

# MS232D C-877 PILine® Controller

#### Benutzerhandbuch

Version: 2.0.0 Datum: 24.10.2024



#### **Dieses Dokument beschreibt folgendes Produkt:**

C-877.1U11
 Kompakter, preisgünstiger
 Piezomotorcontroller / Treiber, 1 Achse, für
 PILine® Systeme mit geringer
 Leistungsaufnahme

Physik Instrumente (PI) SE & Co. KG, Auf der Römerstr. 1, 76228 Karlsruhe, Deutschland Tel. +49 721 4846-0, Fax +49 721 4846-1019, E-Mail info@pi.de, www.pi.de

# $\mathbf{PI}$

Die nachfolgend aufgeführten Marken sind geistiges Eigentum der Physik Instrumente (PI) SE & Co. KG ("PI") und im Markenregister des Deutschen Patent- und Markenamts sowie teilweise auch in anderen Markenregistern unter dem Firmennamen Physik Instrumente (PI) GmbH & Co. KG eingetragen: PI®, PIC®, PICMA®, PILine®, PIFOC®, PiezoWalk®, NEXACT®, NEXLINE®, PInano®, NanoCube®, Picoactuator®, PicoCube®, PIMikroMove®, PIMag®, PIHera®

Hinweise zu Markennamen und Warenzeichen Dritter:

Microsoft® und Windows® sind eingetragene Warenzeichen oder Warenzeichen der Microsoft Corporation in den USA und/oder anderen Ländern.

EtherCAT® ist eine eingetragene Marke und patentierte Technologie lizensiert durch die Beckhoff Automation GmbH. Deutschland.

TwinCAT® ist eine eingetragene und lizenzierte Marke der Beckhoff Automation GmbH.

LabVIEW, National Instruments und NI sind Warenzeichen von National Instruments. Weder die Treibersoftware noch von PI angebotene Softwareprogramme oder andere Waren und Dienstleistungen sind verbunden mit oder gefördert von National Instruments.

Python® ist ein eingetragenes Warenzeichen der Python Software Foundation.

BiSS ist ein Warenzeichen der iC-Haus GmbH.

Bei den nachfolgend aufgeführten Bezeichnungen handelt es sich um geschützte Firmennamen, Warenzeichen oder eingetragene Warenzeichen fremder Inhaber:

Linux, MATLAB, MathWorks, FTDI

Die Verwendung dieser Bezeichnungen dient ausschließlich Identifizierungszwecken.

Von PI zur Verfügung gestellte Softwareprodukte unterliegen den Allgemeinen Softwarelizenzbestimmungen der Physik Instrumente (PI) SE & Co. KG und können Drittanbieter-Softwarekomponenten beinhalten und/oder verwenden. Weitere Informationen finden Sie in den Allgemeinen Softwarelizenzbestimmungen (https://www.physikinstrumente.com/fileadmin/user\_upload/physik\_instrumente/files/legal/General-Software-License-Agreement-Physik-Instrumente.pdf) und in den Drittanbieter-Softwarehinweisen (https://www.physikinstrumente.com/fileadmin/user\_upload/physik\_instrumente/files/legal/Third-Party-Software-Note-Physik-Instrumente.pdf) auf unserer Webseite.

© 2024 Physik Instrumente (PI) SE & Co. KG, Karlsruhe, Deutschland. Die Texte, Bilder und Zeichnungen dieses Handbuchs sind urheberrechtlich geschützt. Physik Instrumente (PI) SE & 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: 24.10.2024

Dokumentnummer: MS232D, ASt, Version 2.0.0

Änderungen vorbehalten. Dieses Handbuch verliert seine Gültigkeit mit Erscheinen einer neuen Revision. Die jeweils aktuelle Revision ist auf unserer Webseite (https://www.physikinstrumente.de/de/) zum Herunterladen verfügbar.



# Inhalt

1	Ube	r dieses Dokument	1
	1.1	Ziel und Zielgruppe dieses Benutzerhandbuchs	
	1.2	Symbole und Kennzeichnungen	
	1.3	Begriffserklärung	
	1.4	Abbildungen	
	1.5	Mitgeltende Dokumente	
	1.6	Handbücher herunterladen	
2	Sich	erheit	5
	2.1	Bestimmungsgemäße Verwendung	5
	2.2	Allgemeine Sicherheitshinweise	
	2.3	Organisatorische Maßnahmen	
	2.4	Europäische Konformitätserklärungen	
3	Proc	duktbeschreibung	7
	3.1	Produktansicht	7
		3.1.1 Vorderwand	
		3.1.2 Rückwand	
		3.1.3 Typenschild	
	3.2	Lieferumfang	9
	3.3	PC-Softwareübersicht	
		3.3.1 PI Software Suite	
	3.4	Positioniererdatenbanken	11
	3.5	ID-Chip-Erkennung	12
	3.6	Kommunikationsschnittstellen	
	3.7	Funktionsprinzipien	
		3.7.1 Blockdiagramm	
		3.7.2 Kommandierbare Elemente	
		3.7.3 Wichtige Komponenten der Firmware	
		3.7.4 Betriebsarten	
		3.7.5 Physikalische Einheiten	16
		3.7.6 Auslösen von Bewegungen	17
		3.7.7 Erzeugung des Dynamikprofils	18
		3.7.8 Regelalgorithmus und weitere Stellwertkorrekturen	21
		3.7.9 Optionale Zweiphasenansteuerung	26
		3.7.10 On-Target-Status	
		3.7.11 Unterstützte Motortypen	
		3.7.12 Automatische Frequenzregelung	
		3.7.13 Referenzschaltererkennung	
		3.7.14 Endschaltererkennung	
		3.7.15 Stellweg und Verfahrbereichsgrenzen	
		3.7.16 Referenzierung	

4	Aus	packen	39
5	Insta	allation	41
	5.1	Allgemeine Hinweise zur Installation	41
	5.2	C-877 montieren	
	5.3	C-877 an Schutzleiter anschließen	42
	5.4	Netzteil an C-877 anschließen	
	5.5	Positionierer anschließen	
	5.6	PC-Software installieren	
		5.6.1 Erstinstallation ausführen	
		5.6.2 Updates installieren	
		·	
	5.7	PC anschließen	
		5.7.1 C-877 über die USB-Schnittstelle anschließen	47
6	Inbe	triebnahme	49
	6.1	Allgemeine Hinweise zur Inbetriebnahme	49
	6.2	C-877 einschalten	
	6.3	Kommunikation herstellen	
		6.3.1 Kommunikation über USB-Schnittstelle herstellen	50
	6.4	Bewegungen starten	
	6.5	Regelparameter optimieren	56
7	Betr	ieb	63
	7.1	Schutzfunktionen des C-877	63
		7.1.1 Schutz vor Überhitzung	63
		7.1.2 Verhalten bei Bewegungsfehler	
		7.1.3 Verhalten bei Systemfehler	
		7.1.4 Betriebsbereitschaft wiederherstellen	
	7.2	Datenrekorder	
		7.2.1 Datenrekorder einrichten	
		7.2.2 Aufzeichnung starten	
		7.2.3 Aufgezeichnete Daten auslesen	
	7.3	Controllermakros	
		7.3.1 Übersicht: Makrofunktionalitäten und Beispielmakros	
		7.3.2 Befehle und Parameter für Makros	
		7.3.3 Mit Makros arbeiten	
		7.5.4 Controllermation stellerm and lader	
8	GCS-	Befehle	77
	8.1	Schreibweise	
	8.2	GCS-Syntax für Syntaxversion 2.0	
	8.3	Empfänger- und Senderadresse	
	8.4	Variablen	
	8.5	Befehlsübersicht	82

	8.6 8.7	Befehlsbeschreibungen für GCS 2.0 Fehlercodes	
	0.7	Terrer codes	144
9	Anpa	ssen von Einstellungen	167
	9.1	Einstellungen des C-877	
	9.2	Parameterwerte im C-877 ändern	
		9.2.1 Allgemeine Befehle für Parameter	
		9.2.2 Befehle für Schnellzugriff auf einzelne Parameter	
		9.2.4 Parameterwerte ändern: Generelle Vorgehensweise	
	9.3	Positionierertyp anlegen oder ändern	172
	9.4	Parameterübersicht	176
10	Wart	cung	187
	10.1	C-877 reinigen	187
	10.2	Firmware aktualisieren	187
11	Störu	ungsbehebung	191
12	Kund	lendienst	195
13	Tech	nische Daten	197
	13.1	Spezifikationen	197
		13.1.1 Datentabelle	197
		13.1.2 Bemessungsdaten	
		13.1.3 Umgebungsbedingungen und Klassifizierungen	198
	13.2	Abmessungen	199
	13.3	Pinbelegung	
		13.3.1 Motor und Sensor	
		13.3.2 Netzteilanschlus	200
14	Altge	erät entsorgen	201



# 1 Über dieses Dokument

# 1.1 Ziel und Zielgruppe dieses Benutzerhandbuchs

Dieses Benutzerhandbuch enthält die erforderlichen Informationen für die bestimmungsgemäße Verwendung des C-877.

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. 4) 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
RS-232	Bedienelement-Beschriftung auf dem Produkt (Beispiel: Buchse der RS-232 Schnittstelle)
<u>^</u>	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 müssen nacheinander die Menüeinträge <b>Start</b> und <b>Einstellungen</b> gewählt werden)
POS?	Befehlszeile oder Befehl aus dem universellen Befehlssatz GCS von PI (Beispiel: Befehl zum Abfragen der aktuellen Achsenposition)
Device S/N	Parameterbezeichnung (Beispiel: Parameter, in dem die Seriennummer gespeichert ist)



Symbol/ Kennzeichnung **Bedeutung** 

5

Wert, der über die PC-Software eingegeben bzw. ausgewählt werden muss

# 1.3 Begriffserklärung

Begriff	Erklärung
Achse	Auch als "logische Achse" bezeichnet. Die logische Achse bildet die Bewegung der Mechanik in der Firmware des C-877 ab. Bei Mechaniken, die Bewegung in mehreren Richtungen erlauben (z. B. in X, Y und Z), entspricht jede Bewegungsrichtung einer logischen Achse.
Positionierer	An den C-877 angeschlossene Mechanik. Bei Positionierern mit nur einer Bewegungsachse ist die Bezeichnung "Achse" gleichbedeutend mit "Positionierer". Positionierer, die Bewegung in mehreren Achsen erlauben, werden auch als "mehrachsige Positionierer" bezeichnet. Für diese Positionierer muss zwischen den einzelnen Achsen unterschieden werden.
Stellwert	Der Stellwert ist die Eingangsgröße für die PILine® Treiberelektronik des C-877. Die Treiberelektronik wandelt den Stellwert in die Piezospannung für die Achse des Positionierers um.
Zweiphasen- ansteuerung	Für die positive und negative Bewegungsrichtung hat ein PILine® Piezomotor jeweils ein separates Piezosegment. Je nach Bewegungsrichtung wird normalerweise nur das entsprechende Segment mit der Piezospannung angesteuert. Mit der Zweiphasenansteuerung kann in festgelegten Zeitintervallen parallel zum ersten Segment auch das zweite Segment angesteuert werden, um den Vortrieb des Piezomotors für die Dauer des Intervalls zu unterbrechen. Die intervallweise Unterbrechung des Vortriebs kann das Einschwingverhalten der Achse je nach Anwendung verbessern.
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 Referenzierung durchgeführt werden, bevor absolute Zielpositionen kommandiert und erreicht werden können.
Dynamikprofil	Umfasst die vom Profilgenerator des C-877 für jeden Zeitpunkt der Bewegung berechnete Zielposition, Geschwindigkeit und Beschleunigung der Achse. Die errechneten Werte werden "kommandierte Werte" genannt.
Flüchtiger Speicher	RAM-Baustein, in dem bei eingeschaltetem Controller die Parameter gespeichert sind (Arbeitsspeicher). Die Parameterwerte im flüchtigen Speicher bestimmen das aktuelle Verhalten des Systems. In der PC-Software von PI werden die Parameterwerte im flüchtigen Speicher auch als "Active Values" bezeichnet.



Begriff	Erklärung
Permanenter Speicher	Speicherbaustein (Festspeicher, z. B. EEPROM oder Flash-Speicher), von dem beim Start des Controllers die Standardwerte der Parameter in den flüchtigen Speicher geladen werden. In der PC-Software von PI werden die Parameterwerte im permanenten Speicher auch als "Startup Values" bezeichnet.
Firmware	Software, die auf dem Controller installiert ist.
PC-Software	Software, die auf dem PC installiert wird.
GCS	PI General Command Set: Befehlssatz für Controller von PI

# 1.4 Abbildungen

Zugunsten eines besseren Verständnisses können Farbgebung, Größenverhältnisse und Detaillierungsgrad in Illustrationen von den tatsächlichen Gegebenheiten abweichen. Auch fotografische Abbildungen können abweichen und stellen keine zugesicherten Eigenschaften dar.

# 1.5 Mitgeltende Dokumente

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

Beschreibung	Dokument
Kurzanleitung zur Installation und Inbetriebnahme des C-877.1U11	MS242EK Kurzanleitung für digitale Motorcontroller
PI MATLAB Treiber GCS 2.0	SM155D Software-Handbuch
PI GCS 2.0 DLL0	SM151E Software Manual
GCS Array: Datenformatbeschreibung	SM146E Software Manual
PIMikroMove®	SM148E Software Manual
PIStageEditor: Software zur Verwaltung von Positioniererdatenbanken	SM144E Software Manual
PI Update Finder: Updates suchen und herunterladen	A000T0028 Benutzerhandbuch
PI Software on ARM-Based Platforms	A000T0089 Technical Note
Downloading manuals from PI: PDF-Datei mit Links auf die Handbücher für digitale Elektroniken und Software von PI. Wird zusammen mit der PI-Software ausgeliefert.	A000T0081 Technical Note

Die aktuellen Versionen der Benutzerhandbücher stehen auf unserer Website zum Herunterladen (S. 4) bereit.



## 1.6 Handbücher herunterladen

#### **INFORMATION**

Wenn ein Handbuch fehlt oder Probleme beim Herunterladen auftreten:

Wenden Sie sich an unseren Kundendienst (S. 195).

#### Handbücher herunterladen

- 1. Öffnen Sie die Website www.pi.de.
- 2. Suchen Sie auf der Website nach der Produktnummer (z. B. C-877).
- 3. Um die Produktdetailseite zu öffnen, wählen Sie in den Suchergebnissen das Produkt.
- 4. Wählen Sie Downloads.
  - Die Handbücher werden unter **Dokumentation** angezeigt. Softwarehandbücher werden unter **Allgemeine Software-Dokumentation** angezeigt.
- 5. Wählen Sie für das gewünschte Handbuch HINZUFÜGEN und dann ANFORDERN.
- Füllen Sie das Anfrageformular aus und wählen Sie ANFRAGE SENDEN.
   Der Download-Link wird an die eingegebene E-Mail-Adresse gesendet.



# 2 Sicherheit

# 2.1 Bestimmungsgemäße Verwendung

Der C-877 ist ein Laborgerät im Sinne der DIN EN 61010-1. Er ist für die Verwendung in Innenräumen und in einer Umgebung vorgesehen, die frei von Schmutz, Öl und Schmiermitteln ist.

Entsprechend seiner Bauform ist der C-877 für den Betrieb von Positionierern mit PILine® Ultraschall-Piezomotoren und Stecker D-Sub 15 (m) vorgesehen.

Der C-877 ist für den geregelten Betrieb mit inkrementellen Positionssensoren vorgesehen. Außerdem kann er die Referenz- und Endschaltersignale des angeschlossenen Positionierers auslesen und weiterverarbeiten.

Der C-877 darf nur unter Einhaltung der technischen Spezifikationen und Anweisungen in diesem Benutzerhandbuch verwendet werden. Für die Prozessvalidierung ist der Benutzer verantwortlich.

Der C-877 darf nicht für andere als die in diesem Benutzerhandbuch genannten Zwecke verwendet werden. Insbesondere darf der C-877 nicht für den Antrieb von ohmschen oder induktiven Lasten verwendet werden.

# 2.2 Allgemeine Sicherheitshinweise

Der C-877 ist nach dem Stand der Technik und den anerkannten sicherheitstechnischen Regeln gebaut. Bei unsachgemäßer Verwendung des C-877 können Benutzer gefährdet werden und/oder Schäden am C-877 entstehen.

