI_CNTR_INVALID_REAL_NR	Invalid floating point number
26	PI_CNTR_MISSING_PARAM	Parameter missing
27	PI_CNTR_SOFT_LIMIT_OUT_OF_RANGE	Soft limit out of range
28	PI_CNTR_NO_MANUAL_PAD	No manual pad found
29	PI_CNTR_NO_JUMP	No more step-response values
30	PI_CNTR_INVALID_JUMP	No step-response values recorded
31	PI_CNTR_AXIS_HAS_NO_REFERENCE	Axis has no reference sensor
32	PI_CNTR_STAGE_HAS_NO_LIM_SWITCH	Axis has no limit switch
33	PI_CNTR_NO_RELAY_CARD	No relay card installed
34	PI_CNTR_CMD_NOT_ALLOWED_FOR_STAGE	Command not allowed for selected stage(s)
35	PI_CNTR_NO_DIGITAL_INPUT	No digital input installed
36	PI_CNTR_NO_DIGITAL_OUTPUT	No digital output configured
37	PI_CNTR_NO_MCM	No more MCM responses
38	PI_CNTR_INVALID_MCM	No MCM values recorded
39	PI_CNTR_INVALID_CNTR_NUMBER	Controller number invalid
40	PI_CNTR_NO_JOYSTICK_CONNECTED	No joystick configured
41	PI_CNTR_INVALID_EGE_AXIS	Invalid axis for electronic gearing, axis can not be slave
42	PI_CNTR_SLAVE_POSITION_OUT_OF_RANGE	Position of slave axis is out of range
43	PI_CNTR_COMMAND_EGE_SLAVE	Slave axis cannot be commanded directly when electronic gearing is enabled
44	PI_CNTR_JOYSTICK_CALIBRATION_FAILED	Calibration of joystick failed
45	PI_CNTR_REFERENCING_FAILED	Referencing failed
46	PI_CNTR_OPM_MISSING	OPM (Optical Power Meter) missing
47	PI_CNTR_OPM_NOT_INITIALIZED	OPM (Optical Power Meter) not initialized or cannot be initialized
48	PI_CNTR_OPM_COM_ERROR	OPM (Optical Power Meter) Communication Error
49	PI_CNTR_MOVE_TO_LIMIT_SWITCH_FAILED	Move to limit switch failed
50	PI_CNTR_REF_WITH_REF_DISABLED	Attempt to reference axis with referencing disabled
51	PI_CNTR_AXIS_UNDER_JOYSTICK_CONTROL	Selected axis is controlled by joystick

52	PI_CNTR_COMMUNICATION_ERROR	Controller detected communication error
53	PI_CNTR_DYNAMIC_MOVE_IN_PROCESS	MOV! motion still in progress
54	PI_CNTR_UNKNOWN_PARAMETER	Unknown parameter
55	PI_CNTR_NO_REP_RECORDED	No commands were recorded with REP
56	PI_CNTR_INVALID_PASSWORD	Password invalid
57	PI_CNTR_INVALID_RECORDER_CHAN	Data Record Table does not exist
58	PI_CNTR_INVALID_RECORDER_SRC_OPT	Source does not exist; number too low or too high
59	PI_CNTR_INVALID_RECORDER_SRC_CHAN	Source Record Table number too low or too high
60	PI_CNTR_PARAM_PROTECTION	Protected Param: current Command Level (CCL) too low
61	PI_CNTR_AUTOZERO_RUNNING	Command execution not possible while Autozero is running
62	PI_CNTR_NO_LINEAR_AXIS	Autozero requires at least one linear axis
63	PI_CNTR_INIT_RUNNING	Initialization still in progress
64	PI_CNTR_READ_ONLY_PARAMETER	Parameter is read-only
65	PI_CNTR_PAM_NOT_FOUND	Parameter not found in non-volatile memory
66	PI_CNTR_VOL_OUT_OF_LIMITS	Voltage out of limits
67	PI_CNTR_WAVE_TOO_LARGE	Not enough memory available for requested wave curve
68	PI_CNTR_NOT_ENOUGH_DDL_MEMORY	Not enough memory available for DDL table; DDL can not be started
69	PI_CNTR_DDL_TIME_DELAY_TOO_LARGE	Time delay larger than DDL table; DDL can not be started
70	PI_CNTR_DIFFERENT_ARRAY_LENGTH	The requested arrays have different lengths; query them separately
71	PI_CNTR_GEN_SINGLE_MODE_RESTART	Attempt to restart the generator while it is running in single step mode
72	PI_CNTR_ANALOG_TARGET_ACTIVE	Motion commands and wave generator activation are not allowed when analog target is active
73	PI_CNTR_WAVE_GENERATOR_ACTIVE	Motion commands are not allowed when wave generator is active

74	PI_CNTR_AUTOZERO_DISABLED	No sensor channel or no piezo channel connected to selected axis (sensor and piezo matrix)
75	PI_CNTR_NO_WAVE_SELECTED	Generator started (WGO) without having selected a wave table (WSL).
76	PI_CNTR_IF_BUFFER_OVERRUN	Interface buffer did overrun and command couldn't be received correctly
77	PI_CNTR_NOT_ENOUGH_RECORDED_DATA	Data Record Table does not hold enough recorded data
78	PI_CNTR_TABLE_DEACTIVATED	Data Record Table is not configured for recording
79	PI_CNTR_OPENLOOP_VALUE_SET_WHEN_SERVO_ON	Open-loop commands (SVA, SVR) are not allowed when servo is on
80	PI_CNTR_RAM_ERROR	Hardware error affecting RAM
81	PI_CNTR_MACRO_UNKNOWN_COMMAND	Not macro command
82	PI_CNTR_MACRO_PC_ERROR	Macro counter out of range
83	PI_CNTR_JOYSTICK_ACTIVE	Joystick is active
84	PI_CNTR_MOTOR_IS_OFF	Motor is off
85	PI_CNTR_ONLY_IN_MACRO	Macro-only command
86	PI_CNTR_JOYSTICK_UNKNOWN_AXIS	Invalid joystick axis
87	PI_CNTR_JOYSTICK_UNKNOWN_ID	Joystick unknown
88	PI_CNTR_REF_MODE_IS_ON	Move without referenced stage
89	PI_CNTR_NOT_ALLOWED_IN_CURRENT_MOTION_MODE	Command not allowed in current motion mode
90	PI_CNTR_DIO_AND_TRACING_NOT_POSSIBLE	No tracing possible while digital IOs are used on this HW revision. Reconnect to switch operation mode.
91	PI_CNTR_COLLISION	Move not possible, would cause collision
92	PI_CNTR_SLAVE_NOT_FAST_ENOUGH	Stage is not capable of following the master. Check the gear ratio.
93	PI_CNTR_CMD_NOT_ALLOWED_WHILE_AXIS_IN_MOTION	This command is not allowed while the affected axis or its master is in motion.
94	PI_CNTR_OPEN_LOOP_JOYSTICK_ENABLED	Servo cannot be switched on when open-loop joystick control is enabled.
95	PI_CNTR_INVALID_SERVO_STATE_FOR_PARAMETER	This parameter cannot be changed in current servo mode.

96	PI_CNTR_UNKNOWN_STAGE_NAME	Unknown stage name
100	PI_LABVIEW_ERROR	PI LabVIEW driver reports error. See source control for details.
200	PI_CNTR_NO_AXIS	No stage connected to axis
201	PI_CNTR_NO_AXIS_PARAM_FILE	File with axis parameters not found
202	PI_CNTR_INVALID_AXIS_PARAM_FILE	Invalid axis parameter file
203	PI_CNTR_NO_AXIS_PARAM_BACKUP	Backup file with axis parameters not found
204	PI_CNTR_RESERVED_204	PI internal error code 204
205	PI_CNTR_SMO_WITH_SERVO_ON	SMO with servo on
206	PI_CNTR_UUDECODE_INCOMPLETE_HEADER	uudecode: incomplete header
207	PI_CNTR_UUDECODE_NOTHING_TO_DECODE	uudecode: nothing to decode
208	PI_CNTR_UUDECODE_ILLEGAL_FORMAT	uudecode: illegal UUE format
209	PI_CNTR_CRC32_ERROR	CRC32 error
210	PI_CNTR_ILLEGAL_FILENAME	Illegal file name (must be 8-0 format)
211	PI_CNTR_FILE_NOT_FOUND	File not found on controller
212	PI_CNTR_FILE_WRITE_ERROR	Error writing file on controller
213	PI_CNTR_DTR_HINDERS_VELOCITY_CHANGE	VEL command not allowed in DTR Command Mode
214	PI_CNTR_POSITION_UNKNOWN	Position calculations failed
215	PI_CNTR_CONN_POSSIBLY_BROKEN	The connection between controller and stage may be broken
216	PI_CNTR_ON_LIMIT_SWITCH	The connected stage has driven into a limit switch, some controllers need CLR to resume operation
217	PI_CNTR_UNEXPECTED_STRUT_STOP	Strut test command failed because of an unexpected strut stop
218	PI_CNTR_POSITION_BASED_ON_ESTIMATION	While MOV! is running position can only be estimated!
219	PI_CNTR_POSITION_BASED_ON_INTERPOLATION	Position was calculated during MOV motion
230	PI_CNTR_INVALID_HANDLE	Invalid handle
231	PI_CNTR_NO_BIOS_FOUND	No bios found
232	PI_CNTR_SAVE_SYS_CFG_FAILED	Save system configuration failed
233	PI_CNTR_LOAD_SYS_CFG_FAILED	Load system configuration failed
301	PI_CNTR_SEND_BUFFER_OVERFLOW	Send buffer overflow
302	PI_CNTR_VOLTAGE_OUT_OF_LIMITS	Voltage out of limits

303	PI_CNTR_OPEN_LOOP_MOTION_SET_WHEN_SERVO_ON	Open-loop motion attempted when servo ON
304	PI_CNTR_RECEIVING_BUFFER_OVERFLOW	Received command is too long
305	PI_CNTR_EEPROM_ERROR	Error while reading/writing EEPROM
306	PI_CNTR_I2C_ERROR	Error on I2C bus
307	PI_CNTR_RECEIVING_TIMEOUT	Timeout while receiving command
308	PI_CNTR_TIMEOUT	A lengthy operation has not finished in the expected time
309	PI_CNTR_MACRO_OUT_OF_SPACE	Insufficient space to store macro
310	PI_CNTR_EUI_OLDVERSION_CFGDATA	Configuration data has old version number
311	PI_CNTR_EUI_INVALID_CFGDATA	Invalid configuration data
333	PI_CNTR_HARDWARE_ERROR	Internal hardware error
400	PI_CNTR_WAV_INDEX_ERROR	Wave generator index error
401	PI_CNTR_WAV_NOT_DEFINED	Wave table not defined
402	PI_CNTR_WAV_TYPE_NOT_SUPPORTED	Wave type not supported
403	PI_CNTR_WAV_LENGTH_EXCEEDS_LIMIT	Wave length exceeds limit
404	PI_CNTR_WAV_PARAMETER_NR	Wave parameter number error
405	PI_CNTR_WAV_PARAMETER_OUT_OF_LIMIT	Wave parameter out of range
406	PI_CNTR_WGO_BIT_NOT_SUPPORTED	WGO command bit not supported
500	PI_CNTR_EMERGENCY_STOP_BUTTON_ACTIVATED	The \"red knob\" is still set and disables system
501	PI_CNTR_EMERGENCY_STOP_BUTTON_WAS_ACTIVATED	The \"red knob\" was activated and still disables system - reanimation required
502	PI_CNTR_REDUNDANCY_LIMIT_EXCEEDED	Position consistency check failed
503	PI_CNTR_COLLISION_SWITCH_ACTIVATED	Hardware collision sensor(s) are activated
504	PI_CNTR_FOLLOWING_ERROR	Strut following error occurred, e.g. caused by overload or encoder failure
505	PI_CNTR_SENSOR_SIGNAL_INVALID	One sensor signal is not valid
506	PI_CNTR_SERVO_LOOP_UNSTABLE	Servo loop was unstable due to wrong parameter setting and switched off to avoid damage.
530	PI_CNTR_NODE_DOES_NOT_EXIST	A command refers to a node that does not exist