- Benutzen Sie den C-877 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-877 verantwortlich.

- Installieren Sie den C-877 in der Nähe der Stromversorgung, damit der Netzstecker schnell und einfach vom Netz getrennt werden kann.
- Verwenden Sie zum Anschließen des C-877 an die Stromversorgung die mitgelieferten Komponenten (Netzteil, Adapter, Netzkabel).
- Wenn eine der mitgelieferten Komponenten für das Anschließen an die Stromversorgung ersetzt werden muss, verwenden Sie eine ausreichend bemessene Komponente.



# 2.3 Organisatorische Maßnahmen

#### **Benutzerhandbuch**

- ➤ Halten Sie dieses Benutzerhandbuch ständig am C-877 verfügbar.

  Die aktuellen Versionen der Benutzerhandbücher stehen auf unserer Website zum Herunterladen (S. 4) bereit.
- Fügen Sie alle vom Hersteller bereitgestellten Informationen, z. B. Ergänzungen und Technical Notes, zum Benutzerhandbuch hinzu.
- Wenn Sie den C-877 an Dritte weitergeben, fügen Sie dieses Handbuch und alle sonstigen vom Hersteller bereitgestellten Informationen bei.
- Führen Sie Arbeiten grundsätzlich anhand des vollständigen Benutzerhandbuchs durch. Fehlende Informationen aufgrund eines unvollständigen Benutzerhandbuchs können zu leichten Verletzungen und zu Sachschäden führen.
- Installieren und bedienen Sie den C-877 nur, nachdem Sie dieses Benutzerhandbuch gelesen und verstanden haben.

## Personalqualifikation

Nur autorisiertes und entsprechend qualifiziertes Personal darf den C-877 installieren, in Betrieb nehmen, bedienen, warten und reinigen.

# 2.4 Europäische Konformitätserklärungen

Für den C-877 wurden Konformitätserklärungen gemäß den folgenden europäischen gesetzlichen Anforderungen ausgestellt:

EMV-Richtlinie

RoHS-Richtlinie

Die zum Nachweis der Konformität zugrunde gelegten Normen sind nachfolgend aufgelistet.

EMV: EN 61326-1

Sicherheit: EN 61010-1 RoHS: EN IEC 63000



# 3 Produktbeschreibung

# 3.1 Produktansicht

# 3.1.1 Vorderwand



Abbildung 1: Vorderwand des C-877.1U11

Beschriftung	Тур	Funktion
<b>24 V</b> ==== 0.8 A	Hohlstecker-Buchse (S. 200)	Anschluss für die Versorgungsspannung
STA	LED grün	<ul> <li>Controllerstatus:</li> <li>An: C-877 ist bereit für den Normalbetrieb</li> <li>Aus: C-877 ist nicht an der Versorgungsspannung angeschlossen oder befindet sich im Firmware-Update-Modus</li> </ul>
ERR	LED rot	Fehleranzeige:  An: Fehler (Fehlercode ≠ 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.
	Gewindebolzen mit Befestigungsmaterial für Schutzleiter	Schutzleiteranschluss (S. 42)  Der Gewindebolzen muss an einen Schutzleiter angeschlossen werden, da der C-877 nicht über den Netzteil-Anschluss geerdet ist.
<b>●</b> ✓-•	Mini-USB Typ B	Universal Serial Bus für Verbindung zum PC



# 3.1.2 Rückwand



Abbildung 2: Rückwand des C-877.1U11

Beschriftung	Тур	Funktion	
Motor A ~71 V <sub>eff</sub>	D-Sub 15 (f) (S. 200)	Anschluss des Positionierers. Nur für PILine® Ultraschall-Piezomotoren!	
- /I Ven -		<ul> <li>Ausgänge für die Piezospannung</li> </ul>	
		<ul> <li>Eingang der Signale des Positionssensors</li> </ul>	
		<ul> <li>Eingang der Signale der Endschalter und des Referenzschalters</li> </ul>	
		<ul> <li>Ausgabe der Versorgungsspannung für Positionssensor, Referenz- und Endschalter</li> </ul>	
		<ul><li>Eingang für Signale des ID-Chips</li></ul>	

# 3.1.3 Typenschild

Beschriftung	Funktion
	DataMatrix-Code (Beispiel; enthält die Seriennummer)
C-877.1U11	Produktbezeichnung
PI	Herstellerlogo
116056789	Seriennummer (Beispiel), individuell für jeden C-877 Bedeutung der Stellen (Zählung von links): 1 = interne Information, 2 und 3 = Herstellungsjahr, 4 bis 9 = fortlaufende Nummer
Country of origin: Germany	Herkunftsland
$\triangle$	Warnzeichen "Handbuch beachten!"
<u>A</u>	Altgeräteentsorgung (S. 201)
CE	Konformitätszeichen CE
WWW.PI.WS	Herstelleradresse (Website)



# 3.2 Lieferumfang

Artikel	Komponente
C-877.1U11	PILine® Piezomotorcontroller
C-501.24050H	Weitbereichsnetzteil 24 V 50 W, Hohlstecker
3763	Netzkabel
000036360	USB-Kabel (Typ A auf Mini-B) zur Verbindung mit dem PC
C-990.CD1	Datenträger mit PC-Software von PI
MS242EK	Kurzanleitung für digitale Motorcontroller

# 3.3 PC-Softwareübersicht

#### 3.3.1 PI Software Suite

Im Lieferumfang (S. 9) des C-877 ist ein Datenträger mit der PI Software Suite enthalten. Einige Bestandteile der PI Software Suite werden in der nachfolgenden Tabelle beschrieben. Informationen zur Kompatibilität der Software mit PC-Betriebssystemen finden Sie in der C-990.CD1 Release News im Hauptverzeichnis des Datenträgers.

#### Bibliotheken, Treiber

PC-Software	Kurzbeschreibung	Empfohlene Verwendung	
Dynamische Programmbibliothek für GCS	Ermöglicht die Software-Programmierung für den C-877 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 NI LabVIEW-Treiber benötigt.	
Treiber zur Verwendung mit NI LabVIEW Software	NI LabVIEW ist eine Software für die Datenerfassung und Prozesssteuerung (von National Instruments separat zu beziehen). Die Treiberbibliothek ist eine Sammlung von Virtual-Instrument-Treibern für Elektroniken von PI. Die Treiber unterstützen das GCS.	Für Anwender, die NI LabVIEW zur Programmierung ihrer Anwendung verwenden möchten.	
MATLAB-Treiber	MATLAB ist eine Entwicklungsumgebung und Programmiersprache für numerische Berechnungen (von MathWorks separat zu	Für Anwender, die MATLAB zur Programmierung ihrer Anwendung verwenden	



PC-Software	Kurzbeschreibung	Empfohlene Verwendung
	beziehen). Der PI MATLAB-Treiber besteht aus einer MATLAB-Klasse, die in jedes beliebige MATLAB-Skript eingebunden werden kann. Diese Klasse unterstützt das GCS. Der PI MATLAB-Treiber benötigt keine zusätzlichen MATLAB-Toolboxen.	möchten.
USB-Treiber	Treiber für die USB-Schnittstelle	Für Anwender, die den Controller über die USB- Schnittstelle an den PC anschließen möchten.

# **Anwender-Software**

PC-Software	Kurzbeschreibung	Empfohlene Verwendung
PIMikroMove®	Grafische Benutzerschnittstelle für Windows, mit der der C-877 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) Komplette Umgebung für die Befehlseingabe, zum Ausprobieren von verschiedenen Befehlen PIMikroMove® verwendet die dynamische Programmbibliothek zur Kommandierung des Controllers.	Für Anwender, die einfache Automatisierungsaufgab en 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.
PITerminal	Terminalprogramm, das für nahezu alle PI- Controller verwendet werden kann.	Für Anwender, die die Befehle des GCS direkt an den Controller senden möchten.
PIStages3Editor	Programm zum Öffnen und Editieren von Positioniererdatenbanken des Formats .db.	Für Anwender, die sich intensiv mit den Inhalten der Positioniererdatenbanke n auseinandersetzen möchten.
PIUpdateFinder	Überprüft die auf dem PC installierte Software von PI. Wenn auf dem PI-Server aktuellere Versionen der PC-Software	Für Anwender, die die PC-Software



PC-Software	Kurzbeschreibung	Empfohlene Verwendung
	vorhanden sind, wird das Herunterladen angeboten.	aktualisieren möchten.
PIFirmwareManager	Programm zur Unterstützung des Anwenders bei der Aktualisierung der Firmware des C-877.	Für Anwender, die die Firmware aktualisieren möchten.

## 3.4 Positioniererdatenbanken

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

Dateiname der Datenbank	Editierbar?	Beschreibung
PIMicosStages2.dat	Nein, Updates können von der PI Website heruntergeladen werden (S. 45).	Standard-Positioniererdatenbank: Enthält Parametersätze für alle Standardpositionierer von PI miCos; wird bei der Installation der PC-Software automatisch auf dem PC gespeichert.
PIStages2.dat	Nein, Updates können von der PI Website heruntergeladen werden (S. 45).	Standard-Positioniererdatenbank: Enthält Parametersätze für alle Standardpositionierer von PI; wird bei der Installation der PC-Software automatisch auf dem PC gespeichert.
PI_UserStages2.dat	Ja, neue Parametersätze können angelegt, editiert und gespeichert werden (S. 172).	Wird automatisch angelegt, wenn Sie mit der PC-Software erstmalig eine Verbindung zu Ihrem Positionierer herstellen (d. h. bei der Positioniererauswahl in PIMikroMove® bzw. bei Verwendung der Befehle VST? oder CST aus der dynamischen Programmbibliothek).
X-xxx.dat	Nein, Updates erhalten Sie von unserem Kundendienst (S. 195).	Enthält den Parametersatz für einen kundenspezifischen Positionierer, für die Installation siehe "Kundenspezifische Positioniererdatenbank installieren" (S. 46).

Die Parameterwerte im flüchtigen Speicher des C-877 können abgefragt und in den permanenten Speicher geschrieben werden, siehe "Anpassen von Einstellungen" (S. 167)

#### **INFORMATION**

Positioniererdatenbanken enthalten nur einen Teil der Informationen, die zum Betrieb eines Positionierers mit dem C-877 erforderlich sind. Weitere Informationen werden beim Einschalten oder Neustart des C-877 als Parameterwerte aus dem ID-Chip (S. 12) des

C-877 PILine® Controller MS232D Version: 2.0.0 11



Positionierers in den flüchtigen Speicher des C-877 geladen.

Parameter, die aus einer Positioniererdatenbank oder vom ID-Chip geladen werden, sind in der Parameterübersicht (S. 176) farbig markiert.

## 3.5 ID-Chip-Erkennung

Positionierer mit PILine® Ultraschall-Piezomotoren und D-Sub 15-Stecker enthalten im Stecker einen ID-Chip, auf dem folgende Daten als Parameter gespeichert sind:

 Informationen zum Positionierer: Typ, Seriennummer, Herstellungsdatum, Version der Hardware

Die Daten des angeschlossenen Positionierers werden beim Einschalten (S. 49) oder Neustart des C-877 aus dem ID-Chip in den flüchtigen Speicher des C-877 geladen.

Die Parameterwerte im flüchtigen Speicher des C-877 können abgefragt und in den permanenten Speicher geschrieben werden, siehe "Anpassen von Einstellungen" (S. 167).

#### **INFORMATION**

Der ID-Chip enthält nur einen Teil der Informationen, die zum Betrieb des Positionierers mit dem C-877 erforderlich sind. Wenn Sie die PC-Software von PI verwenden, werden weitere Informationen als Parameterwerte aus einer Positioniererdatenbank (S. 11) in den flüchtigen Speicher des C-877 geladen.

Parameter, die vom ID-Chip oder aus einer Positioniererdatenbank geladen werden, sind in der Parameterübersicht (S. 176) farbig markiert.

#### 3.6 Kommunikationsschnittstellen

Der C-877 kann von einem PC mit ASCII-Befehlen gesteuert werden. Der Anschluss an den PC erfolgt direkt über die USB-Schnittstelle des C-877.

#### **INFORMATION**

Für die USB-Schnittstelle wird im C-877 ein USB-UART-Modul (FTDI) verwendet. Wenn der C-877 über den USB-Anschluss verbunden und eingeschaltet ist, wird die USB-Schnittstelle in der PC-Software deshalb auch als COM-Port angezeigt. Der C-877 verwendet für diese Schnittstelle die Baudrate 115200.



# 3.7 Funktionsprinzipien

# 3.7.1 Blockdiagramm

Der C-877 steuert die Bewegung der logischen Achse eines Positionierers. Das nachfolgende Blockdiagramm zeigt, wie der C-877 die Piezospannung für die verbundene Achse erzeugt:

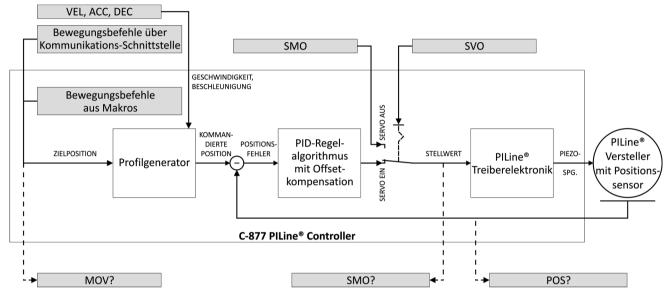


Abbildung 3: C-877: Erzeugung des Stellwerts

Der C-877 unterstützt Positionierer mit PILine® Ultraschall-Piezomotor und inkrementellem Positionssensor.

## 3.7.2 Kommandierbare Elemente

Die folgende Tabelle enthält die mit den Befehlen des GCS (S. 84) kommandierbaren Elemente.

	An- zahl	Kennung	Beschreibung
Logische Achse	1	1 (änderbar)	Die logische Achse bildet die Bewegung des Positionierers in der Firmware des C-877 ab. Sie entspricht einer Achse eines linearen Koordinatensystems.
			In der Firmware des C-877 werden Bewegungen für logische Achsen kommandiert (d. h. für die Bewegungsrichtungen eines Positionierers). Im geregelten Betrieb stehen z. B. die Bewegungsbefehle MOV und MVR zur Verfügung. Bewegungen im ungeregelten Betrieb werden mit SMO ausgelöst.  Die Achsenkennung kann mit dem Befehl SAI? abgefragt und mit dem Befehl SAI geändert werden. Sie kann aus bis

C-877 PILine® Controller MS232D Version: 2.0.0 13



Element	An- zahl	Kennung	Beschreibung
			zu 8 Zeichen bestehen; gültige Zeichen (Anzeige mit dem Befehl TVI?) sind:
			1234567890ABCDEFGHIJKLMNOPQRSTUVWXYZ
			Wenn der Parameter <i>Stage Name</i> (0x3C) den Wert NOSTAGE hat, ist die Achse "deaktiviert". Eine deaktivierte Achse ist nicht für achsenbezogene Befehle zugänglich (z.B. Bewegungsbefehle oder Positionsabfragen). Die Kennung einer deaktivierten Achse kann nur mit SAI? ALL abgefragt werden.
Datenrekorder- tabellen	4	1 bis 4	Der C-877 hat 4 Datenrekordertabellen (Abfrage mit TNR?) mit 1024 Datenpunkten pro Tabelle.
Gesamtsystem	1	1 (nicht änderbar)	C-877 als Gesamtsystem

# 3.7.3 Wichtige Komponenten der Firmware

Die Firmware des C-877 stellt die folgenden funktionalen Einheiten bereit:

Firmware- Komponente	Beschreibung
Parameter	Parameter spiegeln die Eigenschaften des angeschlossenen Positionierers wider (z. B. Stellweg) und bestimmen das Verhalten des C-877 (z. B. Einstellungen für den Regelalgorithmus).
	Die Parameter können in folgende Kategorien eingeteilt werden:
	<ul> <li>Geschützte Parameter, deren Werkseinstellung nicht geändert werden kann</li> </ul>
	<ul> <li>Parameter, die zur Anpassung an die Anwendung vom Benutzer eingestellt werden müssen</li> </ul>
	Weitere Informationen finden Sie im Abschnitt "Anpassen von Einstellungen" (S. 167).
	Bei Positionierern mit ID-Chip sind die Werte einiger Parameter auf dem ID-Chip gespeichert. Sie werden beim Einschalten oder Neustart des C-877 in den flüchtigen Speicher geladen.
Befehlsebenen	Die Befehlsebenen legen das Schreibrecht auf die Parameter fest.  Die aktuelle Befehlsebene kann mit dem Befehl CCL geändert werden.  Dazu kann die Eingabe eines Kennworts erforderlich sein.
ASCII-Befehle (GCS)	Die Kommunikation mit dem C-877 kann mit den Befehlen des PI General Command Set (GCS; Version 2.0) geführt werden. Der GCS ist von der Hardware (Controller, angeschlossene Positionierer) unabhängig. Beispiele für die Verwendung des GCS:
	<ul><li>C-877 konfigurieren</li></ul>
	Betriebsart einstellen



Firmware- Komponente	Beschreibung
	Bewegungen des Positionierers starten
	<ul> <li>System- und Positionswerte abfragen</li> </ul>
	Eine Liste der verfügbaren Befehle finden Sie im Abschnitt "Befehlsübersicht" (S. 81).
Profilgenerator und Regelalgorithmus	Der Profilgenerator führt Berechnungen durch, um die Zielposition, Geschwindigkeit und Beschleunigung der Achse zu jedem Zeitpunkt der Bewegung festzulegen. Der Positionsfehler, der sich aus der Differenz zwischen der Zielposition 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. 18) und "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 21).
Datenrekorder	Der C-877 besitzt einen Echtzeit-Datenrekorder (S. 65). Der Datenrekorder kann verschiedene Signale (z. B. Position) aus verschiedenen Datenquellen (z. B. logische Achsen oder Eingangskanäle) aufzeichnen.
Makros	Der C-877 kann Makros (S. 66) 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-877 ausgeführt wird. Das Startup-Makro vereinfacht den Stand-Alone-Betrieb (Betrieb ohne Verbindung zum PC).  Weitere Informationen finden Sie im Abschnitt "Controllermakros" (S. 66).

Die Firmware kann mit einem Hilfsprogramm aktualisiert werden (S. 187).

# 3.7.4 Betriebsarten

Der C-877 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 ( <b>p</b> roportional-integral- <b>d</b> ifferenzial). Außerdem können weitere Korrekturen erfolgen.
	Das Ergebnis ist der Stellwert für die im C-877 integrierte Treiberelektronik.
	Weitere Informationen finden Sie in den Abschnitten "Erzeugung des Dynamikprofils" (S. 18) und "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 21).



Betriebsart	Beschreibung
Ungeregelter Betrieb (Servomodus Aus)	Im ungeregelten Betrieb berechnet der C-877 kein Dynamikprofil und wertet nicht die Signale des Positionssensors aus. Daher kann der Positionierer ungebremst an das Ende des Stellwegs fahren und trotz Endschalterfunktion auf dem mechanischen Anschlag aufprallen.

#### **INFORMATION**

Der C-877 ist für den geregelten Betrieb mit Positionssensoren vorgesehen (Servomodus Ein). Nach dem Einschalten ist standardmäßig der ungeregelte Betrieb aktiviert (Servomodus Aus).

- Fragen Sie die aktuelle Betriebsart mit den Befehlen SVO?, #4 oder SRG? ab.
- Aktivieren Sie den geregelten Betrieb mit dem Befehl SVO.
- Wenn nötig, programmieren Sie ein Startup-Makro, das den C-877 über den Befehl SVO im geregelten Betrieb startet; siehe "Startup-Makro einrichten" (S. 74).
- Vermeiden Sie Bewegungen im ungeregelten Betrieb.

## 3.7.5 Physikalische Einheiten

Der C-877 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	Zähler und Nenner des Faktors für Impulse pro physikalischer Längeneinheit
0xE	1 bis 1.000.000.000 für jeden Parameter.  Der Faktor für die Impulse pro physikalische Längeneinheit bestimmt
Denominator Of The Counts-Per-	die Längeneinheit für Positionsabfragen und Bewegungsbefehle im geregelten Betrieb.
<b>Physical-Unit Factor</b> 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	Einheitenzeichen
0x07000601	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



Parameter	Beschreibung und mögliche Werte
	werden. Das Einheitenzeichen für Rotationstische lautet normalerweise "deg".
	Der Wert des Parameters 0x07000601 wird nicht vom C-877 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

# 3.7.6 Auslösen von Bewegungen

## Bewegungen im geregelten Betrieb

Die folgende Tabelle gilt für Bewegungen im geregelten Betrieb.

Auslöser der Bewegung	Befehle	Beschreibung
Bewegungsbefehle, gesendet von der	MOV, MVR	Bewegung zu absoluter oder relativer Zielposition
Befehlszeile oder	GOH	Bewegung zur Nullposition
durch die PC- Software	STE	Startet einen Schritt und zeichnet die Sprungantwort auf
	FNL, FPL, FRF	Starten von Referenzfahrten
	FED	Starten von Fahrten zu Signalflanken
Controllermakros mit Bewegungsbefehlen	MAC	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.
	Weitere Makrobet 66).	fehle und Informationen siehe "Controllermakros" (S.

## **INFORMATION**

Absolute Zielpositionen können nur kommandiert werden, wenn die Achse zuvor referenziert wurde, siehe "Referenzierung" (S. 34).

C-877 PILine® Controller MS232D Version: 2.0.0 17



#### Bewegungen im ungeregelten Betrieb

Bewegungen werden mit dem Befehl SMO ausgelöst, der den Stellwert für die PILine® Treiberelektronik im C-877 direkt vorgibt.

Die über Befehle ausgelösten Bewegungen können über folgende Befehle gestoppt werden:

■ #24, STP: abruptes Stoppen

■ HLT: sanftes Stoppen

In beiden Fällen wird zur Information der Fehlercode 10 gesetzt.

## 3.7.7 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-877 erzeugte Dynamikprofil hängt von den Bewegungsgrößen ab, die durch Befehle (S. 84) und Parameter vorgegeben werden:

Bewegungs- größe	Befehle	Parameter	Anmerkungen
Beschleunigung (A)	ACC?	Beschleunigung im geregelten Betrieb (Parameter 0xB; physikalische Längeneinheit/s²); 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/s²); 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?	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.



Bewegungs- größe	Befehle	Parameter	Anmerkungen
Zielposition am Ende der Bewegung	MOV MVR GOH STE	-	Der C-877 setzt die Zielposition in folgenden Fällen auf die aktuelle Position der Achse:
			<ul><li>Einschalten des Servomodus mit dem Befehl SVO</li></ul>
			<ul> <li>Anhalten der Bewegung mit den Befehlen #24,</li> <li>STP oder HLT</li> </ul>

Der Profilgenerator des C-877 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) abbremst und an der vorgegebenen Zielposition anhält.

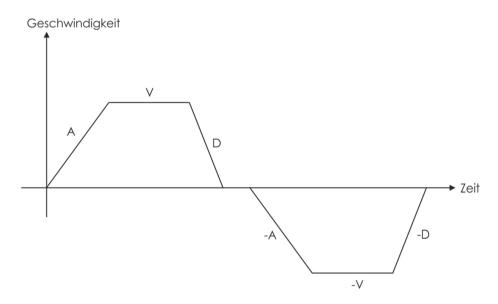


Abbildung 4: Einfaches trapezförmiges Geschwindigkeitsprofil; A = Beschleunigung, D = Abbremsung, V = Geschwindigkeit

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.



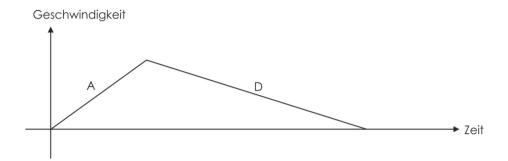


Abbildung 5: Einfaches trapezförmiges Geschwindigkeitsprofil; A = Beschleunigung, D = Abbremsung; keine konstante Geschwindigkeit

Die Flanken für Beschleunigung und Abbremsung können symmetrisch (Beschleunigung = Abbremsung) oder asymmetrisch (Beschleunigung ≠ 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.

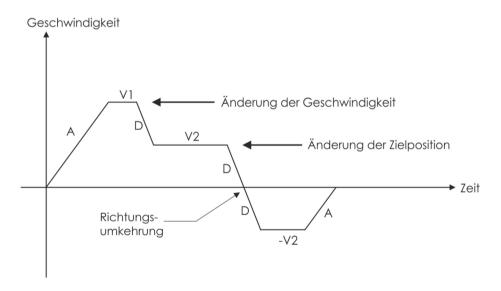


Abbildung 6: Komplexes trapezförmiges Profil mit Parameteränderungen; A = Beschleunigung; D = Abbremsung; V1, V2, -V2 = Geschwindigkeiten

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.



## 3.7.8 Regelalgorithmus und weitere Stellwertkorrekturen

Im geregelten Betrieb wird der Stellwert für die im C-877 integrierte PILine® Treiberelektronik 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. 18)) und der tatsächlichen Position (Rückmeldung des Sensors) ergibt, durchläuft einen P-I-D-Regelalgorithmus (proportional-integral-differenzial).
- Korrekturen des Dynamikprofils: Das errechnete Dynamikprofil kann einer Offset-Korrektur und einer Vorwärtsregelung der Geschwindigkeit unterzogen werden.

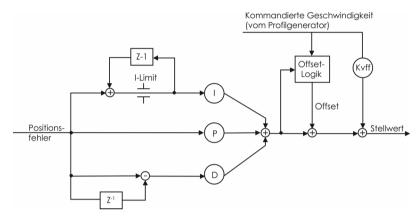


Abbildung 7: P-I-D-Algorithmus, Offset-Korrektur und Vorwärtsregelung der Geschwindigkeit (Kvff)

Für verfeinerte Korrekturen schaltet der C-877 während der Achsenbewegung im geregelten Betrieb zwischen den Parametergruppen 0 bis 4 um. Das Umschalten erfolgt anhand konfigurierbarer Positionsfenster.

Die Parametergruppen 0 bis 4 enthalten jeweils folgende Einstellungen:

- P-, I-, D-Terme und I-Limit für den Regelalgorithmus
- Kvff-Term für die Vorwärtsregelung der Geschwindigkeit
- Fenstergrenzen für Ein- und Austritt

#### **INFORMATION**

Aus Kompatibilitätsgründen hat der C-877 eine zusätzliche Gruppe von Parametern für Regelalgorithmus und Vorwärtsregelung der Geschwindigkeit:

Parameter 0x1, 0x2, 0x3, 0x4 und 0x5

Die Werte dieser Parameter werden automatisch auf die Werte der Regelparameter der Gruppe 1 (0x411, 0x412, 0x413, 0x414 und 0x415) eingestellt.

➤ Verwenden Sie für die Optimierung des dynamischen Verhaltens des Systems die Parameter 0x411, 0x412, 0x413, 0x414 und 0x415 (nicht verwenden: 0x1 bis 0x5); siehe "Regelparameter optimieren" (S. 56).

C-877 PILine® Controller MS232D Version: 2.0.0 21



#### Regelalgorithmus

Der Regelalgorithmus verwendet die folgenden Regelparameter. Die optimale Einstellung der Regelparameter hängt von Ihrer Anwendung und Ihren Wünschen ab; siehe "Regelparameter optimieren" (S. 56).

Parameter	Beschreibung und mögliche Werte
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.
P-Term 0 0x401 P-Term 1 0x411 P-Term 2 0x421 P-Term 2 0x431 P-Term 2 0x441	Proportionalkonstanten (dimensionslos) der Parametergruppen 0 bis 4 0 bis 65535 Ziel: Schnelle Korrektur des Positionsfehlers
I-Term 0 0x402 I-Term 1 0x412 I-Term 2 0x422 I-Term 2 0x432 I-Term 2 0x442	Integralkonstanten (dimensionslos) der Parametergruppen 0 bis 4 0 bis 65535 Ziel: Reduzierung des statischen Positionsfehlers
D-Term 0 0x403 D-Term 1 0x413 D-Term 2 0x423 D-Term 2 0x433 D-Term 2 0x443	Differenzialkonstanten (dimensionslos) der Parametergruppen 0 bis 4 0 bis 65535 Ziel: Dämpfung schneller Regelschwingungen
I-Limit 0 0x404 I-Limit 1 0x414 I-Limit 2 0x424 I-Limit 2 0x434 I-Limit 2 0x444	Begrenzung der Integrationskonstanten (dimensionslos) der Parametergruppen 0 bis 4 0 bis 65535

#### **INFORMATION**

Um nach dem Erreichen der Zielposition ein Servozittern der Achse zu vermeiden, sollte der I-Term der zum Einschwingen verwendeten Parametergruppe (standardmäßig Gruppe 0) deaktiviert oder minimiert sein.

Verwenden Sie zum Deaktivieren oder Minimieren eines I-Terms das zugehörige I-Limit. Beispiel: I-Term 0 (0x402) ist deaktiviert, wenn I-Limit 0 (0x404) den Wert null hat (Standardeinstellung)



Mit folgenden Parametern kann der Eingang des Regelalgorithmus des C-877 konfiguriert werden:

Parameter	Beschreibung und mögliche Werte
Numerator Of The Servo-Loop Input Factor	Zähler und Nenner des Eingangsfaktors des Regelkreises 1 bis 1.000.000 für beide Parameter
0x5A	Der Eingangsfaktor des Regelkreises entkoppelt die Regelparameter von der Encoderauflösung.
<b>Denominator Of The</b> <b>Servo-Loop Input Factor</b> 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 32767  Der Wertebereich entspricht 0 bis 10 V Stellspannung. Kompensiert die interne Vorspannung des Piezomotors.
Motor Offset Negative 0x34	Offset für die negative Bewegungsrichtung (dimensionslos).  0 bis 32767  Der Wertebereich entspricht 0 bis -10 V Stellspannung. Kompensiert die interne Vorspannung des Piezomotors.
<b>Motor Drive Offset</b> 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 32767  Der Wertebereich entspricht 0 bis 10 V Stellspannung. Je nach aktueller Bewegungsrichtung hat der Offsetwert ein positives oder negatives Vorzeichen.
Kvff 0 0x405 Kvff 1 0x415 Kvff 2 0x425 Kvff 2 0x435 Kvff 2 0x445	Vorwärtsregelung der kommandierten Geschwindigkeit für Parametergruppen 0 bis 4 0 bis 65535 Ziel: Minimierung des Positionsfehlers

#### **INFORMATION**

PILine® Ultraschall-Piezomotoren benötigen für den Start der Bewegung eine bestimmte, von null verschiedene Piezospannung. Deshalb werden Offsetwerte (Parameter 0x33, 0x34, 0x48) zum Stellwert und damit zur Stellspannung addiert. Durch einen geschwindigkeitsabhängigen Offset (0x48) sollen die Offsetwerte für die positive und negative Bewegungsrichtung (0x33

C-877 PILine® Controller MS232D Version: 2.0.0 23



und 0x34) möglichst klein gehalten werden. Die optimalen Offsetwerte für die positive und negative Bewegungsrichtung können insbesondere bei vertikal ausgerichteter Bewegungsachse stark voneinander abweichen.

### Umschalten zwischen den Parametergruppen 0 bis 4

Das Umschalten zwischen den Parametergruppen 0 bis 4 für Regelalgorithmus und Vorwärtsregelung der Geschwindigkeit kann durch die nachfolgend aufgelisteten Parameter konfiguriert werden.