531	PI_CNTR_PARENT_NODE_DOES_NOT_EXIST	A command refers to a node that has no parent node
532	PI_CNTR_NODE_IN_USE	Attempt to delete a node that is in use
533	PI_CNTR_NODE_DEFINITION_IS_CYCLIC	Definition of a node is cyclic
534	PI_CNTR_NODE_CHAIN_INVALID	The node chain does not end in the '\0\' node
535	PI_CNTR_NODE_DEFINITION_NOT_CONSISTENT	The definition of a coordinate transformation is erroneous
536	PI_CNTR_HEXAPOD_IN_MOTION	Transformation cannot be defined as long as Hexapod is in motion
537	PI_CNTR_TRANSFORMATION_TYPE_NOT_SUPPORTED	Transformation node cannot be activated
538	PI_CNTR_NODE_TYPE_DIFFERS	A node can only be replaced by a node of the same type
539	PI_CNTR_NODE_PARENT_IDENTICAL_TO_CHILD	A node cannot be linked to itself
540	PI_CNTR_NODE_DEFINITION_INCONSISTENT	Node definition is erroneous or not complete (replace or delete it)
541	PI_CNTR_ZERO_NODE_CANNOT_BE_CHANGED_OR_REPLACED	0 is the root node and cannot be modified
542	PI_CNTR_NODES_NOT_IN_SAME_CHAIN	The nodes are not part of the same chain
543	PI_CNTR_NODE_MEMORY_FULL	Unused nodes must be deleted before new nodes can be stored
544	PI_CNTR_PIVOT_POINT_FEATURE_NOT_SUPPORTED	With some transformations pivot point usage is not supported
555	PI_CNTR_UNKNOWN_ERROR	BasMac: unknown controller error
601	PI_CNTR_NOT_ENOUGH_MEMORY	not enough memory
602	PI_CNTR_HW_VOLTAGE_ERROR	hardware voltage error
603	PI_CNTR_HW_TEMPERATURE_ERROR	hardware temperature out of range
604	PI_CNTR_POSITION_ERROR_TOO_HIGH	Position error of any axis in the system is too high
606	PI_CNTR_INPUT_OUT_OF_RANGE	Maximum value of input signal has been exceeded
1000	PI_CNTR_TOO_MANY_NESTED_MACROS	Too many nested macros
1001	PI_CNTR_MACRO_ALREADY_DEFINED	Macro already defined
1002	PI_CNTR_NO_MACRO_RECORDING	Macro recording not activated
1003	PI_CNTR_INVALID_MAC_PARAM	Invalid parameter for MAC
1004	PI_CNTR_RESERVED_1004	PI internal error code 1004

1005	PI_CNTR_CONTROLLER_BUSY	Controller is busy with some lengthy operation (e.g. reference move, fast scan algorithm)
1006	PI_CNTR_INVALID_IDENTIFIER	Invalid identifier (invalid special characters, ...)
1007	PI_CNTR_UNKNOWN_VARIABLE_OR_ARGUMENT	Variable or argument not defined
1008	PI_CNTR_RUNNING_MACRO	Controller is (already) running a macro
1009	PI_CNTR_MACRO_INVALID_OPERATOR	Invalid or missing operator for condition. Check necessary spaces around operator.
1010	PI_CNTR_MACRO_NO_ANSWER	No answer was received while executing WAC/MEX/JRC/...
1011	PI_CMD_NOT_VALID_IN_MACRO_MODE	Command not valid during macro execution
1024	PI_CNTR_MOTION_ERROR	Motion error: position error too large, servo is switched off automatically
1063	PI_CNTR_EXT_PROFILE_UNALLOWED_CMD	User Profile Mode: Command is not allowed, check for required preparatory commands
1064	PI_CNTR_EXT_PROFILE_EXPECTING_MOTION_ERROR	User Profile Mode: First target position in User Profile is too far from current position
1065	PI_CNTR_PROFILE_ACTIVE	Controller is (already) in User Profile Mode
1066	PI_CNTR_PROFILE_INDEX_OUT_OF_RANGE	User Profile Mode: Block or Data Set index out of allowed range
1071	PI_CNTR_PROFILE_OUT_OF_MEMORY	User Profile Mode: Out of memory
1072	PI_CNTR_PROFILE_WRONG_CLUSTER	User Profile Mode: Cluster is not assigned to this axis
1073	PI_CNTR_PROFILE_UNKNOWN_CLUSTER_IDENTIFIER	Unknown cluster identifier
1090	PI_CNTR_TOO_MANY_TCP_CONNECTIONS_OPEN	There are too many open tcpip connections
2000	PI_CNTR_ALREADY_HAS_SERIAL_NUMBER	Controller already has a serial number
4000	PI_CNTR_SECTOR_ERASE_FAILED	Sector erase failed
4001	PI_CNTR_FLASH_PROGRAM_FAILED	Flash program failed
4002	PI_CNTR_FLASH_READ_FAILED	Flash read failed

4003	PI_CNTR_HW_MATCHCODE_ERROR	HW match code missing/invalid
4004	PI_CNTR_FW_MATCHCODE_ERROR	FW match code missing/invalid
4005	PI_CNTR_HW_VERSION_ERROR	HW version missing/invalid
4006	PI_CNTR_FW_VERSION_ERROR	FW version missing/invalid
4007	PI_CNTR_FW_UPDATE_ERROR	FW update failed
5000	PI_CNTR_INVALID_PCC_SCAN_DATA	PicoCompensation scan data is not valid
5001	PI_CNTR_PCC_SCAN_RUNNING	PicoCompensation is running, some actions can not be executed during scanning/recording
5002	PI_CNTR_INVALID_PCC_AXIS	Given axis can not be defined as PPC axis
5003	PI_CNTR_PCC_SCAN_OUT_OF_RANGE	Defined scan area is larger than the travel range
5004	PI_CNTR_PCC_TYPE_NOT_EXISTING	Given PicoCompensation type is not defined
5005	PI_CNTR_PCC_PAM_ERROR	PicoCompensation parameter error
5006	PI_CNTR_PCC_TABLE_ARRAY_TOO_LARGE	PicoCompensation table is larger than maximum table length
5100	PI_CNTR_NEXLINE_ERROR	Common error in NEXLINE® firmware module
5101	PI_CNTR_CHANNEL_ALREADY_USED	Output channel for NEXLINE® can not be redefined for other usage
5102	PI_CNTR_NEXLINE_TABLE_TOO_SMALL	Memory for NEXLINE® signals is too small
5103	PI_CNTR_RNP_WITH_SERVO_ON	RNP can not be executed if axis is in closed loop
5104	PI_CNTR_RNP_NEEDED	Relax procedure (RNP) needed
5200	PI_CNTR_AXIS_NOT_CONFIGURED	Axis must be configured for this action
6000	PI_CNTR_SENSOR_ABS_INVALID_VALUE	Invalid preset value of absolute sensor
6001	PI_CNTR_SENSOR_ABS_WRITE_ERROR	Error while writing to sensor
6002	PI_CNTR_SENSOR_ABS_READ_ERROR	Error while reading from sensor
6003	PI_CNTR_SENSOR_ABS_CRC_ERROR	Checksum error of absolute sensor
6004	PI_CNTR_SENSOR_ABS_ERROR	General error of absolute sensor
6005	PI_CNTR_SENSOR_ABS_OVERFLOW	Overflow of absolute sensor position

Schnittstellenfehler

0	COM_NO_ERROR	No error occurred during function call
-1	COM_ERROR	Error during com operation (could not be specified)
-2	SEND_ERROR	Error while sending data
-3	REC_ERROR	Error while receiving data
-4	NOT_CONNECTED_ERROR	Not connected (no port with given ID open)
-5	COM_BUFFER_OVERFLOW	Buffer overflow
-6	CONNECTION_FAILED	Error while opening port
-7	COM_TIMEOUT	Timeout error
-8	COM_MULTILINE_RESPONSE	There are more lines waiting in buffer
-9	COM_INVALID_ID	There is no interface or DLL handle with the given ID
-10	COM_NOTIFY_EVENT_ERROR	Event/message for notification could not be opened
-11	COM_NOT_IMPLEMENTED	Function not supported by this interface type
-12	COM_ECHO_ERROR	Error while sending "echoed" data
-13	COM_GPIB_EDVR	IEEE488: System error
-14	COM_GPIB_ECIC	IEEE488: Function requires GPIB board to be CIC
-15	COM_GPIB_ENOL	IEEE488: Write function detected no listeners
-16	COM_GPIB_EADR	IEEE488: Interface board not addressed correctly
-17	COM_GPIB_EARG	IEEE488: Invalid argument to function call
-18	COM_GPIB_ESAC	IEEE488: Function requires GPIB board to be SAC
-19	COM_GPIB_EABO	IEEE488: I/O operation aborted
-20	COM_GPIB_ENEB	IEEE488: Interface board not found
-21	COM_GPIB_EDMA	IEEE488: Error performing DMA
-22	COM_GPIB_EOIP	IEEE488: I/O operation started before previous operation completed
-23	COM_GPIB_ECAP	IEEE488: No capability for intended operation
-24	COM_GPIB_EFSO	IEEE488: File system operation error

-25	COM_GPIB_EBUS	IEEE488: Command error during device call
-26	COM_GPIB_ESTB	IEEE488: Serial poll-status byte lost
-27	COM_GPIB_ESRQ	IEEE488: SRQ remains asserted
-28	COM_GPIB_ETAB	IEEE488: Return buffer full
-29	COM_GPIB_ELCK	IEEE488: Address or board locked
-30	COM_RS_INVALID_DATA_BITS	RS-232: 5 data bits with 2 stop bits is an invalid combination, as is 6, 7, or 8 data bits with 1.5 stop bits
-31	COM_ERROR_RS_SETTINGS	RS-232: Error configuring the COM port
-32	COM_INTERNAL_RESOURCES_ERROR	Error dealing with internal system resources (events, threads, ...)
-33	COM_DLL_FUNC_ERROR	A DLL or one of the required functions could not be loaded
-34	COM_FTDIUSB_INVALID_HANDLE	FTDIUSB: invalid handle
-35	COM_FTDIUSB_DEVICE_NOT_FOUND	FTDIUSB: device not found
-36	COM_FTDIUSB_DEVICE_NOT_OPENED	FTDIUSB: device not opened
-37	COM_FTDIUSB_IO_ERROR	FTDIUSB: IO error
-38	COM_FTDIUSB_INSUFFICIENT_RESOURCES	FTDIUSB: insufficient resources
-39	COM_FTDIUSB_INVALID_PARAMETER	FTDIUSB: invalid parameter
-40	COM_FTDIUSB_INVALID_BAUD_RATE	FTDIUSB: invalid baud rate
-41	COM_FTDIUSB_DEVICE_NOT_OPENED_FOR_ERASE	FTDIUSB: device not opened for erase
-42	COM_FTDIUSB_DEVICE_NOT_OPENED_FOR_WRITE	FTDIUSB: device not opened for write
-43	COM_FTDIUSB_FAILED_TO_WRITE_DEVICE	FTDIUSB: failed to write device
-44	COM_FTDIUSB_EEPROM_READ_FAILED	FTDIUSB: EEPROM read failed
-45	COM_FTDIUSB_EEPROM_WRITE_FAILED	FTDIUSB: EEPROM write failed
-46	COM_FTDIUSB_EEPROM_ERASE_FAILED	FTDIUSB: EEPROM erase failed
-47	COM_FTDIUSB_EEPROM_NOT_PRESENT	FTDIUSB: EEPROM not present
-48	COM_FTDIUSB_EEPROM_NOT_PROGRAMMED	FTDIUSB: EEPROM not programmed
-49	COM_FTDIUSB_INVALID_ARGS	FTDIUSB: invalid arguments
-50	COM_FTDIUSB_NOT_SUPPORTED	FTDIUSB: not supported
-51	COM_FTDIUSB_OTHER_ERROR	FTDIUSB: other error
-52	COM_PORT_ALREADY_OPEN	Error while opening the COM port: was already open
-53	COM_PORT_CHECKSUM_ERROR	Checksum error in received data from COM port

-54	COM_SOCKET_NOT_READY	Socket not ready, you should call the function again
-55	COM_SOCKET_PORT_IN_USE	Port is used by another socket
-56	COM_SOCKET_NOT_CONNECTED	Socket not connected (or not valid)
-57	COM_SOCKET_TERMINATED	Connection terminated (by peer)
-58	COM_SOCKET_NO_RESPONSE	Can't connect to peer
-59	COM_SOCKET_INTERRUPTED	Operation was interrupted by a nonblocked signal
-60	COM_PCI_INVALID_ID	No device with this ID is present
-61	COM_PCI_ACCESS_DENIED	Driver could not be opened (on Vista: run as administrator!)