Parameter	Beschreibung und mögliche Werte
Servo Window Mode	Bezugsgröße für die Positionsfenster
0x4D	0 = Zielposition (Standardeinstellung)
	1 = kommandierte Position (aus Dynamikprofil)
	Dieser Parameter gibt die Bezugsgröße für die Positionsfenster vor,
	die zum Umschalten zwischen den Parametergruppen 0 bis 4 für Regelalgorithmus und Vorwärtsregelung verwendet werden. Das Umschalten erfolgt anhand der Differenz zwischen aktueller Position und gewählter Bezugsgröße.
Number Of Servo	Maximale Anzahl verwendeter Parametergruppen
Parameter Groups	1 bis 5
0x400	Dieser Parameter gibt die maximale Anzahl der Parametergruppen an, zwischen denen während der Achsenbewegung umgeschaltet wird.
	Standardeinstellung: 3 (Parametergruppen 0 bis 2 werden verwendet)
Window Enter 0	Positionsfenster für das Aktivieren der Parametergruppen 0 bis 4
0x406	0 bis 2 <sup>31</sup> Impulse des Encoders
Window Enter 1	Die Parameter geben die Eintrittsfenster für die Parametergruppen
0x416	vor. Die Fenster sind um die mit Parameter 0x4D gewählte
Window Enter 2	Bezugsgröße zentriert. Wenn die aktuelle Position in das Eintrittsfenster einer Parametergruppe eintritt, wird diese
0x426	Parametergruppe aktiviert.
Window Enter 3	Die Parameterwerte entsprechen jeweils der Hälfte der
0x436	Fensterbreite. Sie können nur geändert werden, wenn der
Window Enter 4	Servomodus ausgeschaltet ist.
0x446	
Window Exit 0	Positionsfenster für das Deaktivieren der Parametergruppen 0 bis 4
0x407	0 bis 2 <sup>31</sup> Impulse des Encoders
Window Exit 1	Die Parameter geben die Austrittsfenster für die Parametergruppen
0x417	vor. Die Fenster sind um die mit Parameter 0x4D gewählte
Window Exit 2	Bezugsgröße zentriert. Wenn die aktuelle Position das Austrittsfenster einer Parametergruppe verlässt, wird diese
0x427	Parametergruppe deaktiviert und die nächsthöhere
Window Exit 3	Parametergruppe aktiviert.
0x437	Die Parameterwerte entsprechen jeweils der Hälfte der
Window Exit 4	Fensterbreite. Sie können nur geändert werden, wenn der
0x447	Servomodus ausgeschaltet ist.



#### **INFORMATION**

Für die Positionsfenster gilt Folgendes:

- Das Eintrittsfenster für Parametergruppe n muss kleiner sein als das Eintrittsfenster für Parametergruppe n+1.
- Das Austrittsfenster für Parametergruppe n muss kleiner sein als das Austrittsfenster für Parametergruppe n+1.
- Die Positionsfenster der "äußersten" verwendeten Parametergruppe werden ignoriert. Welche Parametergruppe die äußerste verwendete Gruppe ist, hängt von der Einstellung des Parameters *Number Of Servo Parameter Groups* (0x400) ab. Beispiel: Parameter 0x400 hat den Wert 3. Das Umschalten erfolgt dann zwischen den Parametergruppen 0, 1 und 2. Parametergruppe 2 ist die äußerste verwendete Parametergruppe. Da die Positionsfenster der Parametergruppe 2 ignoriert werden, bleibt sie auch dann aktiviert, wenn die aktuelle Position außerhalb ihres Austrittsfensters liegt (0x427).
- Die Ein- und Austrittsfenster für Parametergruppe 0 werden auch als Einschwingfenster für die Ermittlung des On-Target-Status verwendet (S. 27).
- Für ein stabiles Umschaltverhalten sollte das Austrittsfenster einer Parametergruppe größer sein als ihr Eintrittsfenster.

Die beiden folgenden Abbildungen zeigen das Umschalten zwischen Parametergruppen während der Achsenbewegung. Einstellungen in den Beispielen:

- Bezugsgröße des Umschaltens: Zielposition (obere Abbildung) oder kommandierte Position (untere Abbildung)
- Die Eintrittsfenster der Parametergruppen sind kleiner als ihre Austrittsfenster.
- Die maximale Anzahl verwendeter Parametergruppen ist 3.
- Wenn die aktuelle Position in das Eintrittsfenster für Parametergruppe 0 eintritt, erfolgt die Aktivierung der Parametergruppe 0.

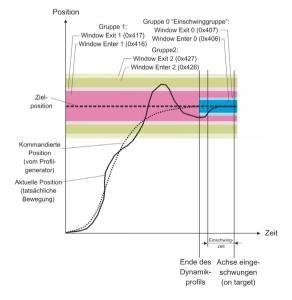


Abbildung 8: Umschalten zwischen Parametergruppen 0 bis 2 anhand der Differenz zwischen aktueller Position und Zielposition



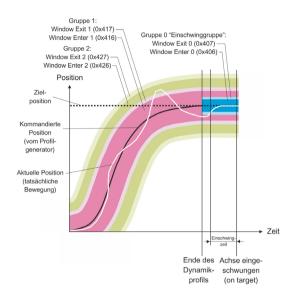


Abbildung 9: Umschalten zwischen Parametergruppen 0 bis 2 anhand der Differenz zwischen aktueller Position und kommandierter Position

## 3.7.9 Optionale Zweiphasenansteuerung

Die Zweiphasenansteuerung (S. 2) ist standardmäßig deaktiviert. Je nach Anwendung kann die Zweiphasenansteuerung aktiviert werden, um das Einschwingverhalten zu verbessern. Um die Zweiphasenansteuerung zu aktivieren, muss das Einschalt-/Pausen-Verhältnis für das parallele Ansteuern beider Piezosegmente eines PILine® Piezomotors durch die unten aufgelisteten Parameter (Gruppen 0 bis 4) entsprechend konfiguriert werden.

Das Umschalten zwischen den Parametergruppen 0 bis 4 erfolgt während der Achsenbewegung im geregelten Betrieb. Der C-877 verwendet zum Umschalten zwischen den Parametergruppen die Parameter, die auch für den Regelalgorithmus verwendet werden, siehe "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 21).

Parameter	Beschreibung und mögliche Werte
2nd Phase On 0 (No. Of Servo Cycles) 0x409	Einschaltdauer der parallelen Ansteuerung beider Piezosegmente für Parametergruppen 0 bis 4
2nd Phase On 1 (No. Of Servo Cycles)	0 (Standardeinstellung) bis 34464 Servozyklen
0x419	
2nd Phase On 2 (No. Of Servo Cycles)	
0x429	
2nd Phase On 3 (No. Of Servo Cycles)	
0x439	
2nd Phase On 4 (No. Of Servo Cycles)	
0x449	



Parameter	Beschreibung und mögliche Werte
<b>2nd Phase Off 0 (No. Of Servo Cycles)</b> 0x40A	Pausendauer der parallelen Ansteuerung beider Piezosegmente für Parametergruppen 0 bis 4
<b>2nd Phase Off 1 (No. Of Servo Cycles)</b> 0x41A	0 (Standardeinstellung) bis 34464 Servozyklen
2nd Phase Off 2 (No. Of Servo Cycles) 0x42A	
2nd Phase Off 3 (No. Of Servo Cycles) 0x43A	
2nd Phase Off 4 (No. Of Servo Cycles) 0x44A	

## 3.7.10 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-877 ermittelt den On-Target-Status anhand folgender Kriterien:

- Einschwingfenster um die Zielposition, wird vorgegeben durch die Ein- und Austrittsfenster für Parametergruppe 0 (Parameter 0x406 und 0x407)
- 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.

Parameter	Beschreibung und mögliche Werte
<b>Settling Time (s)</b> 0x3F	Verzögerungszeit für das Setzen des On-Target-Status 0 bis 1,000 s
Window Enter 0 (encoder counts) 0x406	Einschwingfenster um die Zielposition 0 bis 2 <sup>31</sup> Impulse des Encoders Die Parameter geben die Fenstergrenzen für Eintritt und Austritt vor. Wenn die aktuelle Position das Einschwingfenster verlässt, gilt die
Window Exit 0 (encoder counts) 0x407	Zielposition nicht mehr als erreicht.  Die Parameterwerte entsprechen jeweils der Hälfte der Fensterbreite. Sie können nur geändert werden, wenn der Servomodus ausgeschaltet ist.



## 3.7.11 Unterstützte Motortypen

Der C-877 unterstützt alle aktuell von PI angebotenen oder in PILine<sup>®</sup> Positionierern und Antrieben integrierten Typen von PILine<sup>®</sup> Ultraschall-Piezomotoren. Die Anpassung an den angeschlossenen Motortyp erfolgt über die folgenden Parameter:

Parameter	Beschreibung und mögliche Werte
Maximum Motor Output (V) 0x7C	Maximal zulässige Piezospannung 0 bis 71 V <sub>rms</sub> Dieser Parameter bestimmt die maximal zulässige Amplitude der ausgegebenen Piezospannung.
Output Frequency (kHz) 0x51	Frequenz der Piezospannung  0 bis 500 kHz  Dieser Parameter bestimmt die Frequenz, mit der die ausgegebene Piezospannung oszilliert, um den Piezoaktor im PILine® Ultraschall- Piezomotor anzuregen. Quellen für den Wert des Parameters:
	<ul> <li>Wenn die Frequenzregelung (S. 28) eingeschaltet und aktiv ist: Vorgabe durch die Frequenzregelung</li> <li>Direktes Ändern z. B. mit dem Befehl SPA (schaltet gleichzeitig die Frequenzregelung aus)</li> </ul>

#### **INFORMATION**

Weitere Informationen finden Sie im Benutzerhandbuch Ihres PILine® Positionierers.

## 3.7.12 Automatische Frequenzregelung

Der C-877 ist mit einer Frequenzregelung ausgestattet, die die Frequenz der ausgegebenen Piezospannung optimiert. Im optimalen Arbeitspunkt liegt die Frequenz der Piezospannung so nahe wie möglich bei der Resonanzfrequenz des angeschlossenen Motors. Die Resonanzfrequenz des Motors wird von verschiedenen Faktoren beeinflusst:

- Motortyp
- Einbaubedingungen des Motors
- Durchführung der Einlaufprozedur
- Temperatur

Die Frequenzregelung arbeitet mit 1 kHz.

Die Frequenzregelung kann durch die nachfolgend aufgelisteten Parameter konfiguriert werden:

Parameter	Beschreibung und mögliche Werte
Frequency Control	Zustand der Frequenzregelung
0x52	0 = Frequenzregelung ausgeschaltet



Parameter	Beschreibung und mögliche Werte
	1 = Frequenzregelung eingeschaltet (Standardeinstellung) Wenn die Frequenzregelung eingeschaltet <b>und</b> aktiv ist, setzt sie den Wert des Parameters <b>Output Frequency (kHz)</b> (0x51, siehe "Unterstützte Motortypen" (S. 28)). Das Kriterium für die Aktivierung der Frequenzregelung wird durch Parameter 0x55 vorgegeben. Direktes Ändern des Parameters 0x51 (z. B. mit dem Befehl SPA) schaltet gleichzeitig die Frequenzregelung aus.
Minimum Output Frequency (kHz) 0x53	Minimale Frequenz der Piezospannung (kHz) 0 bis 500 kHz Dieser Parameter gibt den kleinsten möglichen Wert für Parameter 0x51 vor, wenn die Frequenzregelung eingeschaltet und aktiv ist.
Maximum Output Frequency (kHz) 0x54	Maximale Frequenz der Piezospannung (kHz)  0 bis 500 kHz  Dieser Parameter gibt den größten möglichen Wert für Parameter 0x51 vor, wenn die Frequenzregelung eingeschaltet und aktiv ist.
Minimum Motor Output For Frequency Control 0x55	Minimaler Stellwert für Aktivierung der Frequenzregelung 0 bis 32767 Wenn der Betrag des aktuellen Stellwerts mindestens so groß ist wie der Wert dieses Parameters, wird die eingeschaltete Frequenzregelung aktiv und setzt den Wert des Parameters 0x51.

# 3.7.13 Referenzschaltererkennung

Der C-877 nimmt auf Pin 13 der Buchse **Motor** (S. 200) das Signal eines Referenzschalters entgegen.

Mit folgenden Parametern kann konfiguriert werden, wie der C-877 den Referenzschalter erkennt:

Parameter	Beschreibung und mögliche Werte
Invert Reference?	Soll das Referenzsignal invertiert werden?
0x31	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.
Has Reference?	Hat der Positionierer einen Referenzschalter?
0x14	0 = Kein Referenzschalter eingebaut
	1 = Referenzschalter vorhanden (Signaleingang an der Buchse <b>Motor</b> )
	Dieser Parameter aktiviert bzw. deaktiviert Referenzfahrten zum eingebauten Referenzschalter.



Parameter	Beschreibung und mögliche Werte
Reference Signal	Art des Referenzsignals
<b>Type</b> 0x70	0 = richtungserkennender Referenzschalter (Standardeinstellung). Der Signalpegel ändert sich beim Überfahren des Referenzschalters.
	1 = Pulssignal mit einer Pulsweite von mehreren Nanosekunden (Parameter 0x47 muss korrekt gesetzt sein).
	2 = Indexpuls. Das Anfahren erfolgt über den negativen Endschalter.
	3 = Indexpuls. Das Anfahren erfolgt über den positiven Endschalter.

Das Signal des Referenzschalters des Positionierers kann für Referenzfahrten verwendet werden. Bei einem Positionierer mit inkrementellem Positionssensor kennt der Controller nach einer Referenzfahrt zum Referenzschalter die absolute Achsenposition; siehe "Referenzwertbestimmung" (S. 34).

## 3.7.14 Endschaltererkennung

Der C-877 nimmt auf der Buchse Motor (S. 200) Endschaltersignale entgegen:

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

Mit folgenden Parametern kann konfiguriert werden, wie der C-877 die Endschalter erkennt:

Parameter	Beschreibung und mögliche Werte
Limit Mode	Signallogik der Endschalter
0x18	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
Has No Limit Switches?	Hat der Positionierer Endschalter?
0x32	0 = Positionierer hat Endschalter (Signaleingänge an der Buchse <b>Motor</b> )
	1 = Positionierer hat keine Endschalter
	Dieser Parameter aktiviert bzw. deaktiviert das Anhalten der Bewegung an den eingebauten Endschaltern.
Use Limit Switches Only For Reference Moves? 0x77	Sollen die Endschalter nur für Referenzfahrten verwendet werden?  0 = Endschalter zum Anhalten am Ende des Stellwegs und für Referenzfahrten verwenden (Standard)  1 = Endschalter nur für Referenzfahrten verwenden  Dieser Parameter ist für die Verwendung mit Rotationstischen vorgesehen.
	Dieser Parameter wird nur ausgewertet, wenn Parameter 0x32 den Wert 0 hat.

Die Signale der Endschalter (auch Endlagensensoren) eines Lineartisches 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 Positionierer trotzdem auf dem mechanischen Anschlag aufprallt. Um dies zu vermeiden, können Verfahrbereichsgrenzen (S. 31) über Parameter des C-877 eingestellt werden.

Die Endschaltersignale können auch für Referenzfahrten verwendet werden. Bei einem Positionierer mit inkrementellem Positionssensor kennt der Controller nach einer Referenzfahrt zu einem Endschalter die absolute Achsenposition; siehe "Referenzwertbestimmung" (S. 34).

# 3.7.15 Stellweg und Verfahrbereichsgrenzen

Folgende Parameter des C-877 spiegeln den physikalischen Stellweg des Positionierers wider und definieren Verfahrbereichsgrenzen:

Parameter	Beschreibung und mögliche Werte		
Maximum Travel In Positive Direction (Phys. Unit) 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.		
Value At Reference Position (Phys. Unit) 0x16	Positionswert am Referenzschalter (physikalische Einheit) Die aktuelle Position wird auf diesen Wert gesetzt, wenn die Achse eine Referenzfahrt zum Referenzschalter ausgeführt hat. 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.		
Distance From Negative Limit To Reference Position (Phys. Unit) 0x17	Abstand zwischen Referenzschalter und negativem Endschalter (physikalische Einheit) Wenn die Achse eine Referenzfahrt zum negativen Endschalter ausgeführt hat, wird die aktuelle Position auf die Differenz zwischen den Werten der Parameter 0x16 und 0x17 gesetzt.		
Distance From Reference Position To Positive Limit (Phys. Unit) 0x2F	Abstand zwischen Referenzschalter und positivem Endschalter (physikalische Einheit) Wenn die Achse eine Referenzfahrt zum positiven Endschalter ausgeführt hat, wird die aktuelle Position auf die Summe der Werte der Parameter 0x16 und 0x2F gesetzt.		
Maximum Travel In Negative Direction (Phys. Unit) 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.		
<b>Range Limit Min</b> 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		



Parameter	Beschreibung und mögliche Werte		
	Verfahrbereichsgrenze verringert wurde, kann die Achse wieder bewegt werden.		
<b>Range Limit Max</b> 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-877 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 Positionierertyps aus der Positioniererdatenbank geladen.
- 0x07000000 (Range Limit Min) und 0x07000001 (Range Limit Max):
  - Die Verwendung dieser Grenzen ist nur dann empfohlen, wenn ungeregelte 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 ungeregelten 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 Positioniererdatenbank geladen und sind in der Werkseinstellung so gesetzt, dass die Grenzen deaktiviert sind.

## **Beispiele**

Die nachfolgenden Beispiele beziehen sich auf eine Achse eines Positionierers 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. 34) 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

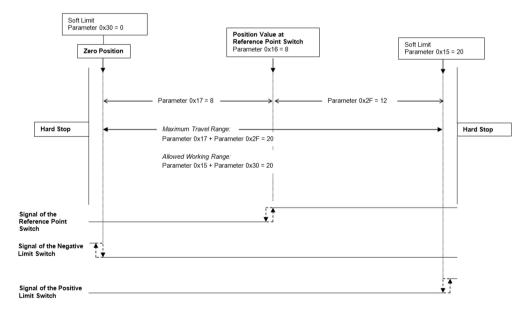


Abbildung 10: Der Stellweg der Achse wird nicht durch Verfahrbereichsgrenzen beschränkt.

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.

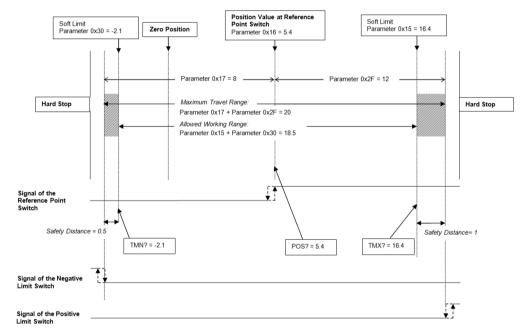


Abbildung 11: Der Stellweg der Achse wird durch Verfahrbereichsgrenzen beschränkt.

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



# 3.7.16 Referenzierung

Der C-877 unterstützt ausschließlich Positionierer, die mit einem inkrementellen Positionssensor ausgestattet sind. Inkrementelle Sensoren liefern nur relative Bewegungsinformationen. Deshalb kennt der Controller beim Einschalten oder Neustart die absolute Position der Achse nicht. Bevor absolute Zielpositionen kommandiert und erreicht werden können, muss für die Achse eine Referenzwertbestimmung durchgeführt werden.

Die Referenzierung 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 Referenzierungsmethode mit dem Befehl RON (S. 121) gewählt wurde, können Sie mit dem Befehl POS (S. 120) 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 Referenzierung standardmäßig durch eine Referenzfahrt. Die Kenntnis der hier beschriebenen Befehle und Parameter ist für die Referenzierung mit PIMikroMove® nicht erforderlich.

#### **INFORMATION**

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

- 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 Referenzierung 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>}</referenceon></axisid>	Bestimmt den Modus der Referenzwertbestimmung ( <referenceon>) für die Achse:  O: Für die Referenzwertbestimmung der Achse kann ein absoluter Positionswert mit POS zugewiesen werden, oder eine Referenzfahrt kann mit FRF, FNL oder FPL gestartet werden.  1 (Standard): Für die Referenzwertbestimmung der Achse muss eine Referenzfahrt mit FRF, FNL oder FPL gestartet werden. Die Verwendung von POS ist nicht zulässig.</referenceon>		
RON?	RON? [{ <axisid>}]</axisid>	Fragt den Modus der Referenzwertbestimmung ab.		
FRF	FRF [{ <axisid>}]</axisid>	<ul> <li>Startet eine Referenzfahrt zum Referenzschalter.</li> <li>Das Anfahren hängt vom Wert des Parameters Reference Signal Type (0x70) ab:</li> <li>O oder 1: Das Anfahren erfolgt immer von derselben Seite, unabhängig von der Achsenposition beim Senden des Befehls.</li> <li>2: Das Anfahren erfolgt über den negativen Endschalter.</li> <li>3: Das Anfahren erfolgt über den positiven Endschalter.</li> </ul>		
FRF?	FRF? [{ <axisid>}]</axisid>	Fragt ab, ob der Referenzwert für eine Achse bereits bestimmt wurde.  1 = Referenzwert wurde bestimmt  0 = Referenzwert wurde nicht bestimmt		
FNL	FNL [{ <axisid>}]</axisid>	Startet eine Referenzfahrt zum negativen Endschalter.		
FPL	FPL [{ <axisid>}]</axisid>	Startet eine Referenzfahrt zum positiven Endschalter.		
POS	POS { <axisid> <position>}</position></axisid>	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. 18).			
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. 18).			
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.			
Distance From Limit To Start Of Ref Search (Phys. Unit) 0x78	Abstand zwischen Endschalter und Startposition für die Bewegung zum Indexpuls. Details siehe Erläuterung unterhalb der Tabelle.			
Distance For Reference Search (Phys. Unit) 0x79	Maximale Strecke für die Bewegung zum Indexpuls Details siehe Erläuterung unterhalb der Tabelle.			

Die Parameter 0x78 und 0x79 werden für Referenzfahrten verwendet, wenn die beiden folgenden Bedingungen erfüllt sind:

- Die Referenzfahrt wird mit FRF gestartet.
- Der Parameter Reference Signal Type (0x70) hat den Wert 2 oder 3.

## Ablauf der Referenzfahrt:

- 1. Die Achse bewegt sich zum entsprechenden Endschalter.
- 2. Die Achse bewegt sich um die mit dem Parameter 0x78 angegebene Strecke weg vom Endschalter.



3. Die Achse bewegt sich zum Indexpuls und legt maximal die mit dem Parameter 0x79 vorgegebene Strecke zurück.

### **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. 31) beschränkt wird.

### **INFORMATION**

Wenn die absolute Position der Achse mit dem Befehl POS 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 Referenzierung nicht anders möglich ist.

### **INFORMATION**

Wenn die aktuellen Parametereinstellungen des C-877 in PIMikroMove® oder durch Eingabe des Befehls WPA unter Verwendung des Kennworts 100 oder 101 in den permanenten Speicher geschrieben werden, gilt die Achse anschließend nicht mehr als "referenziert" (die Antwort auf FRF? ist 0).



# 4 Auspacken

- 1. Packen Sie den C-877 vorsichtig aus.
- 2. Vergleichen Sie die erhaltene Lieferung mit dem Lieferumfang laut Vertrag und mit dem Lieferschein.
- 3. Überprüfen Sie den Inhalt auf Anzeichen von Schäden. Bei Schäden oder fehlenden Teilen wenden Sie sich sofort an unseren Kundendienst (S. 195).
- 4. Bewahren Sie das komplette Verpackungsmaterial auf für den Fall, dass das Produkt zurückgeschickt werden muss.



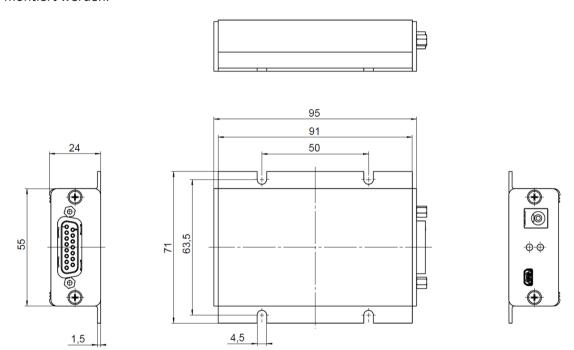
# 5 Installation

# 5.1 Allgemeine Hinweise zur Installation

- Installieren Sie den C-877 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.

# 5.2 C-877 montieren

Der C-877 kann als Tischgerät verwendet oder in beliebiger Ausrichtung auf einer Unterlage montiert werden.



### Werkzeug und Zubehör

- Geeignete Schrauben
- Geeigneter Schraubendreher

#### C-877 montieren

1. Bringen Sie in die Unterlage die erforderlichen Bohrungen ein.

C-877 PILine® Controller MS232D Version: 2.0.0 41



- Die Anordnung der Aussparungen in den Montageleisten des C-877 können Sie der Abbildung entnehmen.
- 2. Befestigen Sie den C-877 an den Aussparungen in den Montageleisten mit jeweils zwei geeigneten Schrauben pro Seite.

## 5.3 C-877 an Schutzleiter anschließen

#### **INFORMATION**

Beachten Sie die jeweils geltenden Normen für die Schutzleiterbefestigung.

## Voraussetzungen

- ✓ Sie haben die allgemeinen Hinweise zur Installation gelesen und verstanden (S. 41).
- ✓ Der C-877 ist nicht an die Stromversorgung angeschlossen.

#### Werkzeug und Zubehör

- Geeigneter Schutzleiter:
  - Kabelquerschnitt ≥0,75 mm²
  - Übergangswiderstand <0,1 Ohm bei 25 A an allen für die Schutzleitermontage relevanten Stellen
- Befestigungsmaterial für den Schutzleiter, sitzt bei Auslieferung des C-877 in folgender Reihenfolge auf dem Schutzleiteranschluss (Gewindebolzen), ausgehend vom Gehäuse:
  - Zahnscheibe
  - Mutter
  - Unterlegscheibe
  - Sicherungsscheibe
  - Mutter
- Geeigneter Schraubenschlüssel

#### C-877 an Schutzleiter anschließen

- 1. Wenn nötig, befestigen Sie einen geeigneten Kabelschuh am Schutzleiter.
- 2. Entfernen Sie die äußere Mutter vom Schutzleiteranschluss des C-877 (mit gekennzeichneter Gewindebolzen (S. 7)).
- 3. Schließen Sie den Schutzleiter an:
  - a) Schieben Sie den Kabelschuh des Schutzleiters auf den Gewindebolzen.
  - b) Schrauben Sie die Mutter auf den Gewindebolzen. Der Kabelschuh des Schutzleiters wird auf diese Weise zwischen der Zahnscheibe und der Mutter eingeklemmt.
  - c) Ziehen Sie die Mutter mit mindestens drei Umdrehungen und einem Drehmoment von 1,2 Nm bis 1,5 Nm fest.



# 5.4 Netzteil an C-877 anschließen

## Voraussetzungen

✓ Das Netzkabel ist **nicht** an der Steckdose angeschlossen.

#### Werkzeug und Zubehör

- Mitgeliefertes 24-V-Weitbereichsnetzteil (für Netzspannungen zwischen 100 und 240 Volt Wechselspannung bei 50 oder 60 Hz)
   Alternativ: ausreichend bemessenes Netzteil, das 24 Volt Gleichspannung und einen maximalen Ausgangsstrom von mindestens 0,8 Ampere liefert
- Mitgeliefertes Netzkabel
   Alternativ: ausreichend bemessenes Netzkabel

#### Netzteil an den C-877 anschließen

- Verbinden Sie den Hohlstecker des Netzteils mit dem 24-V-Anschluss (24 V = -- 0.8 A) des C-877.
- Verbinden Sie das Netzkabel mit dem Netzteil.

# 5.5 Positionierer anschließen

## **HINWEIS**



### Schäden bei Anschluss eines falschen Motors!

Das Anschließen eines Positionierers mit DC-Motor, Schrittmotor oder Voice-Coil-Antrieb an den C-877 kann irreparable Schäden am Positionierer oder Controller verursachen.

> Schließen Sie an den C-877 nur Positionierer mit PILine® Ultraschall-Piezomotoren an.

# Voraussetzungen

- ✓ Der C-877 ist ausgeschaltet, d. h. das Netzteil ist nicht über das Netzkabel an der Steckdose angeschlossen.
- ✓ Sie haben das Benutzerhandbuch des Positionierers gelesen und verstanden.

#### Positionierer anschließen

- 1. Schließen Sie den Positionierer an der Buchse **Motor** des C-877 an.
- 2. Sichern Sie die Steckverbindungen mit den integrierten Schrauben gegen unbeabsichtigtes Abziehen.



## 5.6 PC-Software installieren

Die Kommunikation zwischen dem C-877 und einem PC ist zur Konfiguration des C-877 und zur Bewegungskommandierung mit den Befehlen des GCS notwendig. Dafür stehen verschiedene PC-Softwareanwendungen zur Verfügung.

### 5.6.1 Erstinstallation ausführen

# Zubehör

- PC mit Betriebssystem Windows oder Linux und mindestens 30 MB freiem Speicherplatz
- Datenträger mit der PI Software Suite (im Lieferumfang)
   Informationen zur Kompatibilität der Software mit PC-Betriebssystemen finden Sie in der C-990.CD1 Release News im Hauptverzeichnis des Datenträgers.

#### PC-Software auf Windows installieren

- 1. Starten Sie den Installationsassistenten, indem Sie im Installationsverzeichnis (Hauptverzeichnis des Datenträgers) auf die Datei *PISoftwareSuite.exe* doppelklicken.
  - Das Fenster *InstallShield Wizard* für die Installation der PI Software Suite wird geöffnet.
- 2. Folgen Sie den Anweisungen am Bildschirm.

Die PI Software Suite umfasst unter anderem folgende Komponenten:

- Treiber zur Verwendung mit NI LabVIEW-Software
- Dynamische Programmbibliothek f
  ür GCS
- PIMikroMove®
- PC-Software zum Aktualisieren der Firmware des C-877
- PIUpdateFinder zum Aktualisieren der PI Software Suite
- USB-Treiber

#### PC-Software auf Linux installieren

- 1. Entpacken Sie das tar-Archiv aus dem Verzeichnis /Linux des Datenträgers 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.
- Um die Installation zu starten, geben Sie ./INSTALL ein.
   Achten Sie beim Eingeben des Befehls auf Groß-/Kleinschreibung.
- 5. Folgen Sie den Anweisungen am Bildschirm.

Sie können einzelne Komponenten zur Installation auswählen.



## 5.6.2 Updates installieren

Die PC-Software wird von PI ständig verbessert.

Installieren Sie immer die neueste Version der PC-Software und der Positioniererdatenbank.

#### Voraussetzungen

- ✓ Aktive Verbindung zum Internet
- ✓ Wenn Ihr PC ein Windows-Betriebssystem verwendet:
  - Sie haben das Handbuch für den PIUpdateFinder (A000T0028) von der PI Website heruntergeladen. Sie finden den Link in der Datei "A000T0081-Downloading Manuals from PI.pdf" im Ordner \Manuals auf dem Datenträger mit der PI Software Suite.

#### PC-Software und Positioniererdatenbank auf Windows aktualisieren

- Verwenden Sie den PIUpdateFinder:
  - Folgen Sie den Anweisungen im Handbuch für den PIUpdateFinder (A000T0028).

#### PC-Software auf Linux aktualisieren

- 1. Öffnen Sie die Webseite https://www.physikinstrumente.de/de/produkte/software-suite (https://www.physikinstrumente.de/de/produkte/software-suite).
- 2. Scrollen Sie nach unten zu **Downloads**.
- 3. Für PI Software Suite C-990.CD1: Wählen Sie HINZUFÜGEN+
- 4. Wählen Sie ANFORDERN
- Füllen Sie das Anfrageformular aus und senden Sie die Anfrage ab.
   Der Download-Link wird an die eingegebene E-Mail-Adresse gesendet.
- 6. Entpacken Sie die Archivdatei auf Ihrem PC in ein separates Installationsverzeichnis.
- 7. Wechseln Sie im Verzeichnis mit den entpackten Dateien in das Unterverzeichnis linux.
- 8. Entpacken Sie die Archivdatei im Verzeichnis *linux*, indem Sie in der Konsole den Befehl tar -xvpf <Name der Archivdatei> eingeben.
- 9. Melden Sie sich am PC als Superuser (Root-Rechte) an.
- 10. Installieren Sie das Update.

#### **INFORMATION**

Wenn Software im Bereich *Downloads* fehlt oder Probleme beim Herunterladen auftreten:

Wenden Sie sich an unseren Kundendienst (S. 195).

#### Positioniererdatenbank auf Linux aktualisieren

1. Wenden Sie sich an den Kundendienst (S. 195), um die neueste Version der Positioniererdatenbank *pistages2* oder *pimicosstages2* zu erhalten.



- 2. Melden Sie sich am PC als Superuser (Root-Rechte) an.
- 3. Installieren Sie das Update, die Sie von unserem Kundendienst erhalten haben, auf dem PC.

# 5.6.3 Kundenspezifische Positioniererdatenbank installieren

Mit einem kundenspezifischen Positionierer erhalten Sie von PI gegebenenfalls eine Datei mit einer kundenspezifischen Positioniererdatenbank. Sie müssen diese Datei auf Ihrem PC installieren, damit Sie die Parameterwerte für den kundenspezifischen Positionierer in den C-877 laden können.