DLL-Fehler

-1001	PI_UNKNOWN_AXIS_IDENTIFIER	Unknown axis identifier
-1002	PI_NR_NAV_OUT_OF_RANGE	Number for NAV out of range--must be in [1,10000]
-1003	PI_INVALID_SGA	Invalid value for SGA--must be one of 1, 10, 100, 1000
-1004	PI_UNEXPECTED_RESPONSE	Controller sent unexpected response
-1005	PI_NO_MANUAL_PAD	No manual control pad installed, calls to SMA and related commands are not allowed
-1006	PI_INVALID_MANUAL_PAD_KNOB	Invalid number for manual control pad knob
-1007	PI_INVALID_MANUAL_PAD_AXIS	Axis not currently controlled by a manual control pad
-1008	PI_CONTROLLER_BUSY	Controller is busy with some lengthy operation (e.g. reference move, fast scan algorithm)
-1009	PI_THREAD_ERROR	Internal error--could not start thread
-1010	PI_IN_MACRO_MODE	Controller is (already) in macro mode--command not valid in macro mode
-1011	PI_NOT_IN_MACRO_MODE	Controller not in macro mode--command not valid unless macro mode active
-1012	PI_MACRO_FILE_ERROR	Could not open file to write or read macro

-1013	PI_NO_MACRO_OR_EMPTY	No macro with given name on controller, or macro is empty
-1014	PI_MACRO_EDITOR_ERROR	Internal error in macro editor
-1015	PI_INVALID_ARGUMENT	One or more arguments given to function is invalid (empty string, index out of range, ...)
-1016	PI_AXIS_ALREADY_EXISTS	Axis identifier is already in use by a connected stage
-1017	PI_INVALID_AXIS_IDENTIFIER	Invalid axis identifier
-1018	PI_COM_ARRAY_ERROR	Could not access array data in COM server
-1019	PI_COM_ARRAY_RANGE_ERROR	Range of array does not fit the number of parameters
-1020	PI_INVALID_SPA_CMD_ID	Invalid parameter ID given to SPA or SPA?
-1021	PI_NR_AVG_OUT_OF_RANGE	Number for AVG out of range--must be >0
-1022	PI_WAV_SAMPLES_OUT_OF_RANGE	Incorrect number of samples given to WAV
-1023	PI_WAV_FAILED	Generation of wave failed
-1024	PI_MOTION_ERROR	Motion error: position error too large, servo is switched off automatically
-1025	PI_RUNNING_MACRO	Controller is (already) running a macro
-1026	PI_PZT_CONFIG_FAILED	Configuration of PZT stage or amplifier failed
-1027	PI_PZT_CONFIG_INVALID_PARAMS	Current settings are not valid for desired configuration
-1028	PI_UNKNOWN_CHANNEL_IDENTIFIER	Unknown channel identifier
-1029	PI_WAVE_PARAM_FILE_ERROR	Error while reading/writing wave generator parameter file
-1030	PI_UNKNOWN_WAVE_SET	Could not find description of wave form. Maybe WG.INI is missing?
-1031	PI_WAVE_EDITOR_FUNC_NOT_LOADED	The WGWaveEditor DLL function was not found at startup
-1032	PI_USER_CANCELLED	The user cancelled a dialog
-1033	PI_C844_ERROR	Error from C-844 Controller
-1034	PI_DLL_NOT_LOADED	DLL necessary to call function not loaded, or function not found in DLL

-1035	PI_PARAMETER_FILE_PROTECTED	The open parameter file is protected and cannot be edited
-1036	PI_NO_PARAMETER_FILE_OPENED	There is no parameter file open
-1037	PI_STAGE_DOES_NOT_EXIST	Selected stage does not exist
-1038	PI_PARAMETER_FILE_ALREADY_OPENED	There is already a parameter file open. Close it before opening a new file
-1039	PI_PARAMETER_FILE_OPEN_ERROR	Could not open parameter file
-1040	PI_INVALID_CONTROLLER_VERSION	The version of the connected controller is invalid
-1041	PI_PARAM_SET_ERROR	Parameter could not be set with SPA--parameter not defined for this controller!
-1042	PI_NUMBER_OF_POSSIBLE_WAVES_EXCEEDED	The maximum number of wave definitions has been exceeded
-1043	PI_NUMBER_OF_POSSIBLE_GENERATORS_EXCEEDED	The maximum number of wave generators has been exceeded
-1044	PI_NO_WAVE_FOR_AXIS_DEFINED	No wave defined for specified axis
-1045	PI_CANT_STOP_OR_START_WAV	Wave output to axis already stopped/started
-1046	PI_REFERENCE_ERROR	Not all axes could be referenced
-1047	PI_REQUIRED_WAVE_NOT_FOUND	Could not find parameter set required by frequency relation
-1048	PI_INVALID_SPP_CMD_ID	Command ID given to SPP or SPP? is not valid
-1049	PI_STAGE_NAME_ISNT_UNIQUE	A stage name given to CST is not unique
-1050	PI_FILE_TRANSFER_BEGIN_MISSING	A uuencoded file transferred did not start with "begin" followed by the proper filename
-1051	PI_FILE_TRANSFER_ERROR_TEMP_FILE	Could not create/read file on host PC
-1052	PI_FILE_TRANSFER_CRC_ERROR	Checksum error when transferring a file to/from the controller
-1053	PI_COULDNT_FIND_PISTAGES_DAT	The PiStages.dat database could not be found. This file is required to connect a stage with the CST command
-1054	PI_NO_WAVE_RUNNING	No wave being output to specified axis
-1055	PI_INVALID_PASSWORD	Invalid password

-1056	PI_OPM_COM_ERROR	Error during communication with OPM (Optical Power Meter), maybe no OPM connected
-1057	PI_WAVE_EDITOR_WRONG_PARAMNUM	WaveEditor: Error during wave creation, incorrect number of parameters
-1058	PI_WAVE_EDITOR_FREQUENCY_OUT_OF_RANGE	WaveEditor: Frequency out of range
-1059	PI_WAVE_EDITOR_WRONG_IP_VALUE	WaveEditor: Error during wave creation, incorrect index for integer parameter
-1060	PI_WAVE_EDITOR_WRONG_DP_VALUE	WaveEditor: Error during wave creation, incorrect index for floating point parameter
-1061	PI_WAVE_EDITOR_WRONG_ITEM_VALUE	WaveEditor: Error during wave creation, could not calculate value
-1062	PI_WAVE_EDITOR_MISSING_GRAPH_COMPONENT	WaveEditor: Graph display component not installed
-1063	PI_EXT_PROFILE_UNALLOWED_CMD	User Profile Mode: Command is not allowed, check for required preparatory commands
-1064	PI_EXT_PROFILE_EXPECTING_MOTION_ERROR	User Profile Mode: First target position in User Profile is too far from current position
-1065	PI_EXT_PROFILE_ACTIVE	Controller is (already) in User Profile Mode
-1066	PI_EXT_PROFILE_INDEX_OUT_OF_RANGE	User Profile Mode: Block or Data Set index out of allowed range
-1067	PI_PROFILE_GENERATOR_NO_PROFILE	ProfileGenerator: No profile has been created yet
-1068	PI_PROFILE_GENERATOR_OUT_OF_LIMITS	ProfileGenerator: Generated profile exceeds limits of one or both axes
-1069	PI_PROFILE_GENERATOR_UNKNOWN_PARAMETER	ProfileGenerator: Unknown parameter ID in Set/Get Parameter command
-1070	PI_PROFILE_GENERATOR_PAR_OUT_OF_RANGE	ProfileGenerator: Parameter out of allowed range
-1071	PI_EXT_PROFILE_OUT_OF_MEMORY	User Profile Mode: Out of memory
-1072	PI_EXT_PROFILE_WRONG_CLUSTER	User Profile Mode: Cluster is not assigned to this axis
-1073	PI_UNKNOWN_CLUSTER_IDENTIFIER	Unknown cluster identifier

-1074	PI_INVALID_DEVICE_DRIVER_VERSION	The installed device driver doesn't match the required version. Please see the documentation to determine the required device driver version.
-1075	PI_INVALID_LIBRARY_VERSION	The library used doesn't match the required version. Please see the documentation to determine the required library version.
-1076	PI_INTERFACE_LOCKED	The interface is currently locked by another function. Please try again later.
-1077	PI_PARAM_DAT_FILE_INVALID_VERSION	Version of parameter DAT file does not match the required version. Current files are available at www.pi.ws .
-1078	PI_CANNOT_WRITE_TO_PARAM_DAT_FILE	Cannot write to parameter DAT file to store user defined stage type.
-1079	PI_CANNOT_CREATE_PARAM_DAT_FILE	Cannot create parameter DAT file to store user defined stage type.
-1080	PI_PARAM_DAT_FILE_INVALID_REVISION	Parameter DAT file does not have correct revision.
-1081	PI_USERSTAGES_DAT_FILE_INVALID_REVISION	User stages DAT file does not have correct revision.
-1082	PI_SOFTWARE_TIMEOUT	Timeout Error. Some lengthy operation did not finish within expected time.

10 Anpassen von Einstellungen

In diesem Kapitel

Parameter im C-863 ändern	279
Verstellertyp anlegen oder ändern	284
Parameterübersicht	288

10.1 Parameter im C-863 ändern

INFORMATION

Die Werte im permanenten Speicher werden als Standardwerte beim Einschalten oder Neustart des C-863 in den flüchtigen Speicher geladen und sind sofort gültig.

INFORMATION

Die Anzahl der Schreibzyklen im permanenten Speicher ist durch die begrenzte Lebensdauer des Speicherchips (EEPROM) beschränkt.

- Überschreiben Sie die Standardwerte nur, wenn es notwendig ist.
- Sichern Sie die aktuellen Parameterwerte auf dem PC (S. 281), bevor Sie Änderungen im permanenten Speicher durchführen.
- Wenden Sie sich an unseren Kundendienst (S. 309), wenn der C-863 ein unerwartetes Verhalten zeigt.

10.1.1 Allgemeine Befehle für Parameter

Für die Änderung von Parametern stehen folgende allgemeine Befehle zur Verfügung:

Befehl	Funktion
SPA	Parameter im flüchtigen Speicher ändern.
SEP	Parameter im permanenten Speicher ändern.
WPA	Aktuellen Parameterwert vom flüchtigen in den permanenten Speicher kopieren. Dort wird er als Standardwert verwendet.
RPA	Parameterwert vom permanenten in den flüchtigen Speicher kopieren.

SPA?	Parameterwerte aus dem flüchtigen Speicher abfragen.
SEP?	Parameterwerte aus dem permanenten Speicher abfragen.

Einzelheiten finden Sie in den Befehlsbeschreibungen (S. 166).

Einfacheren Zugang zu den Parameterwerten erhalten Sie mit der PC-Software PIMikroMove®.

10.1.2 Spezielle Befehle für Parameter

Die folgenden, speziellen Befehle ändern die zugehörigen Parameter nur im flüchtigen Speicher. Die geänderten Werte müssen bei Bedarf mit dem Befehl WPA (S. 260) in den permanenten Speicher geschrieben werden.

INFORMATION

Die nachfolgend aufgelisteten Parameter können auch mit den allgemeinen Befehlen geändert werden.

Befehl	Einstellbare Parameter
ACC	Beschleunigung im geregelten Betrieb (ID 0xB)
DEC	Abbremsung im geregelten Betrieb (ID 0xC)
VEL	Geschwindigkeit im geregelten Betrieb (ID 0x49)

Einzelheiten finden Sie in den Befehlsbeschreibungen (S. 166).

10.1.3 Parameterwerte in Textdatei sichern

INFORMATION

Der C-863 wird über Parameter konfiguriert, z. B. zur Anpassung an die angeschlossene Mechanik. Das Ändern von Parameterwerten kann zu unerwünschten Ergebnissen führen.

- Legen Sie vor dem Ändern der Parametereinstellungen des C-863 eine Sicherungskopie auf dem PC an. Sie können dann jederzeit die Originaleneinstellungen wiederherstellen.
- Erstellen Sie nach jeder Optimierung der Parameterwerte oder Anpassung des C-863 an einen bestimmten Versteller eine weitere Sicherungskopie mit neuem Dateinamen.

Voraussetzungen

- ✓ Sie haben die allgemeinen Hinweise zur Inbetriebnahme gelesen und verstanden (S. 73).
- ✓ Sie haben die Kommunikation zwischen dem C-863 und dem PC mit PIMikroMove® oder PITerminal über die RS-232-Schnittstelle (S. 78) oder die USB-Schnittstelle (S. 80) hergestellt.

Parameterwerte in Textdatei sichern

1. Wenn Sie PIMikroMove® verwenden, öffnen Sie das Fenster zum Senden von Befehlen:
 - Wählen Sie im Hauptfenster den Menüeintrag **Tools > Command entry** oder drücken Sie die Taste **F4** auf der Tastatur.

In PITerminal ist nach dem Herstellen der Kommunikation automatisch das Hauptfenster geöffnet, aus dem Befehle gesendet werden können.
2. Fragen Sie die Parameterwerte ab, von denen Sie eine Sicherheitskopie erstellen möchten.
 - Wenn Sie die Parameterwerte aus dem flüchtigen Speicher des C-863 sichern möchten: Senden Sie den Befehl **SPA?**.
 - Wenn Sie die Parameterwerte aus dem permanenten Speicher des C-863 sichern möchten: Senden Sie den Befehl **SEP?**.
3. Klicken Sie auf die Schaltfläche **Save...**.
Das Fenster **Save content of terminal as textfile** öffnet sich.
4. Speichern Sie im Fenster **Save content of terminal as textfile** die abgefragten Parameterwerte in einer Textdatei auf Ihrem PC.

10.1.4 Parameter ändern: Generelle Vorgehensweise

HINWEIS



Unpassende Parametereinstellungen!

Die Werte im permanenten Speicher werden als Standardwerte beim Einschalten oder Neustart des C-863 in den flüchtigen Speicher geladen und sind sofort gültig. Unpassende Parametereinstellungen können zur Beschädigung des angeschlossenen Verstellers führen.

- Ändern Sie Parameter nur nach sorgfältiger Überlegung.
- Sichern Sie die aktuellen Parameterwerte auf dem PC (S. 281), bevor Sie Änderungen im permanenten Speicher durchführen.

Verfügbare Parameter

Die zur Anpassung des C-863 an Ihre Anwendung verfügbaren Parameter hängen von der Firmware Ihres C-863 ab.

- Senden Sie den Befehl `HPA?` (S. 205), um eine Liste aller verfügbaren Parameter mit Kurzbeschreibung zu erhalten.

Parameter im permanenten Speicher

- Senden Sie den Befehl `SEP?` (S. 237), um eine Liste der Parameterwerte im permanenten Speicher zu erhalten.

INFORMATION

Die Parameterwerte im permanenten Speicher werden beim Einschalten oder Neustart des C-863 automatisch in den flüchtigen Speicher geladen.

- Ändern Sie die Parameter im permanenten Speicher nur, wenn Sie sicher sind, dass der C-863 mit den Parameterwerten korrekt funktioniert.

- Ändern Sie die Parameter im permanenten Speicher mit dem Befehl `SEP` (S. 236).

Parameter im flüchtigen Speicher

- Senden Sie den Befehl `SPA?` (S. 243), um eine Liste der Parameterwerte im flüchtigen Speicher zu erhalten.
- Ändern Sie die Parameter im flüchtigen Speicher mit dem Befehl `SPA` (S. 240).

- und/oder -

- Ändern Sie ausgewählte Parameter im flüchtigen Speicher mit speziellen Befehlen (S. 280).

Wenn Sie mit PIMikroMove® arbeiten:

1. Öffnen Sie im Hauptfenster von PIMikroMove® das Einzelachsen-Fenster für den angeschlossenen Versteller, indem Sie den Versteller im Menü **View > Single Axis Window** auswählen.
2. Erweitern Sie die Ansicht des Einzelachsen-Fensters, indem Sie auf die Schaltfläche > am rechten Fensterrand klicken.
3. Wenn der zu ändernde Parameter nicht in der Liste auf der rechten Seite des Fensters enthalten ist, klicken Sie auf **Select parameters...** und fügen ihn zur Liste hinzu.
4. Tippen Sie den neuen Parameterwert in das entsprechende Eingabefeld in der Liste ein.
5. Drücken Sie auf der Tastatur des PC die **Enter**-Taste, um den Parameterwert in den flüchtigen Speicher des Controllers zu übertragen. Dabei ändert der Eintrag seine Farbe von Blau in Schwarz.

Parameter vom flüchtigen Speicher in den permanenten Speicher schreiben

1. Ändern Sie die Parameter im flüchtigen Speicher mit dem Befehl **SPA** (S. 240).
2. Prüfen Sie, ob der C-863 mit den geänderten Parametern korrekt funktioniert.
 - Wenn ja:
Schreiben Sie die geänderten Parameterwerte mit dem Befehl **WPA** (S. 260) in den permanenten Speicher.
 - Wenn nein:
Ändern und prüfen Sie die Parameter im flüchtigen Speicher erneut.

Wenn Sie mit PIMikroMove® arbeiten:

- Schreiben Sie die aktuellen Werte aller Parameter in den permanenten Speicher:
 - a) Wählen Sie im Hauptfenster von PIMikroMove® den Menüeintrag **C-863 > Save parameters to non-volatile memory**.
 - b) Geben Sie nach Aufforderung das Kennwort 100 ein.

Parameter vom permanenten Speicher in den flüchtigen Speicher schreiben

INFORMATION

- Wenden Sie dieses Vorgehen nur an, wenn Sie sicher sind, dass der C-863 mit den Parameterwerten im permanenten Speicher korrekt funktioniert.
- Schreiben Sie die Parameterwerte mit dem Befehl **RPA** (S. 233) vom permanenten Speicher in den flüchtigen Speicher.

10.2 Verstellertyp anlegen oder ändern

Sie können in der PC-Software von PI den für Ihren Versteller geeigneten Parametersatz aus einer Verstellerdatenbank auswählen. Die Software überträgt die Werte des ausgewählten Parametersatzes in den flüchtigen Speicher des Controllers. Weitere Informationen siehe "Verstellerdatenbanken" (S. 47).

Die Verstellerdatenbank **PI_UserStages2.dat** ist für die Anlage, Bearbeitung und Speicherung neuer Parametersätze vorgesehen. Dies kann z. B. in folgenden Fällen erforderlich sein:

- Sie möchten einen Versteller mit anderen Regelparametern als denjenigen aus der Verstellerdatenbank **PIStages2.dat** betreiben.
- Sie möchten die Verfahrbereichsgrenzen des Verstellers an Ihre Anwendung anpassen.
- Sie haben einen kundenspezifischen Versteller.

INFORMATION

Einen neuen Verstellertyp können Sie am einfachsten anlegen, indem Sie in **PIMikroMove®** einen vorhandenen Verstellertyp ändern und ihn unter einer neuen Benennung abspeichern.

INFORMATION

Wenn in den Verstellerdatenbanken PISTages2.dat und PI_UserStages2.dat ein Verstellertyp mit gleicher Benennung vorhanden ist, werden bei der Auswahl des Verstellertyps in der PC-Software immer die Parametereinstellungen aus PISTages2.dat geladen. Die Parametereinstellungen aus PI_UserStages2.dat werden in diesem Fall nicht verwendet.

- Vergeben Sie beim Speichern von Verstellertypen nur Benennungen, die **nicht** bereits in der Verstellerdatenbank PISTages2.dat verwendet werden.

Im Folgenden wird PIMikroMove® zum Anlegen eines neuen Verstellertyps und zum Ändern eines vorhandenen Verstellertyps verwendet.

Voraussetzung

- ✓ PIMikroMove® ist auf dem PC installiert (S. 57).
- ✓ Sie haben das PIMikroMove® Handbuch gelesen und verstanden. Das Handbuch finden Sie auf der Produkt-CD.
- ✓ Sie haben die neueste Version der Verstellerdatenbank PISTages2.dat auf dem PC installiert (S. 58).
- ✓ Wenn Sie für Ihren Versteller eine kundenspezifische Verstellerdatenbank von PI erhalten haben, dann haben Sie diese Datenbank auf Ihrem PC installiert (S. 60).
- ✓ Sie haben die Kommunikation zwischen dem C-863 und dem PC mit PIMikroMove® hergestellt (S. 78).

Verstellertyp in Verstellerdatenbank anlegen

1. Wählen Sie in PIMikroMove® im Schritt **Select connected stages** einen passenden Verstellertyp aus:
 - a) Markieren Sie den Verstellertyp in der Liste **Stage database entries**.
 - b) Klicken Sie auf **Assign**.
 - c) Bestätigen Sie die Auswahl mit **OK**, um die Parametereinstellungen für den ausgewählten Verstellertyp aus der Verstellerdatenbank in den flüchtigen Speicher des C-863 zu laden.

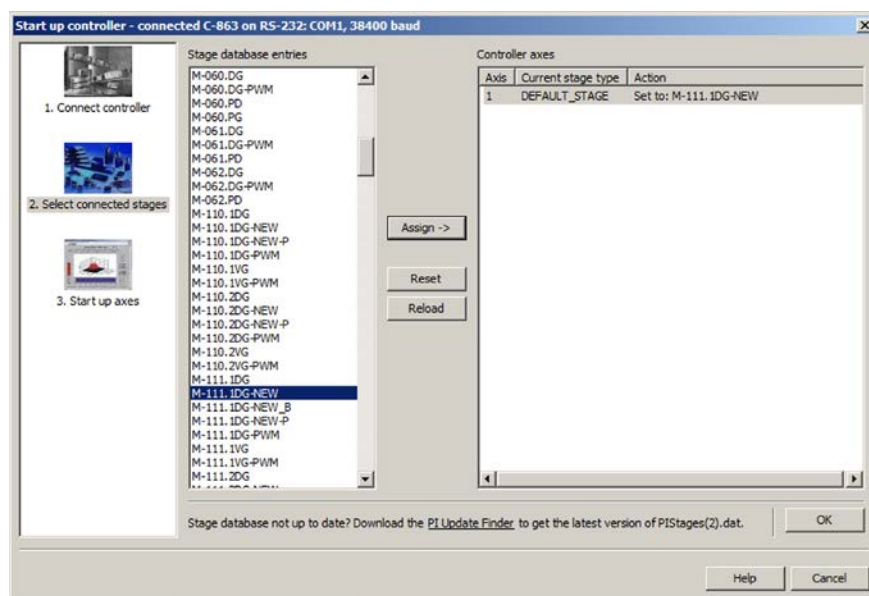


Abbildung 33: Start up controller – Select connected stages

Nach dem Klicken von **OK** wechselt das Fenster **Start up controller** zum Schritt **Start up axes**.

2. Klicken Sie im Schritt **Start up axes** auf **Close**, um das Fenster **Start up controller** zu schließen.
3. Öffnen Sie im Hauptfenster von PIMikroMove® das Einzelachsen-Fenster für den ausgewählten Verstellertyp, indem Sie den Verstellertyp im Menü **View > Single Axis Window** auswählen.
4. Erweitern Sie die Ansicht des Einzelachsen-Fensters, indem Sie auf die Schaltfläche **>** am rechten Fensterrand klicken.

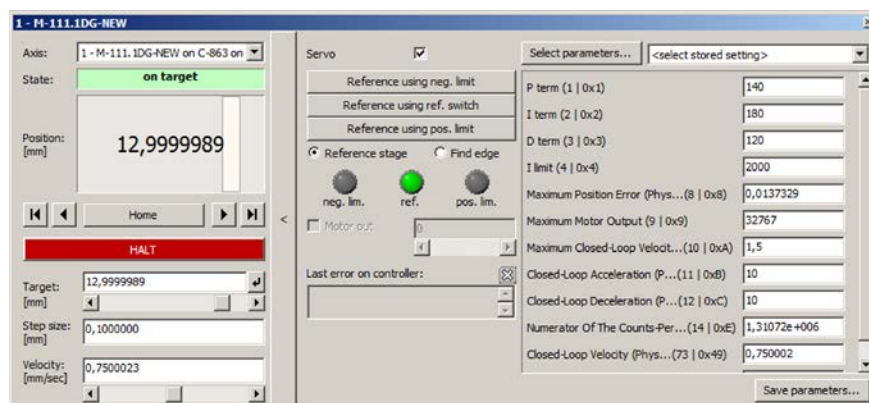


Abbildung 34: PIMikroMove: Erweitertes Einzelachsenfenster (Single Axis Window)

5. Geben Sie neue Werte für die Verstellerparameter ein:
 - a) Wenn der zu ändernde Parameter nicht in der Liste auf der rechten Seite des Fensters enthalten ist, klicken Sie auf **Select parameters...** und fügen ihn zur Liste hinzu.
 - b) Tippen Sie den neuen Parameterwert in das entsprechende Eingabefeld in der Liste ein.
 - c) Drücken Sie auf der Tastatur des PC die **Enter**-Taste, um den Parameterwert in den flüchtigen Speicher des Controllers zu übertragen. Dabei ändert der Eintrag seine Farbe von Blau in Schwarz.
6. Klicken Sie auf die Schaltfläche **Save parameters...** unterhalb der Parameterliste.

Der Dialog **Save Parameters as User Stage Type** öffnet sich.

7. Speichern Sie im Dialog **Save Parameters as User Stage Type** die geänderten Parameterwerte als neuen Verstellertyp:
 - a) Lassen Sie den Eintrag im Feld **Parameters of axis:** unverändert.
 - b) Tragen Sie im Feld **Save as:** die Benennung für den neuen Verstellertyp ein.
 - c) Klicken Sie auf **OK**.