### Kundenspezifische Positioniererdatenbank 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:\ProgramData.

2. Kopieren Sie die Datei der Positioniererdatenbank 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 Positioniererdatenbank zugreifen kann, müssen beide Dateien im selben Verzeichnis liegen.

#### Kundenspezifische Positioniererdatenbank auf Linux installieren

- 1. Melden Sie sich am PC als Superuser (Root-Rechte) an.
- 2. Kopieren Sie die Datei der Positioniererdatenbank in das Verzeichnis /usr/local/PI/pi\_gcs\_translator/.

### 5.7 PC anschließen

Die Kommunikation zwischen dem C-877 und einem PC ist zur Konfiguration des C-877 und zur Bewegungskommandierung mit den Befehlen des GCS notwendig. Der C-877 verfügt dazu über eine USB-Schnittstelle.

In diesem Abschnitt erfahren Sie, wie Sie die entsprechenden Kabelverbindungen zwischen C-877 und PC herstellen.

Die Schritte zur Herstellung der Kommunikation zwischen C-877 und PC sind im Kapitel "Inbetriebnahme" beschrieben: "Kommunikation über USB herstellen" (S. 50)



# 5.7.1 C-877 über die USB-Schnittstelle anschließen

## Voraussetzungen

✓ Der PC verfügt über eine freie USB-Schnittstelle.

# Werkzeug und Zubehör

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

### C-877 an den PC anschließen

Verbinden Sie die USB-Buchse des C-877 und die USB-Schnittstelle des PC mit dem USB-Kabel.



# 6 Inbetriebnahme

# 6.1 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-877.
- Prüfen Sie die Funktion der Endschalter bei etwa 10 % bis 20 % der Maximalgeschwindigkeit.
- ➤ Halten Sie bei einer Fehlfunktion der Endschalter die Bewegung sofort an.

## 6.2 C-877 einschalten

### **INFORMATION**

Der C-877 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 SVO?, #4 oder SRG? ab.
- Aktivieren Sie den geregelten Betrieb mit dem Befehl SVO.
- Wenn nötig, programmieren Sie ein Startup-Makro, das den C-877 über den Befehl SVO im geregelten Betrieb startet; siehe "Startup-Makro einrichten" (S. 74).

#### **INFORMATION**

Der ID-Chip wird nicht ausgelesen, wenn Sie den Positionierer bei eingeschaltetem C-877 anschließen.

Starten Sie nach dem Anschließen eines Positionierers den C-877 mit dem Befehl RBT (S. 121) oder mit den entsprechenden Funktionen der PC-Software neu, um die Daten vom ID-Chip auszulesen.



### Voraussetzungen

- ✓ Sie haben die allgemeinen Hinweise zur Inbetriebnahme gelesen und verstanden (S. 49).
- ✓ Der C-877 wurde ordnungsgemäß installiert (S. 41).

#### C-877 einschalten

Verbinden Sie das Netzkabel des Netzteils mit der Steckdose.

Der C-877 lädt in folgender Reihenfolge Informationen in den flüchtigen Speicher:

- a) Parameterwerte aus dem permanenten Speicher
- b) Parameterwerte vom ID-Chip des Positionierers

Die LED **STA** an der Vorderwand des C-877 zeigt den Status des C-877 an:

- grün: C-877 ist bereit für den Normalbetrieb
- aus: C-877 ist nicht an der Stromversorgung angeschlossen oder könnte defekt sein
- Wenn der C-877 korrekt mit dem Netzteil verbunden ist (S. 43) und die LED **STA** nach dem Einschalten nicht leuchtet, kontaktieren Sie unseren Kundendienst (S. 195).

# 6.3 Kommunikation herstellen

#### **INFORMATION**

Für die USB-Schnittstelle wird im C-877 ein USB-UART-Modul (FTDI) verwendet. Wenn der C-877 über den USB-Anschluss verbunden und eingeschaltet ist, wird die USB-Schnittstelle in der PC-Software deshalb auch als COM-Port angezeigt. Der C-877 verwendet für diese Schnittstelle die Baudrate 115200.

Im Folgenden ist das Vorgehen für PIMikroMove® beschrieben.

Informationen zur Herstellung der Kommunikation auf Linux-Systemen finden Sie in der Technical Note "PI Software on ARM-Based Platforms", A000T0089 (S. 3).

#### 6.3.1 Kommunikation über USB-Schnittstelle 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. 49).
- ✓ Der C-877 ist an die USB-Schnittstelle des PC angeschlossen (S. 47).
- ✓ Der C-877 ist eingeschaltet (S. 49).



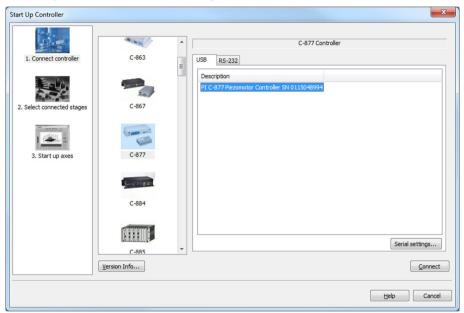
- ✓ Der PC ist eingeschaltet.
- ✓ Die benötigte Software sowie die USB-Treiber sind auf dem PC installiert (S. 44).
- ✓ Sie haben das Handbuch der verwendeten PC-Software gelesen und verstanden. Links auf die Software-Handbücher finden Sie in der Datei A000T0081 auf dem Datenträger mit der PI-Software.

#### Kommunikation über USB 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 *C-877* aus.
- 3. Wählen Sie auf der rechten Seite des Fensters die Registerkarte USB aus.
- 4. Wählen Sie auf der Registerkarte USB den angeschlossenen C-877 aus.
- 5. Klicken Sie auf *Connect*, um die Kommunikation herzustellen.

Wenn die Kommunikation erfolgreich hergestellt wurde, leitet PIMikroMove® Sie durch die Konfiguration des C-877 für den angeschlossenen Positionierer, siehe "Bewegungen starten" (S. 52).

Wenn die Kommunikation nicht hergestellt werden konnte, suchen Sie in "Störungsbehebung" (S. 191) nach einer Lösung des Problems.



# 6.4 Bewegungen starten

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

- Konfiguration des C-877 für den angeschlossenen Positionierer
- Einschalten des Servomodus
- Ausführen einer Referenzfahrt; Details siehe "Referenzierung" (S. 34).

Dann können erste Testbewegungen des Positionierers durchgeführt werden.

#### **HINWEIS**



#### Auswahl eines falschen Positionierertyps

Die Auswahl eines falschen Positionierertyps in der PC-Software kann Schäden am Positionierer verursachen.

> Stellen Sie sicher, dass der in der PC-Software ausgewählte Positionierertyp mit dem angeschlossenen Positionierer übereinstimmt.

#### **HINWEIS**



#### Schwingungen!

Ungeeignete Einstellungen der Regelparameter des C-877 können den Positionierer zum Schwingen bringen. Schwingungen können den Positionierer und/oder die auf ihm angebrachte Last beschädigen.

- > Befestigen Sie den Positionierer und alle Lasten ausreichend.
- Wenn der angeschlossene Positionierer schwingt (ungewöhnliches Laufgeräusch), schalten Sie den Servomodus sofort aus oder trennen Sie den C-877 von der Stromversorgung.
- > Schalten Sie den Servomodus erst wieder ein, nachdem Sie die Einstellungen der Regelparameter des C-877 geändert haben; siehe "Regelparameter optimieren" (S. 56).
- Wenn aufgrund einer sehr hohen Last Schwingungen bereits während der Referenzfahrt auftreten, folgen Sie für die Referenzfahrt den Anweisungen in "Störungsbehebung" (S. 191).

#### **INFORMATION**

Nach dem Herstellen der Kommunikation zwischen C-877 und PC leitet PIMikroMove® Sie durch die Konfiguration des C-877 für den angeschlossenen Positionierer. Die Auswahl der angebotenen Konfigurationsschritte durch PIMikroMove® basiert auf der Auswertung der Werte folgender Parameter aus dem flüchtigen Speicher des C-877:

- **Stage Name** (ID 0x3C): Der Wert wird von PIMikroMove® als Kriterium zum Auffinden eines passenden Parametersatzes in den Positioniererdatenbanken verwendet.
- Stage Type (ID 0x0F000100): Der Wert wurde beim Einschalten des C-877 vom ID-Chip (S. 12) des angeschlossenen Positionierers geladen.

Mögliche Konfigurationsschritte:

Wenn die Werte der Parameter 0x3C und 0x0F000100 identisch sind, geht PIMikroMove®



- davon aus, dass alle Parameter des C-877 bereits an den angeschlossenen Positionierer angepasst sind. Das Fenster **Start up controller** wechselt direkt zum Schritt **Start up axes**, wo die Referenzfahrt gestartet werden kann.
- Wenn die Werte der Parameter 0x3C und 0x0F000100 nicht identisch sind, öffnet sich das Fenster Stage Type Configuration. Über die Schaltfläche Yes, configure for ... kann ein passender Parametersatz aus einer Positioniererdatenbank in den C-877 geladen werden. Nach dem Laden des Parametersatzes wechselt das Fenster Start up controller zum Schritt Start up axes. Wenn kein passender Parametersatz in den Positioniererdatenbanken vorhanden ist, enthält das Fenster Stage Type Configuration einen entsprechenden Hinweis.
- Wenn der Wert des Parameters 0x0F000100 leer ist, weil der Positionierer z. B. keinen ID-Chip hat, wechselt das Fenster Start up controller zum Schritt Select connected stages.

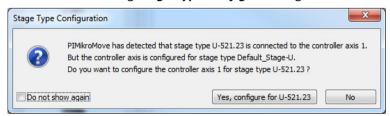
#### Voraussetzungen

- ✓ Sie haben die allgemeinen Hinweise zur Inbetriebnahme gelesen und verstanden (S. 49).
- ✓ PIMikroMove® ist auf dem PC installiert (S. 44).
- ✓ Sie haben das Handbuch für PIMikroMove® gelesen und verstanden. Links auf die Softwarehandbücher finden Sie in der Datei A000T0081 auf dem Datenträger mit der PI-Software.
- ✓ Sie haben die neuesten Versionen der Positioniererdatenbank(en) auf dem PC installiert (S. 45).
- ✓ Wenn Sie für Ihren Positionierer eine kundenspezifische Positioniererdatenbank von PI erhalten haben, dann haben Sie diese Datenbank auf Ihrem PC installiert (S. 46).
- ✓ Sie haben den Positionierer so installiert, wie er in Ihrer Anwendung eingesetzt wird (entsprechende Last, Ausrichtung und Befestigung).
- ✓ Sie haben den Positionierer an den C-877 angeschlossen (S. 46).
- ✓ Sie haben die Kommunikation zwischen dem C-877 und dem PC mit PIMikroMove® hergestellt (S. 50).

### Bewegungen starten mit PIMikroMove®

- 1. Wenn einer der beiden folgenden Punkte zutrifft, konfigurieren Sie den C-877 für den angeschlossenen Positionierer:
  - Der Dialog Stage Type Configuration hat sich geöffnet.
  - Im Fenster Start up controller wird der Schritt Select connected stages angezeigt.

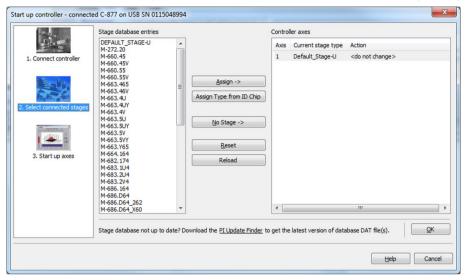
Wenn sich der Dialog Stage Type Configuration geöffnet hat:





 Klicken Sie auf die Schaltfläche Yes, configure for ..., um den passenden Parametersatz aus einer Positioniererdatenbank in den C-877 zu laden. Dies öffnet den Dialog Save all changes permanently?

Wenn im Fenster Start up controller der Schritt Select connected stages angezeigt wird:



- a) Wählen Sie den passenden Positionierertyp aus. Sie haben zwei Möglichkeiten:
  - Klicken Sie auf Assign Type from ID Chip.
  - Markieren Sie den passenden Positionierertyp in der Liste Stage database entries, und klicken Sie auf Assign.
- b) Bestätigen Sie die Auswahl mit **OK**, um die Parametereinstellungen für den ausgewählten Positionierertyp aus der Positioniererdatenbank in den C-877 zu laden. Dies öffnet den Dialog **Save all changes permanently?**.
- Geben Sie im Dialog Save all changes permanently? an, wie Sie die Parametereinstellungen in den C-877 laden wollen:
  - Temporär laden: Klicken Sie auf Keep the changes temporarily, um die Parametereinstellungen in den flüchtigen Speicher des C-877 zu laden. Die Einstellungen gehen beim Ausschalten oder Neustart des C-877 verloren.
  - Als Standardwerte laden: Klicken Sie auf Save all settings permanently on controller, um die Parametereinstellungen in den permanenten Speicher des C-877 zu laden. Die Einstellungen sind nach dem Einschalten oder Neustart des C-877 sofort vorhanden und müssen nicht erneut geladen werden.

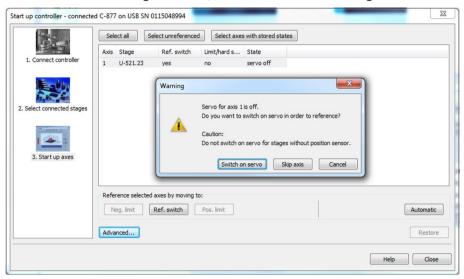
Das Fenster Start up controller wechselt zum Schritt Start up axes.

- 3. Führen Sie im Schritt **Start up axes** die Referenzfahrt für die Achse aus, damit der Controller die absolute Achsenposition kennt: Sie haben folgende Optionen (vom Positionierer/Controller nicht unterstützte Optionen sind nicht vorhanden oder nicht aktivierbar):
  - 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.



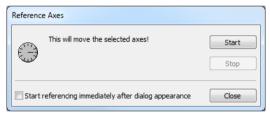
 Wenn Sie die Referenzfahrt zum positiven Endschalter starten wollen, klicken Sie auf **Pos. limit**.

Falls eine Warnmeldung erscheint, dass der Servomodus ausgeschaltet ist:



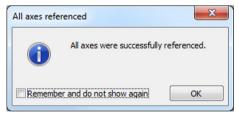
a) Schalten Sie den Servomodus durch einen Klick auf die Schaltfläche **Switch on servo** ein.

Wenn nach dem Einschalten des Servos der Dialog Reference Axes angezeigt wird:



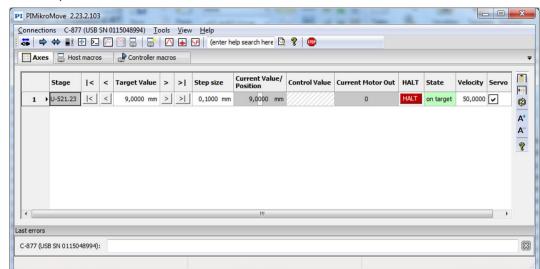
b) Klicken Sie auf die Schaltfläche Start. Die Achse führt die Referenzfahrt aus.

Wenn nach erfolgreicher Referenzfahrt die entsprechende Meldung angezeigt wird:



- c) Schließen Sie die Meldung mit OK.
- 4. Nach erfolgreicher Referenzfahrt schließen Sie das Fenster **Start up controller** durch einen Klick auf **Close**.





Das Hauptfenster von PIMikroMove® öffnet sich.

5. Starten Sie einige Testbewegungen der Achse.

Im Hauptfenster von PIMikroMove® können Sie z. B. Bewegungen um eine bestimmte Strecke (Vorgabe in Spalte *Step size*) oder zu den Grenzen des Stellwegs ausführen, indem Sie auf die entsprechenden Pfeiltasten für die Achse klicken.



# 6.5 Regelparameter optimieren

Die dynamischen Eigenschaften des Systems (z.B. Starten der Bewegung, Schwingungsverhalten während der Bewegung, Überschwingen und Einschwingzeit) werden durch Anpassung der Parameter für Regelalgorithmus und Dynamikprofilkorrekturen optimiert. Die optimalen Einstellungen hängen von Ihrer Anwendung und Ihren Wünschen ab.