Der neue Verstellertyp wurde in der Verstellerdatenbank PI_UserStages2.dat gespeichert. Die Anzeige des angeschlossenen Verstellertyps wurde im Einzelachsen-Fenster und im Hauptfenster von PIMikroMove® aktualisiert. Der neue Verstellertyp steht ab sofort auch für die Auswahl im Schritt **Select connected stages** zur Verfügung.

Verstellertyp in Verstellerdatenbank ändern

1. Wählen Sie in PIMikroMove® im Schritt **Select connected stages** einen Verstellertyp aus, den Sie zuvor wie oben beschrieben neu angelegt haben. Gehen Sie bei der Auswahl vor wie in Schritt 1 der Anleitung **Verstellertyp in Verstellerdatenbank anlegen**.
2. Führen Sie die Schritte 2 bis 6 aus **Verstellertyp in Verstellerdatenbank anlegen** aus.
3. Speichern Sie im Dialog **Save Parameters as User Stage Type** die geänderten Parameterwerte des Verstellertyps:
 - a) Lassen Sie den Eintrag im Feld **Parameters of axis:** unverändert.
 - b) Lassen Sie den Eintrag im Feld **Save as:** unverändert.
 - c) Klicken Sie auf **OK**.

- d) Klicken Sie im Dialog **Stage type already defined** auf **Change settings**. Der Dialog **Save Parameters as User Stage Type** schließt sich nach kurzer Zeit automatisch.

Die Parameterwerte des Verstellertyps wurden in der Verstellerdatenbank PI_UserStages2.dat sowie im Hauptfenster von PIMikroMove® aktualisiert.

10.3 Parameterübersicht

INFORMATION

Das Kennwort zum Speichern der Parameterwerte im permanenten Speicher lautet 100.

Parameter-ID (hexa-dezimal)	Daten-typ	Parametername	Beschreibung
0x1	INT	P-Term	0 bis 32767; Details siehe "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 29).
0x2	INT	I-Term	0 bis 32767; Details siehe "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 29).
0x3	INT	D-Term	0 bis 32767; Details siehe "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 29).
0x4	INT	I-Limit	0 bis 32767; Details siehe "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 29).
0x5	INT	Kvff	0 bis 32767; Details siehe "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 29).
0x8	FLOAT	Maximum Position Error (Phys. Unit)	Maximaler Positionsfehler Wird für die Erkennung von Bewegungsfehlern verwendet. Details siehe "Bewegungsfehler" (S. 99).
0x9	INT	Maximum Motor Output	Maximaler Stellwert für die Ansteuerung der Achse (dimensionslos) 0 bis 32767
0xA	FLOAT	Maximum Closed-Loop Velocity (Phys. Unit/s)	Maximale Geschwindigkeit im geregelten Betrieb Gibt den Maximalwert für Parameter 0x49 an.

Parameter-ID (hexa-dezimal)	Daten-typ	Parametername	Beschreibung
0xB	FLOAT	Closed-Loop Acceleration (Phys. Unit/s ²)	Beschleunigung im geregelten Betrieb Wird begrenzt durch Parameter 0x4A. Details siehe "Erzeugung des Dynamikprofils" (S. 26).
0xC	FLOAT	Closed-Loop Deceleration (Phys. Unit/s ²)	Abbremsung im geregelten Betrieb Wird begrenzt durch Parameter 0x4B. Details siehe "Erzeugung des Dynamikprofils" (S. 26).
0xE	INT	Numerator Of The Counts-Per-Physical-Unit Factor	Zähler und Nenner des Faktors für Impulse pro physikalische Längeneinheit
0xF	INT	Denominator Of The Counts-Per-Physical-Unit Factor	Details siehe "Physikalische Einheiten" (S. 23).
0x13	INT	Is Rotary Stage?	Handelt es sich um einen Rotationsversteller? 0 = Kein Rotationsversteller 1 = Rotationsversteller Keine Auswertung durch den C-863, sondern nur durch die PC-Software: PIMikroMove® entscheidet anhand dieses Wertes, welche Bewegungen zulässig sind.
0x14	INT	Has Reference?	Hat der Versteller einen Referenzschalter? Details siehe "Referenzschaltererkennung" (S. 33).
0x15	FLOAT	Maximum Travel In Positive Direction (Phys. Unit)	Verfahrbereichsgrenze in positiver Richtung Siehe Beispiele in "Stellweg und Verfahrbereichsgrenzen" (S. 37).
0x16	FLOAT	Value At Reference Position (Phys. Unit)	Positionswert am Referenzschalter Siehe Beispiele in "Stellweg und Verfahrbereichsgrenzen" (S. 37).
0x17	FLOAT	Distance From Negative Limit To Reference Position (Phys. Unit)	Abstand zwischen Referenzschalter und negativem Endschalter Siehe Beispiele in "Stellweg und Verfahrbereichsgrenzen" (S. 37).
0x18	INT	Limit Mode	Signallogik der Endschalter Details siehe "Endschaltererkennung" (S. 34).

Parameter-ID (hexa-dezimal)	Daten-typ	Parametername	Beschreibung
0x1A	INT	Has Brake?	Hat der Versteller eine Bremse? 0 = Keine Bremse vorhanden 1 = Bremse vorhanden. In diesem Fall sind das Ein-/Ausschalten des Servomodus und die Bremsenaktivierung/-deaktivierung miteinander verkoppelt, siehe BRA (S. 174) und SV0 (S. 247).
0x2F	FLOAT	Distance From Reference Position To Positive Limit (Phys. Unit)	Abstand zwischen Referenzschalter und positivem Endschalter Siehe Beispiele in "Stellweg und Verfahrbereichsgrenzen" (S. 37).
0x30	FLOAT	Maximum Travel In Negative Direction (Phys. Unit)	Verfahrbereichsgrenze in negativer Richtung Siehe Beispiele in "Stellweg und Verfahrbereichsgrenzen" (S. 37).
0x31	INT	Invert Reference?	Soll das Referenzsignal invertiert werden? Details siehe "Referenzschaltererkennung" (S. 33).
0x32	INT	Has No Limit Switches?	Hat der Versteller Endschalter? Details siehe "Endschaltererkennung" (S. 34).
0x33	INT	Motor Offset Positive	Offset für die positive Bewegungsrichtung Details siehe "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 29).
0x34	INT	Motor Offset Negative	Offset für die negative Bewegungsrichtung Details siehe "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 29).
0x36	INT	Settling Window (encoder counts)	Einschwingfenster um die Zielposition Details siehe "On-Target-Status" (S. 32).
0x3C	CHAR	Stage Name	Verstellernamen Maximal 20 Zeichen; Standardwert: DEFAULT_STAGE
0x3F	FLOAT	Settling Time (s)	Verzögerungszeit für das Setzen des On-Target-Status. Details siehe "On-Target-Status" (S. 32).

Parameter-ID (hexa-dezimal)	Daten-typ	Parametername	Beschreibung
0x47	INT	Reference Travel Direction	Standardrichtung für die Referenzfahrt Details siehe "Referenzwertbestimmung" (S. 40).
0x48	INT	Motor Drive Offset	Geschwindigkeitsabhängiger Offset Details siehe "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 29).
0x49	FLOAT	Closed-Loop Velocity (Phys. Unit/s)	Geschwindigkeit im geregelten Betrieb Wird begrenzt durch Parameter 0xA Details siehe "Erzeugung des Dynamikprofils" (S. 26).
0x4A	FLOAT	Maximum Closed-Loop Acceleration (Phys. Unit/s ²)	Maximale Beschleunigung im geregelten Betrieb Gibt den Höchstwert für Parameter 0xB an.
0x4B	FLOAT	Maximum Closed-Loop Deceleration (Phys. Unit/s ²)	Maximale Abbremsung im geregelten Betrieb Gibt den Höchstwert für Parameter 0xC an.
0x4D	INT	Servo Window Mode	Nur aus Kompatibilitätsgründen vorhanden.
0x50	FLOAT	Velocity For Reference Moves (Phys. Unit/s)	Geschwindigkeit für Referenzfahrt Details siehe "Referenzwertbestimmung" (S. 40).
0x5A	INT	Numerator Of The Servo-Loop Input Factor	Zähler und Nenner des Eingangsfaktors des Regelkreises Details siehe "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 29).
0x5B	INT	Denominator Of The Servo-Loop Input Factor	
0x5C	INT	Source Of Reference Signal	Referenzsignalquelle für die Befehle FRF oder FED Details siehe "Befehle und Parameter für digitale Eingänge" (S. 114) und "Digitale Eingangssignale als Schaltersignale verwenden" (S. 117).

Parameter-ID (hexa-dezimal)	Daten-typ	Parametername	Beschreibung
0x5D	INT	Source Of Negative Limit Signal	Referenzsignalquelle für die Befehle FNL oder FED Details siehe "Befehle und Parameter für digitale Eingänge" (S. 114) und "Digitale Eingangssignale als Schaltersignale verwenden" (S. 117).
0x5E	INT	Source Of Positive Limit Signal	Referenzsignalquelle für die Befehle FPL oder FED Details siehe "Befehle und Parameter für digitale Eingänge" (S. 114) und "Digitale Eingangssignale als Schaltersignale verwenden" (S. 117).
0x5F	INT	Invert Digital Input Used For Negative Limit	Invertiert die Polarität der digitalen Eingänge, die als Quellen des negativen Endsaltersignals dienen. Details siehe "Befehle und Parameter für digitale Eingänge" (S. 114) und "Digitale Eingangssignale als Schaltersignale verwenden" (S. 117).
0x60	INT	Invert Digital Input Used For Positive Limit	Invertiert die Polarität der digitalen Eingänge, die als Quellen des positiven Endsaltersignals dienen. Details siehe "Befehle und Parameter für digitale Eingänge" (S. 114) und "Digitale Eingangssignale als Schaltersignale verwenden" (S. 117).
0x61	INT	Invert Direction Of Motion For Joystick-Controlled Axis?	Soll die Bewegungsrichtung für joystickgesteuerte Achsen invertiert werden? Details siehe "Befehle und Parameter für Joystick-Steuerung" (S. 121).
0x63	FLOAT	Distance Between Limit And Hard Stop (Phys. Unit)	Abstand zwischen eingebautem Endsaltler und mechanischem Anschlag Details siehe "Referenzwertbestimmung" (S. 40).

Parameter-ID (hexa-dezimal)	Daten-typ	Parametername	Beschreibung
0x70	INT	Reference Signal Type	Art des Referenzsignals Details siehe "Referenzschaltererkennung" (S. 33).
0x71	INT	D-Term Delay (No. Of Servo Cycles)	D-Term-Verzögerung Details siehe "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 29).
0x72	INT	Ignore Macro Error?	Makrofehler ignorieren? Details siehe "Befehle und Parameter für Makros" (S. 132).
0x94	FLOAT	Notch Filter Frequency 1 (Hz)	Frequenz des ersten Notchfilters Details siehe "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 29).
0x95	FLOAT	Notch Filter Edge 1	Anstieg der Flanke des ersten Notchfilters Details siehe "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 29).
0x400	INT	Number of Servo Parameter Sets	Nur aus Kompatibilitätsgründen vorhanden.
0x401	INT	P-Term 0	Parametersatz 0: Nur aus Kompatibilitätsgründen vorhanden.
0x402	INT	I-Term 0	
0x403	INT	D-Term 0	
0x404	INT	I-Limit 0	
0x405	INT	Kvff 0	
0x406	INT	Window Enter 0 (encoder counts)	Nur aus Kompatibilitätsgründen vorhanden. Wert identisch mit dem Wert des Parameters 0x36.
0x407	INT	Window Exit 0 (encoder counts)	Nur aus Kompatibilitätsgründen vorhanden.
0x411	INT	P-Term 1	Parametersatz 1: Nur aus Kompatibilitätsgründen vorhanden. Werte identisch mit den Werten der Parameter 0x1 bis 0x5.
0x412	INT	I-Term 1	
0x413	INT	D-Term 1	
0x414	INT	I-Limit 1	
0x415	INT	Kvff 1	

Parameter-ID (hexa-dezimal)	Daten-typ	Parametername	Beschreibung
0x416	INT	Window Enter 1 (encoder counts)	Nur aus Kompatibilitätsgründen vorhanden.
0x417	INT	Window Exit 1 (encoder counts)	
0x421	INT	P-Term 2	Parametersatz 2: Nur aus Kompatibilitätsgründen vorhanden.
0x422	INT	I-Term 2	
0x423	INT	D-Term 2	
0x424	INT	I-Limit 2	
0x425	INT	Kvff 2	
0x426	INT	Window Enter 2 (encoder counts)	
0x427	INT	Window Exit 2 (encoder counts)	
0x431	INT	P-Term 3	Parametersatz 3: Nur aus Kompatibilitätsgründen vorhanden.
0x432	INT	I-Term 3	
0x433	INT	D-Term 3	
0x434	INT	I-Limit 3	
0x435	INT	Kvff 3	
0x436	INT	Window Enter 3 (encoder counts)	
0x437	INT	Window Exit 3 (encoder counts)	
0x441	INT	P-Term 4	Parametersatz 4: Nur aus Kompatibilitätsgründen vorhanden.
0x442	INT	I-Term 4	
0x443	INT	D-Term 4	
0x444	INT	I-Limit 4	
0x445	INT	Kvff 4	
0x446	INT	Window Enter 4 (encoder counts)	
0x447	INT	Window Exit 4 (encoder counts)	

Parameter-ID (hexa-dezimal)	Daten-typ	Parametername	Beschreibung
0x07000000	FLOAT	Range Limit Min	Zusätzliche Verfahrbereichsgrenze für die negative Bewegungsrichtung (physikalische Einheit) Details siehe "Stellweg und Verfahrbereichsgrenzen" (S. 35).
0x07000001	FLOAT	Range Limit Max	Zusätzliche Verfahrbereichsgrenze für die positive Bewegungsrichtung (physikalische Einheit) Details siehe "Stellweg und Verfahrbereichsgrenzen" (S. 35).
0x07000601	CHAR	Axis Unit	Einheitenzeichen Details siehe "Physikalische Einheiten" (S. 23).