Typischerweise erfolgt die Optimierung empirisch, d. h., das Verhalten des Positionierers wird bei verschiedenen Werten im geregelten Betrieb beobachtet. Die Optimierung wird über die folgenden Parameter vorgenommen:

- P-, I-, D-Terme und I-Limit der Parametergruppen 0 bis 4 (IDs 0x4n1, 0x4n2, 0x4n3, 0x4n4; n nimmt je nach Parametergruppe einen Wert von 0 bis 4 an)
- Parameter für das Umschalten zwischen den Parametergruppen 0 bis 4, z. B.
   Fenstergrenzen (IDs 0x4n6, 0x4n7; n nimmt je nach Parametergruppe einen Wert von 0 bis 4 an)
- Geschwindigkeitsabhängiger Offset und Offsets für die positive und negative Bewegungsrichtung (IDs 0x48, 0x33, 0x34)



### **INFORMATION**

Erfahrene Anwender können auch die Zweiphasenansteuerung aktivieren (S. 21), um das Einschwingverhalten zu verbessern.

Im Folgenden ist das Vorgehen zur Optimierung der dynamischen Eigenschaften des Systems für PIMikroMove® beschrieben. Die Zweiphasenansteuerung wird dabei **nicht** berücksichtigt.

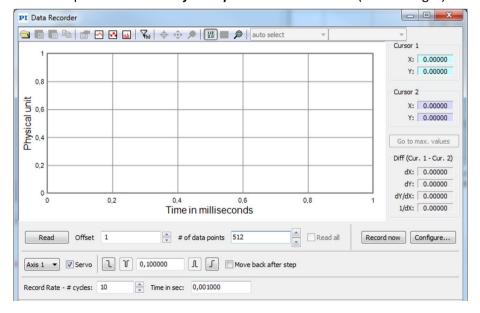
#### Voraussetzung

- ✓ Sie haben den Positionierer so installiert, wie er in Ihrer Anwendung eingesetzt wird (entsprechende Last, Ausrichtung und Befestigung).
- ✓ Sie haben mit PIMikroMove® erste Bewegungen gestartet (S. 52).
- ✓ Alle Geräte sind noch betriebsbereit.

## Regelparameter prüfen: Sprungantwort messen

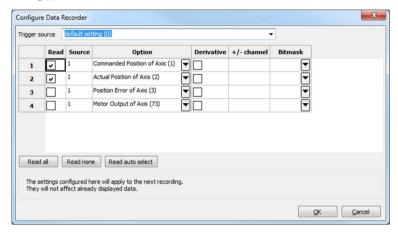
Mit der Aufzeichnung der Sprungantwort ermitteln Sie das Einschwingverhalten des Positionierers im geregelten Betrieb.

- 1. Öffnen Sie im Hauptfenster von PIMikroMove® das Fenster *Data Recorder* über den Menüeintrag *C-877 > Show data recorder*.
- 2. Schalten Sie mit dem Kontrollkästchen Servo den Servomodus ein (Häkchen setzen).
- 3. Konfigurieren Sie den Datenrekorder.
  - a) Stellen Sie als Amplitude des auszuführenden Sprungs einen Wert ein, der für Ihre Anwendung typisch ist, z. B. 0,100000 (Angabe in physikalischer Einheit, z. B. Millimeter oder Grad).
  - b) Stellen Sie für die Aufzeichnungsrate des Datenrekorders im Feld **Record Rate #** cycles den Wert 10 ein.
  - c) Stellen Sie für die Anzahl der für die grafische Darstellung auszulesenden Datenpunkte im Feld # of data points den Wert 1024 (oder weniger) ein.





d) Klicken Sie auf die Schaltfläche *Configure...* und stellen Sie im Fenster *Configure Data Recorder* sicher, dass als aufzuzeichnende Größen "Commanded Position of Axis" und "Actual Position of Axis" ausgewählt sind. Schließen Sie das Fenster mit *OK*.



4. Starten Sie im Fenster **Data Recorder** 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 (siehe untenstehende Beispiele).
  - Wenn nötig, vergrößern Sie die Darstellung, indem Sie auf die Schaltfläche klicken und mit gedrückter linker Maustaste den zur Lupe umgewandelten Mauszeiger über einen Bereich der grafischen Darstellung ziehen (ein Klick mit der rechten Maustaste in das Grafikfeld verkleinert die Darstellung wieder auf die ursprüngliche Größe).

## Beispiele für Sprungantworten:

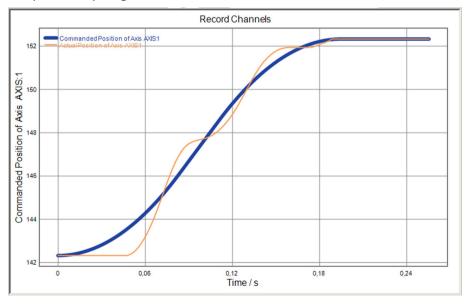


Abbildung 12: Unpassende Einstellungen, infolgedessen Schwingungen und inakzeptables Einschwingverhalten



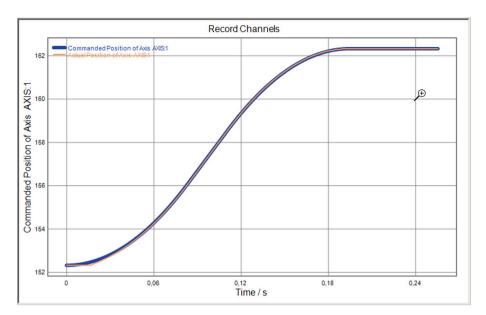


Abbildung 13: Einschwingverhalten schon fast optimal, aber Verhalten beim Start der Bewegung noch nicht zufriedenstellend (Offseteinstellungen müssen noch optimiert werden)

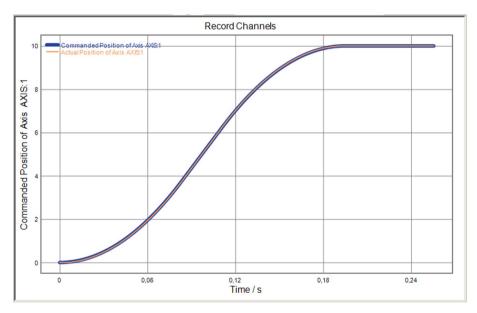


Abbildung 14: Optimale dynamische Eigenschaften des Systems, keine Anpassungen notwendig

Wenn das Ergebnis zufriedenstellend ist:

 Sie verfügen bereits über optimale Parametereinstellungen und brauchen nichts weiter zu unternehmen.

Wenn das Ergebnis nicht zufriedenstellend ist:

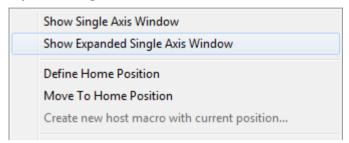
 Optimieren Sie die Parameter f
ür die dynamischen Eigenschaften des Systems, siehe unten.

Version: 2.0.0

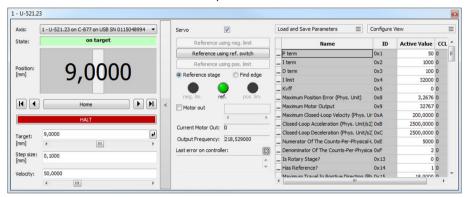


## Regelparameter optimieren

 Öffnen Sie im Hauptfenster von PIMikroMove® das erweiterte Einzelachsen-Fenster für den angeschlossenen Positionierer, indem Sie mit der rechten Maustaste auf die entsprechende Zeile der Registerkarte Axes klicken und im Kontextmenü Show Expanded Single Axis Window auswählen.



2. Geben Sie neue Werte für die anzupassenden Parameter ein:



- a) Wenn der zu ändernde Parameter nicht in der Liste auf der rechten Seite des Fensters enthalten ist, klicken Sie auf **Configure view > Select parameters...** und fügen ihn zur Liste hinzu. Sie können auch bestimmte Gruppen von Parametern oder alle achsenbezogenen Parameter einblenden.
- b) Tippen Sie den neuen Parameterwert in das entsprechende Eingabefeld in der Spalte *Active Value* der Liste ein.
- c) Drücken Sie auf der Tastatur des PC die Enter-Taste oder klicken Sie mit der Maus außerhalb des Eingabefeldes, um den Parameterwert in den flüchtigen Speicher des Controllers zu übertragen. Anmerkung: Wenn ein Parameterwert im flüchtigen Speicher (Spalte Active Value) vom Parameterwert im permanenten Speicher (Spalte Startup Value) abweicht, ist die Zeile in der Liste farbig markiert.
- 3. Zeichnen Sie im Fenster *Data Recorder* erneut die Sprungantwort des Positionierers auf.

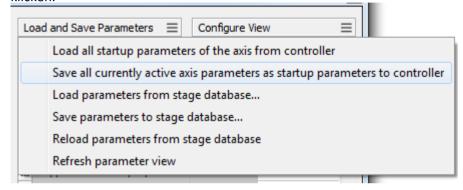
Wenn das Ergebnis nicht zufriedenstellend ist:

 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 Parametersatz in der Positioniererdatenbank auf dem PC, indem Sie auf *Load and Save Parameters -> Saveparameters to stage database...* klicken, siehe "Positionierertyp anlegen oder ändern" (S. 172).
- Übertragen Sie die aktuellen Werte der aufgelisteten Parameter aus dem flüchtigen in den permanenten Speicher des C-877, indem Sie auf Load and Save Parameters -> Save all currently active axis parameters as startup parameters to controller klicken.





# 7 Betrieb

## 7.1 Schutzfunktionen des C-877

# 7.1.1 Schutz vor Überhitzung

Wenn ein hoher Stellwert über längere Zeit gesetzt bleibt, kann sich der angeschlossene Positionierer erwärmen. Überhitzen kann zu Schäden am Positionierer führen.

Durch den Parameter *PID Maximum Output Time (s)* (ID 0x7B) wird die maximale Zeitdauer vorgegeben, für die ein hoher Stellwert im geregelten Betrieb gesetzt sein darf. Ein hoher Stellwert liegt vor, wenn gilt:

Aktueller Betrag des Stellwerts ≥ 95 % von *Maximum Motor Output* (ID 0x9).

Wenn der hohe Stellwert nach Ablauf der maximalen Zeitdauer noch gesetzt ist, reagiert der C-877 wie folgt, um das System vor Schäden zu schützen:

- Der Stellwert wird für die betroffene Achse auf den Wert Null gesetzt.
- Der Servomodus wird für die betroffene Achse ausgeschaltet.

# 7.1.2 Verhalten bei Bewegungsfehler

Ein Bewegungsfehler liegt vor, wenn sich die aktuelle Position der Achse bei Stellwertänderungen nicht ändert.

Ein Bewegungsfehler kann im geregelten oder ungeregelten Betrieb auftreten.

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

Bewegungsfehler können z. B. folgende Ursachen haben:

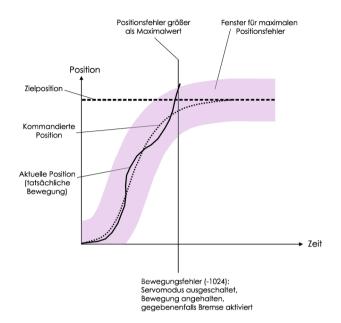
- Störung des Antriebs
- Störung des Positionssensors
- Störung des Positionierers

Wenn ein Bewegungsfehler auftritt, reagiert der C-877 wie folgt, um das System vor Schäden zu schützen:

- Der Servomodus wird für die Achse ausgeschaltet.
- Die Bewegung wird gestoppt.
- Der Fehlercode -1024 wird ausgegeben.

Stellen Sie danach für den C-877 die Betriebsbereitschaft wieder her.





# 7.1.3 Verhalten bei Systemfehler

Ein Systemfehler liegt vor, wenn der C-877 nicht mehr ansprechbar ist.

Die Ursache für einen Systemfehler kann z. B. ein Speicherüberlauf in der Firmware des C-877 sein.

Wenn ein Systemfehler auftritt, reagiert der C-877 wie folgt:

 Die Sicherheitsfunktion Watchdog Timer veranlasst nach einer gewissen Wartezeit einen Neustart des C-877.

## 7.1.4 Betriebsbereitschaft wiederherstellen

- 1. Senden Sie den Befehl ERR?, um den Fehlercode auszulesen.
  - Der Befehl ERR? setzt den Fehlercode bei der Abfrage auf null zurück.
- 2. Überprüfen Sie Ihr System und vergewissern Sie sich, dass folgende Punkte erfüllt sind:
  - Die Achse kann gefahrlos bewegt werden.
  - Der C-877 ist nicht überhitzt (interne Temperatur beträgt maximal 65 °C).
- 3. Wenn nach Fehler oder Überhitzung der Servomodus ausgeschaltet wurde:
  - Schalten Sie mit dem Befehl SVO den Servomodus für die Achse ein.

Beim Einschalten des Servomodus wird die Zielposition auf die aktuelle Achsenposition gesetzt.



## 7.2 Datenrekorder

#### 7.2.1 Datenrekorder einrichten

Der C-877 enthält einen Echtzeit-Datenrekorder. Der Datenrekorder kann z. B. die aktuelle Position der Achse aufzeichnen.

Die aufgezeichneten Daten werden temporär in 4 Datenrekordertabellen mit jeweils 1024 Punkten gespeichert. Jede Datenrekordertabelle enthält die Daten 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.

#### **INFORMATION**

Die Einstellungen zur Einrichtung des Datenrekorders lassen sich nur im flüchtigen Speicher des C-877 ändern. Nach dem Einschalten oder dem Neustart des C-877 sind Standardeinstellungen aktiv, wenn nicht durch ein Startup-Makro bereits eine Konfiguration erfolgt.

#### Allgemeine Informationen über den Datenrekorder auslesen

> Senden Sie den Befehl HDR? (S. 107).

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

#### Aufzuzeichnende Daten konfigurieren

Sie können den Datenrekordertabellen die Datenquellen und die Aufzeichnungsoptionen zuordnen.

- ➤ Senden Sie den Befehl DRC? (S. 97), um die aktuelle Konfiguration auszulesen.

  Datenrekordertabellen mit Aufzeichnungsoption 0 sind deaktiviert, d. h. es wird nichts aufgezeichnet. In der Standardeinstellung zeichnen die Datenrekordertabellen des C-877 Folgendes auf:
  - Datenrekordertabelle 1: Aufzeichnungsoption 1: kommandierte Position der Achse
  - Datenrekordertabelle 2: Aufzeichnungsoption 2: aktuelle Position der Achse
  - Datenrekordertabelle 3: Aufzeichnungsoption 3: Positionsfehler der Achse
  - Datenrekordertabelle 4: Aufzeichnungsoption 73: Stellwert der Achse
- ➤ Konfigurieren Sie den Datenrekorder mit dem Befehl DRC (S. 96).

#### Auslösen der Aufzeichnung konfigurieren

Sie können festlegen, wie die Aufzeichnung ausgelöst werden soll.

- Fragen Sie mit DRT? (S. 100) die aktuelle Triggeroption ab.
- Andern Sie die Triggeroption mit dem Befehl DRT (S. 99). Die Triggeroption gilt für alle Datenrekordertabellen, deren Aufzeichnungsoption nicht auf 0 eingestellt ist.



# Aufzeichnungsrate einstellen

Senden Sie den Befehl RTR? (S. 124), 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 Servozykluszeit des C-877 beträgt 100  $\mu$ s.

Andern Sie die Aufzeichnungsrate mit dem Befehl RTR (S. 123).

Mit zunehmender Aufzeichnungsrate erhöhen Sie die maximale Dauer der Datenaufzeichnung.

# 7.2.2 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. 134) gestartet wird.

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

# 7.2.3 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? (S. 98) aus. Die Daten werden im GCS-Array-Format ausgegeben (siehe Benutzerhandbuch SM146E).
- Fragen Sie die Anzahl der in der letzten Aufzeichnung enthaltenen Punkte mit dem Befehl DRL? (S. 97) ab.



# 7.3 Controllermakros

# 7.3.1 Übersicht: Makrofunktionalitäten und Beispielmakros

Der C-877 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-877 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. 80) können für das Makro und im Makro selbst gesetzt und in verschiedenen Operationen verwendet werden.