11 Wartung

In diesem Kapitel

C-863 reinigen	297
Firmware aktualisieren	298

11.1 C-863 reinigen

HINWEIS



Kurzschlüsse oder Überschlüsse!

Der C-863 enthält elektrostatisch gefährdete Bauteile, die beim Eindringen von Reinigungsflüssigkeiten in das Gehäuse durch Kurzschlüsse oder Überschlüsse beschädigt werden können.

- Trennen Sie vor dem Reinigen den C-863 von der Stromversorgung, indem Sie den Netzstecker ziehen.
- Vermeiden Sie das Eindringen von Reinigungsflüssigkeit in das Gehäuse.

- Wenn notwendig, reinigen Sie die Gehäuseoberflächen des C-863 mit einem Tuch, das leicht mit einem milden Reinigungs- oder Desinfektionsmittel angefeuchtet wurde.
- Verwenden Sie **keine** organischen Lösungsmittel.

11.2 Firmware aktualisieren

INFORMATION

Der Befehl `*IDN?` liest unter anderem die Versionsnummer der Firmware aus.

Beispiel für eine Antwort des C-863:

```
(c)2011 Physik Instrumente(PI) Karlsruhe, C-863.11,0,1.2.0.0
```

- C-863.11: Gerätebezeichnung
- 1.2.0.0: Firmware-Version

INFORMATION

Wenn sich der C-863 im Firmware-Update-Modus befindet (DIP-Schalter 8 in Stellung "EIN" (oben)), werden die DIP-Schalter-Einstellungen für Baudrate und Controlleradresse ignoriert. Beim Herstellen der Kommunikation mit dem Programm **TMS320F28xx Updater** wird die Baudrate automatisch eingestellt.

INFORMATION

Bei der Aktualisierung der Firmware werden die Parametereinstellungen des C-863 nicht geändert. Wenn mit der Aktualisierung der Firmware neue Parameter eingeführt werden, ist vor dem Betrieb des C-863 eine spezielle Initialisierung für die Werte der neuen Parameter erforderlich.

INFORMATION

Bei der Aktualisierung der Firmware bleiben die auf dem C-863 gespeicherten Makros erhalten.

Voraussetzung

- ✓ Sie haben den C-863 über die RS-232-Schnittstelle an den PC angeschlossen (S. 65).
- ✓ Sie haben sichergestellt, dass der C-863 **nicht** Bestandteil eines Daisy-Chain-Netzwerks ist.
- ✓ Sie haben sichergestellt, dass **kein** Kabel an der Buchse **RS-232 Out** angeschlossen ist.

- ✓ Sie haben die Dokumentation gelesen und verstanden, die Sie mit den aktuellen Flash- und Bootloader-Dateien von unserem Kundendienst erhalten haben. Der Dokumentation haben Sie entnommen, ob mit der Aktualisierung der Firmware neue Parameter eingeführt werden.

Werkzeug und Zubehör

- Aktuelle Flash-Datei (enthält die DSP-Firmware); auf Anfrage von unserem Kundendienst (S. 309) erhältlich
- Aktuelle Bootloader-Datei; auf Anfrage von unserem Kundendienst (S. 309) erhältlich
- Produkt-CD mit dem Programm **TMS320F28xx Updater**, im Lieferumfang

TMS320F28xx Updater und Firmware-Dateien auf dem PC installieren

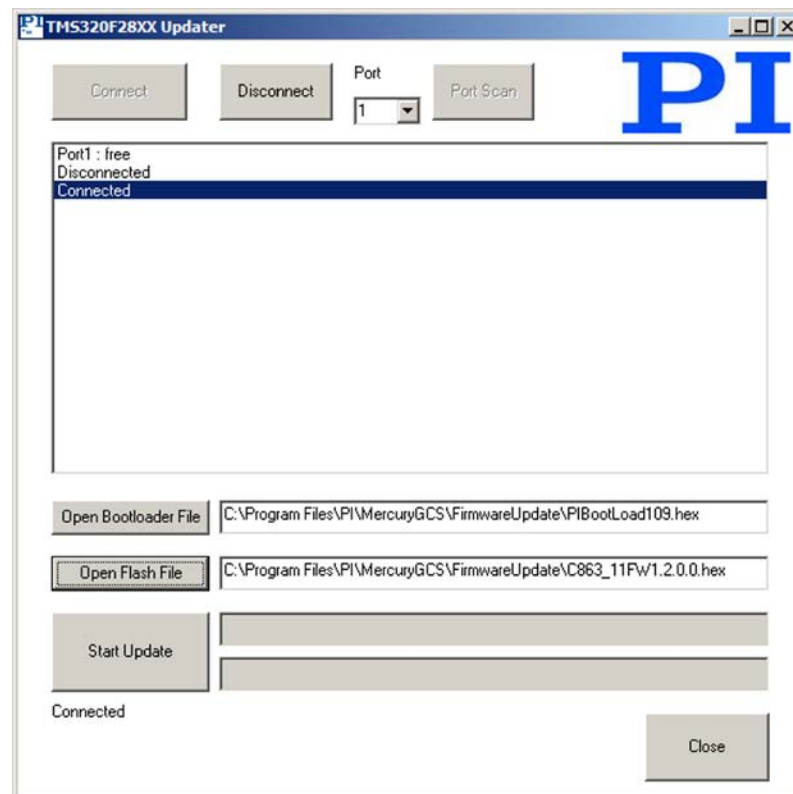
1. Legen Sie die Produkt-CD in das Laufwerk des PC ein.
2. Wenn der Installationsassistent nicht automatisch startet:
Starten Sie den Installationsassistenten, indem Sie im Hauptverzeichnis der CD auf das Icon  klicken.
3. Folgen Sie den Anweisungen am Bildschirm.
 - a) Wählen Sie je nach Verfügbarkeit **Custom** für die benutzerdefinierte Installation oder **Modify** für die Ergänzung einer bereits auf dem PC bestehenden Installation.
 - b) Markieren Sie **Firmware Updater** für das Programm **TMS320F28xx Updater**.
4. Folgen Sie weiterhin den Anweisungen auf dem Bildschirm, um die Installation zu beenden.
5. Kopieren Sie die aktuelle Flash-Datei und die aktuelle Bootloader-Datei in das Installationsverzeichnis des Programms **TMS320F28xx Updater** auf dem PC.

Firmware des C-863 aktualisieren

1. Wenn mit der Aktualisierung der Firmware neue Parameter eingeführt werden: Sichern Sie die aktuellen Parameterwerte des C-863 in einer Textdatei (S. 281).
2. Schalten Sie den C-863 aus, indem Sie das Netzkabel des Netzteils ziehen.
3. Stellen Sie am C-863 mit dem DIP-Schalter 8 den Firmware-Update-Modus (S. 77) ein (Stellung EIN).

4. Schalten Sie den C-863 ein, indem Sie das Netzkabel des Netzteils mit der Steckdose verbinden.
5. Starten Sie am PC das Programm **TMS320F28xx Updater**.
6. Klicken Sie auf die Schaltfläche **Connect**.

Die Kommunikation zwischen C-863 und PC wird hergestellt. In der Meldungsliste des Programmfensters erscheint die Meldung **Connected**.

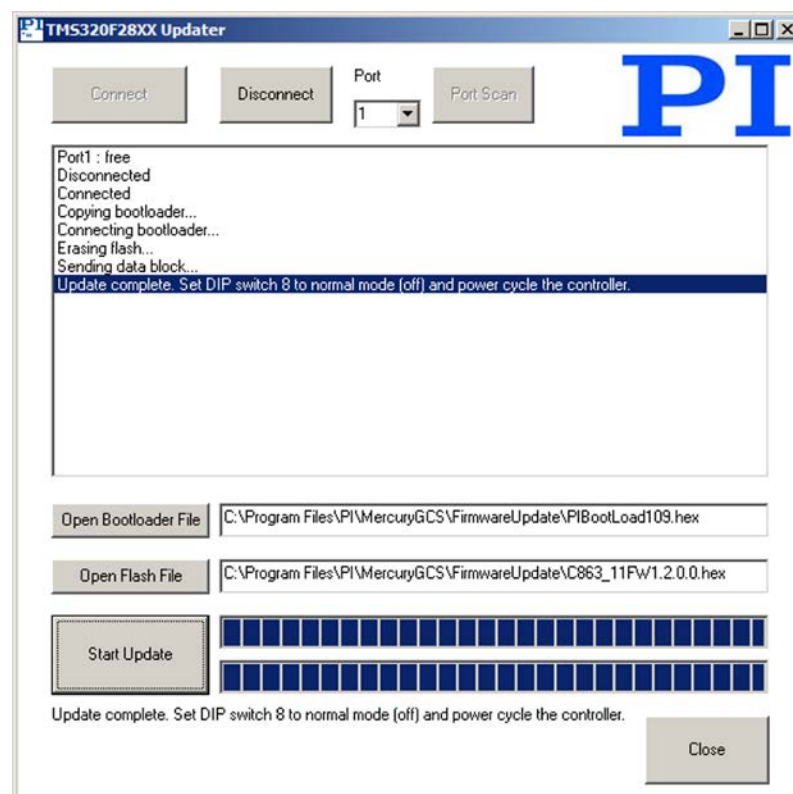


Wenn die Kommunikation nicht hergestellt werden kann:

- a) Klicken Sie auf die Schaltfläche **Port Scan**.
 - b) Ändern Sie gegebenenfalls die Einstellung im Auswahlfeld **Port**.
 - c) Klicken Sie erneut auf **Connect**.
7. Wählen Sie die aktuelle Bootloader-Datei aus:
 - a) Klicken Sie auf die Schaltfläche **Open Bootloader File**.
 - b) Wechseln Sie im Dateiauswahlfenster in das Installationsverzeichnis des Programms **TMS320F28xx Updater**.
 - c) Führen Sie dort einen Doppelklick auf die aktuelle Bootloader-Datei aus.

8. Wählen Sie die aktuelle Flash-Datei aus:
 - a) Klicken Sie auf die Schaltfläche **Open Flash File**.
 - b) Wechseln Sie im Dateiauswahlfenster in das Installationsverzeichnis des Programms **TMS320F28xx Updater**.
 - c) Führen Sie dort einen Doppelklick auf die aktuelle Flash-Datei aus (C863_11FWz.z.z.z.hex, wobei z die Firmwareversion bezeichnet).
9. Starten Sie die Aktualisierung der Firmware, indem Sie auf die Schaltfläche **Start Update** klicken.
 - Wenn sich ein Hinweisenfenster öffnet, folgen Sie den dort angegebenen Anweisungen.

Die Firmware des C-863 wird aktualisiert. Der Fortschritt der Aktualisierung wird in der Meldungsliste und durch die beiden Fortschrittsbalken angezeigt. Die Aktualisierung war erfolgreich, wenn in der Meldungsliste die Meldung **Update complete. Set DIP switch 8 to normal mode (off) and power cycle the controller.** erscheint.



10. Schließen Sie das Programm **TMS320F28xx Updater**, indem Sie auf die Schaltfläche **Close** klicken.

11. Schalten Sie den C-863 aus, indem Sie das Netzkabel des Netzteils ziehen.
12. Stellen Sie am C-863 mit dem DIP-Schalter 8 den Normalbetrieb (S. 77) ein (Stellung AUS).
13. Schalten Sie den C-863 ein, indem Sie das Netzkabel des Netzteils mit der Steckdose verbinden.

Wurden mit der Aktualisierung der Firmware neue Parameter eingeführt?

- Wenn nein: Die Aktualisierung der Firmware ist beendet.
- Wenn ja: Vor dem Betrieb des C-863 ist eine spezielle Initialisierung für die Werte der neuen Parameter erforderlich, siehe unten.

Neue Parameter des C-863 initialisieren

1. Starten Sie am PC PITerminal oder PIMikroMove® und öffnen Sie gegebenenfalls das Fenster zum Senden von Befehlen.
2. Stellen Sie sicher, dass die aktuellen Parameterwerte des C-863 in einer Textdatei gesichert wurden (siehe "Parameterwerte in Textdatei sichern" (S. 281)).

Bei der Initialisierung der neuen Parameter werden **alle** Parameter des C-863 auf ihre Werkseinstellung gesetzt. Ungesicherte Parameterwerte gehen somit bei der Initialisierung verloren.

3. Initialisieren Sie die neuen Parameter, indem Sie den Befehl `ZZZ 100 parameter` senden.
4. Passen Sie die Parameterwerte an (siehe allgemeines Vorgehen zum Ändern von Parametern (S. 282)):
 - Setzen Sie die Parameter, die bereits vor der Firmware-Aktualisierung vorhanden waren, auf die gesicherten Werte aus der Textdatei zurück.
 - Setzen Sie die Parameter, die mit der Aktualisierung der Firmware eingeführt wurden, auf geeignete Werte.

12 Störungsbehebung

Störung	Mögliche Ursachen	Behebung
Versteller bewegt sich nicht	Kabel nicht korrekt angeschlossen	➤ Prüfen Sie die Kabelanschlüsse.
	Versteller oder Verstellerskabel defekt	➤ Wenn vorhanden, tauschen Sie den defekten Versteller gegen einen anderen Versteller aus und testen Sie die neue Kombination.
	Versteller nicht an Netzteil angeschlossen	Versteller mit integriertem PWM-Verstärker werden über ein separates Netzteil versorgt. ➤ Wenn der Versteller über einen integrierten PWM-Verstärker verfügt, schließen Sie ihn an ein geeignetes Netzteil an. ➤ Stellen Sie sicher, dass das Netzteil ordnungsgemäß arbeitet.
	Signallogik der Endschalter falsch eingestellt	Damit sich der Versteller bewegen kann, müssen alle Einstellungen des C-863 mit dem Endschalter-Logikpegel des Verstellers übereinstimmen; siehe "Endschaltererkennung" (S. 34). ➤ Stellen Sie DIP-Schalter 7 passend ein; siehe "Logikpegel der Endschalter" (S. 76). ➤ Stellen Sie den Parameter Limit Mode (ID 0x18) passend ein; siehe "Parameter ändern: Generelle Vorgehensweise (S. 282)".
	Endschaltersignale nicht kompatibel mit dem C-863	Versteller von Drittanbietern verwenden möglicherweise ungeeignete Endschaltersignale. ➤ Wenden Sie sich an den Kundendienst (S. 309) und an den Hersteller des Verstellers.
	Falsche Konfiguration	➤ Prüfen Sie die Parametereinstellungen des C-863 mit den Befehlen SPA? (flüchtiger Speicher) und SEP? (permanenter Speicher). Details zu Parametereinstellungen siehe "Anpassen von Einstellungen" (S. 279).
	Falscher Befehl oder falsche Syntax	➤ Senden Sie den Befehl ERR? und prüfen Sie den zurückgemeldeten Fehlercode.

Störung	Mögliche Ursachen	Behebung
	Falsche Achse kommandiert	Auch bei Systemen mit nur einer Achse ist in Befehlen eine Achsenkennung notwendig. ➤ Stellen Sie sicher, dass die richtige Achsenkennung verwendet wird und dass die kommandierte Achse zum richtigen Versteller gehört.
	Joystick-Steuerung ist aktiv	Bewegungsbefehle sind nicht zulässig, wenn ein Joystick für die Achse aktiv ist. ➤ Deaktivieren Sie den Joystick mit dem Befehl JON (S. 214).
Versteller führt unbeabsichtigte Bewegung aus	Joystick ist nicht angeschlossen, aber im C-863 aktiviert	➤ Aktivieren Sie den Joystick in der Software nur, wenn tatsächlich ein Joystick an den C-863 angeschlossen ist (S. 123).
	Joystick nicht kalibriert	➤ Kalibrieren Sie den Joystick (S. 125).
	Startup-Makro wird ausgeführt	➤ Prüfen Sie, ob ein Makro als Startup-Makro festgelegt ist, und heben Sie die Auswahl des Startup-Makros gegebenenfalls auf (S. 133).
	Bremse des Verstellers bei ausgeschaltetem Servomodus mit dem Befehl BRA (S. 174) deaktiviert	➤ Sichern Sie den Versteller gegen unbeabsichtigte Bewegungen, bevor Sie die Bremse per Befehl deaktivieren!
Versteller schwingt oder wird ungenau positioniert	Die Last wurde geändert.	➤ Stellen Sie das System der Laständerung entsprechend neu ein (S. 96).

Störung	Mögliche Ursachen	Behebung
Versteller schwingt bereits während der Referenzfahrt	Sehr hohe Last auf dem Versteller	Gehen Sie bei einer sehr hohen Last während der Referenzfahrt mit PIMikroMove® wie folgt vor: 1. Starten Sie die Referenzfahrt nicht im Schritt Start up axes, sondern klicken Sie auf Close, um das Fenster Start up controller zu schließen. 2. Öffnen Sie im Hauptfenster das Einzelachsen-Fenster für den angeschlossenen Versteller, indem Sie den Versteller im Menü View > Single Axis Window auswählen. 3. Erweitern Sie die Ansicht des Einzelachsen-Fensters durch Anklicken der Schaltfläche > am rechten Fensterrand. 4. Stellen Sie mit dem Kontrollkästchen Servo sicher, dass der Servomodus eingeschaltet ist. 5. Starten Sie die Referenzfahrt durch Klicken auf eine der Schaltflächen Reference... 6. Wenn der Versteller schwingt: Halten Sie die Referenzfahrt im Dialog Reference Axes sofort an, schließen Sie den Dialog und schalten Sie den Servomodus aus, indem Sie den Haken aus dem entsprechenden Kontrollkästchen im Einzelachsen-Fenster entfernen. 7. Geben Sie neue Werte für die Regelparameter ein, siehe "Regelparameter optimieren" (S. 96). 8. Starten Sie die Referenzfahrt erneut. 9. Wenn der Versteller immer noch schwingt, wiederholen Sie die Schritte 6 bis 8 so lange, bis die Referenzfahrt ohne Schwingungen erfolgreich beendet wird.

Störung	Mögliche Ursachen	Behebung
Kommunikation mit dem Controller funktioniert nicht	Falsches Kommunikationskabel wird verwendet oder es ist defekt	➤ Verwenden Sie für die RS-232-Verbindung ein Nullmodemkabel (S. 65). ➤ Prüfen Sie gegebenenfalls, ob das Kabel an einem fehlerfreien System funktioniert.
	USB-Treiber nicht installiert	Um die Kommunikation zwischen C-863 und PC über USB-Schnittstelle herstellen zu können, müssen Treiber von der Produkt-CD installiert werden. Wenn die Kommunikation über USB nicht hergestellt werden kann oder das PC-Betriebssystem meldet, dass neue Hardware gefunden wurde: 1. Melden Sie sich am PC mit Administratorrechten an. 2. Legen Sie die Produkt-CD ins Laufwerk des PC ein. 3. Folgen Sie den Anweisungen am PC-Bildschirm oder öffnen Sie auf geeignete Weise das Fenster Eigenschaften von PI C-863 Controller. 4. Wählen Sie die Treiber im Verzeichnis \USB Driver auf der Produkt-CD aus, wenn Sie dazu aufgefordert werden. Wenn der C-863 nach erfolgreicher Treiberinstallation immer noch nicht vom PC erkannt wird: ➤ Ziehen Sie das USB-Kabel vom C-863 ab und stecken Sie es wieder ein.
	Baudrate nicht richtig konfiguriert	➤ Überprüfen Sie die Einstellungen der DIP-Schalter 5 und 6 für die Baudrate (S. 76). ➤ Stellen Sie in einem Daisy-Chain-Netzwerk sicher, dass für alle Controller die gleiche Baudrate eingestellt ist.
	Controlleradresse nicht richtig konfiguriert	➤ Überprüfen Sie die Einstellungen der DIP-Schalter 1 bis 4 für die Controlleradresse (S. 75).
	Ein anderes Programm greift auf die Schnittstelle zu.	➤ Schließen Sie das andere Programm.

Störung	Mögliche Ursachen	Behebung
	Probleme mit spezieller Software	➤ Prüfen Sie, ob das System mit einer anderen Software, wie z. B. einem Terminal-Programm oder einer Entwicklungsumgebung, funktioniert. Sie können die Kommunikation testen, indem Sie ein Terminal-Programm (z. B. PI Terminal) starten und *IDN? oder HLP? eingeben. ➤ Achten Sie darauf, dass Sie Befehle mit einem LF (line feed) abschließen. Ein Befehl wird erst ausgeführt, wenn der LF empfangen wurde.
Kunden-Software läuft nicht mit den PI-Treibern	Falsche Kombination der Treiberrouinen/Vis	➤ Prüfen Sie, ob das System mit einem Terminal-Programm läuft. Wenn ja: ➤ Lesen Sie die Angaben im zugehörigen Software-Handbuch und vergleichen Sie den Beispielcode auf der Produkt-CD mit Ihrem Programmcode.
LEDs leuchten nicht, obwohl der C-863 eingeschaltet ist	Firmware-Update-Modus eingestellt	1. Schalten Sie den C-863 aus, indem Sie das Netzkabel des Netzteils ziehen. 2. Stellen Sie am C-863 mit dem DIP-Schalter 8 den Normalbetrieb (S. 77) ein (Stellung AUS). 3. Schalten Sie den C-863 ein, indem Sie das Netzkabel des Netzteils mit der Steckdose verbinden.

Wenn die Störung Ihres Systems nicht in der Tabelle angeführt ist oder wenn sie nicht wie beschrieben behoben werden kann, kontaktieren Sie unseren Kundendienst (S. 309).

13 Kundendienst

Wenden Sie sich bei Fragen und Bestellungen an Ihre PI-Vertretung oder schreiben Sie uns eine E-Mail (<mailto:info@pi.ws>).

Geben Sie bei Fragen zu Ihrem System folgende Systeminformationen an:

- Produktcodes und Seriennummern von allen Produkten im System
- Firmwareversion des Controllers (sofern vorhanden)
- Version des Treibers oder der Software (sofern vorhanden)
- PC-Betriebssystem (sofern vorhanden)

Die aktuellen Versionen der Benutzerhandbücher stehen auf unserer Website zum Herunterladen (S. 5) bereit.

14 Technische Daten

In diesem Kapitel

Spezifikationen	311
Systemanforderungen	314
Abmessungen.....	314
Pinbelegung.....	315

14.1 Spezifikationen

14.1.1 Datentabelle

	C-863.11
Funktion	DC-Motorsteuerung, servogeregelt, 1 Kanal
Kanäle	1
Bewegung und Regler	
Reglertyp	PID-Regler, Parameteränderung im Betrieb
Servozykluszeit	50 µs
Profilgenerator	Trapezförmiges Geschwindigkeitsprofil
Encodereingang	A/B-Quadratur TTL-Pegel, einseitig geerdet oder differenziell gem. RS-422; 60 MHz
Blockiererkennung	Automatischer Motorstopp bei Überschreitung eines programmierbaren Positionsfehlers
Endschalter	2 × TTL (Polarität programmierbar)
Referenzschalter	1 × TTL
Motorbremse	1 × TTL, per Software schaltbar
Elektrische Eigenschaften	
Max. Ausgangsspannung*	0 bis ±15 V für direkte Ansteuerung DC-Motor
Max. Ausgangsleistung	30 W
Strombegrenzung	2 A

Schnittstellen und Bedienung	
Schnittstelle / Kommunikation	USB; RS-232, D-Sub 9-pol. (m)
Motoranschluss	D-Sub 15-pol. (w)
Controllernetzwerk	Bis zu 16 Einheiten** an einer Schnittstelle
I/O-Leitungen	4 analoge/digitale Eingänge, 4 digitale Ausgänge (TTL), 5 V TTL
Befehlssatz	PI General Command Set (GCS)
Bedienersoftware	PIMikroMove®
Softwaretreiber	LabVIEW-Treiber, dynamische Bibliotheken für Windows und Linux
Unterstützte Funktionen	Punkt-zu-Punkt-Bewegung, Startup-Makro, Datenrecorder zur Aufnahme von Betriebsgrößen wie Motorspannung, Geschwindigkeit, Position oder Positionsfehler; interne Sicherheitsschaltung: Watchdog Timer
Manuelle Bedienhilfe	Optional: Pushbutton-Box, Joystick (für 2 Achsen), Y-Kabel für 2D-Bewegungen
Umgebung	
Betriebsspannung	15 bis 30 V, im Lieferumfang: externes Netzteil 15 V / 2 A
Max. Stromaufnahme	80 mA plus Motorstrom (max. 3 A)
Betriebstemperaturbereich	5 bis 50 °C
Masse	0,3 kg
Abmessungen	130 mm × 76 mm × 40 mm

* Die Ausgangsspannung ist abhängig vom verwendeten Netzteil.