In diesem Handbuch finden Sie Beispielmakros für folgende Aufgaben:

- Achse hin und her bewegen (S. 71)
- Achse mit variablem Verfahrweg hin und her bewegen (S. 72)
- Achse durch Startup-Makro für den geregelten Betrieb vorbereiten (S. 74)

# 7.3.2 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. 89)	ADD <variable> <float1> <float2></float2></float1></variable>	Addiert zwei Werte und speichert das Ergebnis als Variable (S. 80). Für lokale Variablen nur innerhalb von Makros verwendbar.			
CPY (S. 91)	CPY <variable> <cmd?></cmd?></variable>	Kopiert eine Antwort auf einen Befehl in eine Variable (S. 80). Für lokale Variablen nur innerhalb von Makros verwendbar.			
DEL (S. 93)	DEL <uint></uint>	Kann nur in Makros verwendet werden. Verzögert um <uint> Millisekunden.</uint>			
JRC (S. 110)	JRC <jump> <cmd?> <op> <value></value></op></cmd?></jump>	Kann nur in Makros verwendet werden. Löst einen relativen Sprung des Makroausführungszeigers in Abhängigkeit von einer Bedingung aus.			



Befehl	Syntax	Funktion				
MAC (S. 111)	MAC BEG <macroname></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></macroname>	Legt das angegebene Makro als Startup-Makro fest.				
	MAC DEF?	Fragt das Startup-Makro ab.				
	MAC DEL <macroname></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>]]</string2></string1></uint></macroname>	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.</string2></string1>				
	MAC START <macroname> [<string1> [<string2>]]</string2></string1></macroname>	Startet eine Ausführung des angegebenen Makros. Mit <string1> und <string2> können die Werte lokaler Variablen für das Makro gesetzt werden.</string2></string1>				
MAC? (S. 114)	MAC? [ <macroname>]</macroname>	Listet alle Makros oder den Inhalt des angegebenen Makros auf.				
MEX (S. 116)	MEX <cmd?> <op> <value></value></op></cmd?>	Kann nur in Makros verwendet werden. Stoppt die Makroausführung in Abhängigkeit von einer Bedingung.				
RMC? (S. 121)	RMC?	Listet die aktuell laufenden Makros auf.				
VAR (S. 138)	VAR <variable> <string></string></variable>	Setzt eine Variable (S. 80) auf einen bestimmten Wert oder löscht sie. Für lokale Variablen nur innerhalb von Makros verwendbar.				
VAR? (S. 140)	VAR? [{ <variable>}]</variable>	Gibt Variablenwerte zurück.				
WAC (S. 142)	WAC <cmd?> <op> <value></value></op></cmd?>	Kann nur in Makros verwendet werden. Wartet, bis eine Bedingung erfüllt ist.				
#8 (S. 86)	-	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				
<b>Ignore Macro Error?</b> 0x72	Legt fest, ob das Controllermakro gestoppt wird, wenn bei dessen Ausführung ein Fehler auftritt.				
	■ 0 = Bei Fehler Makro anhalten (Standard)				
	■ 1 = Fehler ignorieren				

# 7.3.3 Mit Makros arbeiten

Die Arbeit mit Makros umfasst Folgendes:

- Aufzeichnen von Makros (S. 70)
- Starten der Makroausführung (S. 71)
- Stoppen der Makroausführung (S. 73)
- Einrichten eines Startup-Makros (S. 74)
- Löschen von Makros (S. 75)

# **INFORMATION**

Der C-877 kann bis zu 32 Makros gleichzeitig speichern. Maximal 5 Verschachtelungsebenen sind in Makros möglich.

# **INFORMATION**

Grundsätzlich können alle GCS-Befehle (S. 77) Bestandteil eines Makros werden. Ausnahmen:

- RBT für den Neustart des C-877
- 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**

Um die Anwendung von Makros flexibler zu gestalten, können Sie in Makros lokale und globale Variablen verwenden. Weitere Informationen siehe "Variablen" (S. 80).



### **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. 80) in Makros, und geben Sie beim Starten der Makroausführung die entsprechenden Variablenwerte an.
- Wenden Sie sich an unseren Kundendienst (S. 195), wenn der C-877 unerwartetes Verhalten zeigt.

### **INFORMATION**

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

### **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

- 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-877 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-877 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 pr
  üfende 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 END

MAC BEG macro3

MAC START macro1

MAC START macro2

MAC END
```

# 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. 167).

- 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 Verfahrweg 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 Verfahrweg 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-877 erneut an, da sie nur in den flüchtigen Speicher des C-877 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.

### 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-877 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 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 END

MAC BEG startcl SVO 1 1 DEL 1000 FNL 1



MAC DEF startcl

# **INFORMATION**

Bei Verwendung dieses Makros sollten die Parametereinstellungen des C-877 im permanenten Speicher an den angeschlossenen Positionierer angepasst sein. Alternativ können die Parametereinstellungen im flüchtigen Speicher auch durch das Startup-Makro gesetzt werden. Weitere Informationen siehe "Anpassen von Einstellungen" (S. 167).

### 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.

### 7.3.4 Controllermakros sichern und laden

Das Sichern von Controllermakros auf dem PC kann z. B. vor der Aktualisierung der Firmware (S. 187) sinnvoll sein.

## **INFORMATION**

Für das Sichern und Laden von Controllermakros wird die Verwendung der Registerkarte *Controller macros* in PIMikroMove® empfohlen. Eine detaillierte Beschreibung der Registerkarte finden Sie imPIMikroMove® Handbuch.

# Controllermakros mit PIMikroMove® auf dem PC sichern

- 1. Wählen Sie im Hauptfenster von PIMikroMove® die Registerkarte Controller macros.
- Wählen Sie in der Liste *Macros on controller* die Makros aus, die Sie auf den PC sichern wollen:
  - Zur Auswahl eines einzelnen Makros klicken Sie den gewünschten Listeneintrag an.
  - Zur Auswahl mehrerer Makros klicken Sie mit gedrückter Umschalt-Taste (Shift) alle gewünschten Listeneinträge an.
  - Um die Auswahl aufzuheben, klicken Sie auf eine freie Fläche in der Liste.

Durch die Auswahl eines oder mehrerer Makros wird die Schaltfäche (Save selected macros to PC) aktiv.

3. Speichern Sie die ausgewählten Makros auf dem PC:



- a) Klicken Sie auf die Schaltfläche , um ein Fenster zur Verzeichnisauswahl zu öffnen.
- b) Wählen Sie das Verzeichnis auf dem PC aus, in dem Sie die Makros speichern wollen.
- c) Klicken Sie auf Speichern.

Die Makros werden als Textdateien (<Makroname>.txt) im ausgewählten Verzeichnis des PC gespeichert.

# Controllermakros mit PIMikroMove® vom PC in den C-877 laden

- 1. Wählen Sie im Hauptfenster von PIMikroMove® die Registerkarte *Controller macros*.
- 2. Laden Sie Makros vom PC in den C-877:
  - a) Klicken Sie auf die Schaltfläche 🎒 , um ein Dateiauswahlfenster zu öffnen.
  - b) Wählen Sie im Dateiauswahlfenster die Textdateien (<Makroname>.txt) aus, deren Inhalt Sie als Makro vom PC in den C-877 laden wollen.
  - c) Klicken Sie auf Öffnen.

Für jede ausgewählte Textdatei (<Makroname>.txt) wird der Inhalt als Makro <Makroname> in den C-877 geladen.



# 8 GCS-Befehle

# 8.1 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 befehlsspezifischer 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.
- LineFeed (Zeilenvorschub, ASCII-Zeichen 10) ist das Standardabschlusszeichen (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.

# 8.2 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 muss mit Zeilenvorschub (LF) beendet werden.

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-877 unterstützten Kennungen sind im Abschnitt "Kommandierbare Elemente" (S. 13) beschrieben.

### Beispiel 1:

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

Senden: MOVSP1SP10.0LF

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: MOVSP1SP17.3SP2SP2.05LF

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: RPALF

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:

Antwort auf diesen Befehl:

Beispiel 5:

Senden: TSP? SP 2 SP 1 LF

Empfangen: 2=-1158.4405 **SP LF** 

1=+0000.0000**LF** 

### **INFORMATION**

Beim C-877 kann nur ein einzelnes Element pro Befehlszeile adressiert werden (z. B. Achse 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.

# 8.3 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 die Senderadresse (0) weggelassen werden. Jedoch sind sowohl die Empfänger- als auch die Senderadresse in der Controllerantwort enthalten. Mehrzeilige Antworten enthalten die Empfänger- und Senderadresse nur in der ersten Zeile.

Weil die Empfängeradresse weggelassen werden kann, wenn der anzusprechende Controller die Adresse 1 hat, kann die Empfängeradresse (1) weggelassen werden. Wenn die Empfängeradresse beim Ansprechen des Controllers weggelassen wird, werden Empfängerund Senderadresse auch in der Antwort des Controllers weggelassen.

Beispiel: Abfrage der Ident-Bezeichnung eines C-877.1U11 mit der Adresse 1

Senden: \*IDN?

### Der Controller antwortet:

(c) 2024 Physik Instrumente (PI) Karlsruhe, C-877.1U11, 0, 1.2.0.0

Senden: 1 \*IDN?

### Derselbe Controller antwortet:

0 1 (c)2024 Physik Instrumente(PI) Karlsruhe, C-877.1U11, 0, 1.2.0.0

# 8.4 Variablen

Für eine flexiblere Programmierung unterstützt der C-877 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 k\u00f6nnen aus den Zeichen A bis Z und 0 bis 9 bestehen. Sie m\u00fcssen 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, wievielmal das Makro ausgeführt werden soll. Weitere Informationen finden Sie in der Beschreibung des Befehls MAC (S. 111).

- 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. 89), CPY (S. 91), MAT (S. 115) und VAR (S. 138) 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, MAT oder VAR angelegt und verändert. Sie können mit dem Befehl VAR gelöscht werden.
- Die Variablenwerte können mit VAR? abgefragt werden.



# 8.5 Befehlsübersicht

Befehl	Argumente	Beschreibung					
#4		Request Status Register (S. 85)					
#5		Request Motion Status (S. 85)					
#7		Request Controller Ready Status (S. 86)					
#8		Query If Macro Is Running (S. 86)					
#24		Stop All Axes (S. 87)					
*IDN?		Get Device Identification (S. 87)					
ACC	{ <axisid> <acceleration>}</acceleration></axisid>	Set Closed-Loop Acceleration (S. 88)					
ACC?	[{ <axisid>}]</axisid>	Get Closed-Loop Acceleration (S. 88)					
ADD	<variable> <float1> <float2></float2></float1></variable>	Add and Save To Variable (S. 89)					
CCL	<level> [<pswd>]</pswd></level>	Set Command Level (S. 90)					
CCL?		Get Command Level (S. 91)					
СРҮ	<variable> <cmd?></cmd?></variable>	Copy Into Variable (S. 91)					
CST?	[{ <axisid>}]</axisid>	Get Assignment Of Stages To Axes (S. 91)					
CSV?		Get Current Syntax Version (S. 92)					
DEC	{ <axisid> <deceleration>}</deceleration></axisid>	Set Closed-Loop Deceleration (S. 92)					
DEC?	[{ <axisid>}]</axisid>	Get Closed-Loop Deceleration (S. 93)					
DEL	<uint></uint>	Delay The Command Interpreter (S. 93)					
DFH	[{ <axisid>}]</axisid>	Define Home Position (S. 94)					
DFH?	[{ <axisid>}]</axisid>	Get Home Position Definition (S. 95)					
DRC	{ <rectableid> <source/> <recoption>}</recoption></rectableid>	Set Data Recorder Configuration (S. 96)					
DRC?	[{ <rectableid>}]</rectableid>	Get Data Recorder Configuration (S. 97)					
DRL?	[{ <rectableid>}]</rectableid>	Get Number Of Recorded Points (S. 97)					
DRR?	[ <startpoint> <numberofpoints> [{<rectableid>}]]</rectableid></numberofpoints></startpoint>	Get Recorded Data Values (S. 98)					
DRT	{ <rectableid> <triggersource> <value>}</value></triggersource></rectableid>	Set Data Recorder Trigger Source (S. 99)					
DRT?	[{ <rectableid>}]</rectableid>	Get Data Recorder Trigger Source (S. 100)					
ERR?		Get Error Number (S. 101)					
FED	{ <axisid> <edgeid> <param/>}</edgeid></axisid>	Find Edge (S. 102)					
FNL	[{ <axisid>}]</axisid>	Fast Reference Move To Negative Limit (S. 103)					
FPL	[{ <axisid>}]</axisid>	Fast Reference Move To Positive Limit (S. 104)					



Befehl	Argumente	Beschreibung					
FRF	[{ <axisid>}]</axisid>	Fast Reference Move To Reference Switch (S. 105)					
FRF?	[{ <axisid>}]</axisid>	Get Referencing Result (S. 106)					
GOH	[{ <axisid>}]</axisid>	Go To Home Position (S. 107)					
HDR?		Get All Data Recorder Options (S. 107)					
HLP?		Get List of Available Commands (S. 108)					
HLT	[{ <axisid>}]</axisid>	Halt Motion Smoothly (S. 108)					
HPA?		Get List Of Available Parameters (S. 109)					
JRC	<jump> <cmd?> <op> <value></value></op></cmd?></jump>	Jump Relatively Depending On Condition (S. 110)					
LIM?	[{ <axisid>}]</axisid>	Indicate Limit Switches (S. 111)					
MAC	<pre><keyword> {<parameter>} insbesondere: BEG <macroname> DEF <macroname> DEF? DEL <macroname> END ERR? FREE? NSTART <macroname> <uint> [<string1> [<string2>]] START <macroname> [<string1> [<string2>]]</string2></string1></macroname></string2></string1></uint></macroname></macroname></macroname></macroname></parameter></keyword></pre>	Call Macro Function (S. 111)					
MAC?	[ <macroname>]</macroname>	List Macros (S. 114)					
MAN?	<cmd></cmd>	Get Help String For Command (S. 115)					
MEX	<cmd?> <op> <value></value></op></cmd?>	Stop Macro Execution Due To Condition (S. 116)					
MOV	{ <axisid> <position>}</position></axisid>	Set Target Position (S. 117)					
MOV?	[{ <axisid>}]</axisid>	Get Target Position (S. 118)					
MVR	{ <axisid> <distance>}</distance></axisid>	Set Target Relative To Current Position (S. 118)					
ONT?	[{ <axisid>}]</axisid>	Get On-Target State (S. 119)					
POS	{ <axisid> <position>}</position></axisid>	Set Real Position (S. 120)					
POS?	[{ <axisid>}]</axisid>	Get Real Position (S. 120)					
RBT		Reboot System (S. 121)					
RMC?		List Running Macros (S. 121)					
RON	{ <axisid> <referenceon>}</referenceon></axisid>	Set Reference Mode (S. 121)					
RON?	[{ <axisid>}]</axisid>	Get Reference Mode (S. 122)					
RPA	[{ <itemid> <pamid>}]</pamid></itemid>	Reset Volatile Memory Parameters (S. 123)					



Befehl	Argumente	Beschreibung				
RTR	<recordtablerate></recordtablerate>	Set Record Table Rate (S. 123)				
RTR?		Get Record Table Rate (S. 124)				
SAI	{ <axisid> <newidentifier>}</newidentifier></axisid>	Set Current Axis Identifiers (S. 125)				
SAI?	[ALL]	Get List Of Current Axis Identifiers (S. 125)				
SEP	<pswd> {<itemid> <pamid> <pamvalue>}</pamvalue></pamid></itemid></pswd>	Set Non-Volatile Memory Parameters (S. 125)				
SEP?	[{ <itemid> <pamid>}]</pamid></itemid>	Get Non-Volatile Memory Parameters (S. 127)				
SMO	{ <axisid> <controlvalue>}</controlvalue></axisid>	Set Open-Loop Control Value (S. 127)				
SMO?	[{ <axisid>}]</axisid>	Get Control Value (S. 129)				
SPA	{ <itemid> <pamid> <pamvalue>}</pamvalue></pamid></itemid>	Set Volatile Memory Parameters (S. 129)				
SPA?	[{ <itemid> <pamid>}]</pamid></itemid>	Get Volatile Memory Parameters (S. 132)				
SRG?	{ <axisid> <registerid>}</registerid></axisid>	Query Status Register Value (S. 132)				
STE	<axisid> <amplitude></amplitude></axisid>	Start Step And Response Measurement (S. 134)				
STP		Stop All Axes (S. 134)				
SVO	{ <axisid> <servostate>}</servostate></axisid>	Set Servo Mode (S. 135)				
SVO?	[{ <axisid>}]</axisid>	Get Servo