** 16 Einheiten über USB; 6 Einheiten über RS-232.

14.1.2 Bemessungsdaten

Der C-863 ist für folgende Betriebsgrößen ausgelegt:

Eingang an:	Maximale Betriebsspannung	Betriebsfrequenz	Maximale Stromaufnahme
			
Hohlstecker-Buchse	30 V	— — —	3 A

Ausgang an:	Maximale Ausgangsspannung 	Maximaler Ausgangsstrom 	Maximale Ausgabefrequenz 
D-Sub 15-polig (f): Pins 2 und 9	= Betriebsspannung	2,5 A	36 kHz (PWM)
D-Sub 15-polig (f): Pins 3 und 11	5 V TTL	10 mA	36 kHz (PWM)

14.1.3 Umgebungsbedingungen und Klassifizierungen

Folgende Umgebungsbedingungen und Klassifizierungen sind für den C-863 zu beachten:

Einsatzbereich	Nur zur Verwendung in Innenräumen
Maximale Höhe	2000 m
Relative Luftfeuchte	Höchste relative Luftfeuchte 80 % für Temperaturen bis 31 °C Linear abnehmend bis 50 % relativer Luftfeuchte bei 40 °C
Lagertemperatur	0 °C bis 70 °C
Transporttemperatur	–25 °C bis +85 °C
Überspannungskategorie	II
Schutzklasse	I
Verschmutzungsgrad	2
Messkategorie	I
Schutzart gemäß IEC 60529	IP20

14.2 Systemanforderungen

Für den Betrieb des C-863 müssen folgende Systemanforderungen erfüllt sein:

- PC mit Betriebssystem Windows (XP, Vista, 7) oder Linux
- Kommunikationsschnittstelle zum PC:
 - Freier COM-Port am PC
 - oder -
 - USB-A-Buchse am PC
- C-863 mit Netzteil
- Mechanik (Versteller) mit DC-Motor
- RS-232-Nullmodemkabel oder USB-Kabel zur Verbindung des Controllers mit dem PC
- Produkt-CD mit PC-Software

14.3 Abmessungen

Abmessungen in mm

Allgemeintoleranzen nach DIN ISO 2768 - f - H

Rauheit Ra 1.6

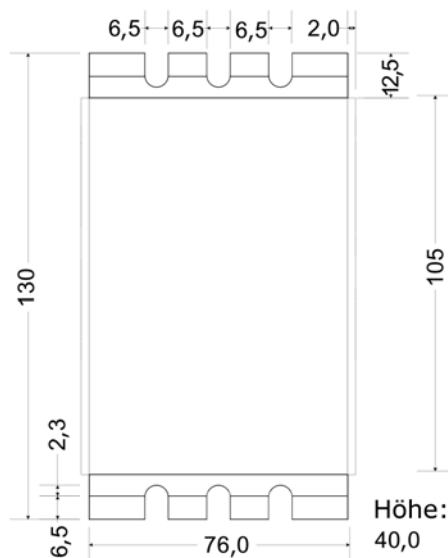
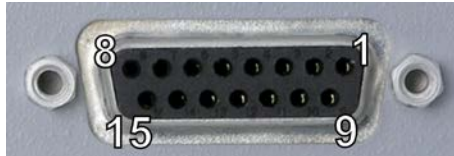


Abbildung 35: C-863, Abmessungen in Millimetern

14.4 Pinbelegung

14.4.1 DC Motor only

D-Sub-Buchse, 15-polig, weiblich



Pin	Funktion
1	Ausgang: Programmierbare Motorbremse (0 oder + 5 V)
2	Ausgang: Motor + (differentiell; Power PWM); für Versteller ohne PWM-Verstärker
3	Ausgang: PWM Magnitude (TTL); für Versteller mit PWM-Verstärker
4	Ausgang: +5 V
5	Eingang: Positiver Endschalter
6	GND Endschalter
7	Eingang: Encoder: A (-)
8	Eingang: Encoder: B (-)
9	Ausgang: Motor – (differentiell; Power PWM); für Versteller ohne PWM-Verstärker
10	Power GND
11	Ausgang: PWM Sign (TTL); für Versteller mit PWM-Verstärker
12	Eingang: Negativer Endschalter
13	Eingang: Referenzschalter
14	Eingang: Encoder: A (+) / ENCA
15	Eingang: Encoder: B (+) / ENCB

14.4.2 I/O

Mini-DIN-Buchse, 9-polig, weiblich



Abbildung 36: Vorderansicht der Mini-DIN-Buchse

Pin	Funktion
1	Eingang 1 (analog: 0 bis +5V/ digital: TTL)
2	Eingang 2 (analog: 0 bis +5V/ digital: TTL)
3	Eingang 3 (analog: 0 bis +5V/ digital: TTL)
4	Eingang 4 (analog: 0 bis +5V/ digital: TTL)
5	Ausgang 1 (digital: TTL)
6	Ausgang 2 (digital: TTL)
7	Ausgang 3 (digital: TTL)
8	Ausgang 4 (digital: TTL)
9	Vcc (+5 V)

14.4.3 Kabel C-170.IO für den Anschluss an die I/O-Buchse

Mini-DIN-Stecker, 9-polig, männlich, offenes Ende

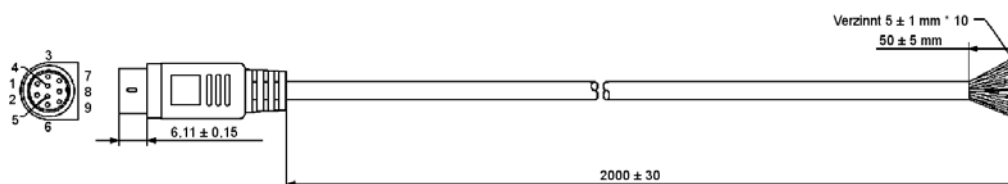


Abbildung 37: C-170.IO-Kabel

Spezifikationen

Temperaturbereich: -25 °C bis +85 °C

Nennstrom: 1 A AC/DC

Isolationswiderstand: 50 MW min.

Nennspannung: 50 V AC/DC

Stoßspannung: 500 V AC für 1 Minute

Pin	Adernfarbe	Funktion an der I/O-Buchse des C-863
1	schwarz	Eingang 1 (analog: 0 bis +5V / digital: TTL)
2	weiß	Eingang 2 (analog: 0 bis +5V / digital: TTL)
3	rot	Eingang 3 (analog: 0 bis +5V / digital: TTL)
4	gelb	Eingang 4 (analog: 0 bis +5V / digital: TTL)
5	violett	Ausgang 1 (digital, TTL)
6	blau	Ausgang 2 (digital, TTL)
7	grün	Ausgang 3 (digital, TTL)
8	braun	Ausgang 4 (digital, TTL)
9	grau	Vcc (+5V)
Mantel	Schirm, schwarz beschichtet (dicker als der an Pin 1 angeschlossene Draht)	GND

14.4.4 Joystick

Mini-DIN-Buchse, 6-polig, weiblich (PS/2)



Abbildung 38: Vorderansicht der Mini-DIN-Buchse

Pin	Funktion
1	GND
2	Eingang: Achse 1 von Joystick-Gerät 2 (-10 bis 10 V)
3	Ausgang: Vcc (3,3 V)
4	Eingang: Achse 1 von Joystick-Gerät 1 (0 bis 3,3 V)
5	Nicht verbunden
6	Eingang: Joystick-Taste 1 von Joystick-Gerät 1 (0 oder 3,3 V)

14.4.5 Kabel C-819.20Y für Joystick C-819.20

Das Kabel C-819.20Y ermöglicht den Anschluss von 2 Controllern an den Joystick C-819.20.



Abbildung 39: Y-Kabel C-819.20Y für Joystick mit 2 Controllern

Mini-DIN-Stecker, 6-polig, weiblich auf 2 Mini-DIN-Stecker, 6-polig, männlich

Mini-DIN 6-polig, weiblich (zum Joystick)	Signal	Mini-DIN 6-polig, männlich, X-Zweig (zu Controller 1)	Mini-DIN 6-polig, männlich, Y-Zweig (zu Controller 2)
Pin 1	GND	Pin 1	Pin 1
Pin 2	Taste für Y-Achse des Joysticks	Nicht verbunden	Pin 6
Pin 3	Spannungsversorgung des Joysticks	Pin 3	Nicht verbunden
Pin 4	X-Achse des Joysticks	Pin 4	Nicht verbunden
Pin 5	Y-Achse des Joysticks	Nicht verbunden	Pin 4
Pin 6	Taste für X-Achse des Joysticks	Pin 6	Nicht verbunden

14.4.6 RS-232 In und RS-232 Out

RS-232 In: D-Sub-Einbaustecker, 9-polig, männlich



RS-232 Out: D-Sub-Buchse, 9-polig, weiblich



Pin	Funktion
1	Nicht verbunden
2	RxD (PC zum Controller)
3	TxD (Controller zum PC)
4	Nicht verbunden
5	GND
6	Nicht verbunden
7	Nicht verbunden
8	Nicht verbunden
9	Nicht verbunden

INFORMATION

Die Pins der Buchsen **RS-232 In** und **RS-232 Out** sind im C-863 1:1 miteinander verbunden.

INFORMATION

In einem Daisy-Chain-Netzwerk, das über die RS-232-Schnittstelle des ersten Controllers an den PC angeschlossen ist, speist nur der PC die RxD-Leitung. In Abhängigkeit davon, wie leistungsfähig der RS-232-Treiber des PC ist, kann daher der Umfang des Netzwerks auf 6 Geräte beschränkt sein.

INFORMATION

Der C-863 kopiert alle Signale, die er vom PC über USB erhält, auf die RxD-Leitung der Buchsen **RS-232 In** und **RS-232 Out**. Das Signal der TxD-Leitung kopiert der C-863 über USB zum PC.

14.4.7 Netzteilanschluss 15-30 VDC**Hohlstecker-Buchse**

Pin	Funktion
Mittelstift	Eingang: 15 bis 30 V DC
Außenleiter	GND (Power)

15 Altgerät entsorgen

Nach geltendem EU-Recht dürfen Elektrogeräte in den Mitgliedsstaaten der EU nicht über den kommunalen Restmüll entsorgt werden.

Entsorgen Sie das Altgerät unter Beachtung der internationalen, nationalen und regionalen Richtlinien.

Um der Produktverantwortung als Hersteller gerecht zu werden, übernimmt die Physik Instrumente (PI) GmbH & Co. KG kostenfrei die umweltgerechte Entsorgung eines PI-Altgerätes, sofern es nach dem 13. August 2005 in Verkehr gebracht wurde.

Falls Sie ein solches Altgerät von PI besitzen, können Sie es versandkostenfrei an folgende Adresse senden:

Physik Instrumente (PI) GmbH & Co. KG
Auf der Römerstr. 1
D-76228 Karlsruhe



16 EG-Konformitätserklärung

PI

Konformitätserklärung

gemäß DIN EN ISO/IEC 17050-1:2005

Hersteller:	Physik Instrumente (PI) GmbH & Co. KG	
Herstelleradresse:	Auf der Römerstraße 1 D-76228 Karlsruhe	

Der Hersteller erklärt hiermit, dass das Produkt

Produktbezeichnung: Mercury DC-Motor Controller, 1 Kanal

Modellnummer(n): C-863

Produktausführungen: alle

die folgenden europäischen Richtlinien erfüllt:

2004/108/EG, EMV-Richtlinie

2011/65/EG, RoHS-Richtlinie

Die zum Nachweis der Konformität zugrundegelegten Normen sind nachfolgend aufgelistet.

Elektromagnetische Emission: EN 61000-6-3:2007, EN 55011:2009

Elektromagnetische Störfestigkeit: EN 61000-6-1:2007

20. November 2012
Karlsruhe


Norbert Ludwig
Geschäftsführer

Physik Instrumente (PI) GmbH & Co. KG, Auf der Römerstraße 1, 76228 Karlsruhe, Germany
Telefon +49 721 4846-0, Telefax +49 721 4846-1019, E-Mail info@pi.ws, www.pi.ws

PIEZO NANO POSITIONING
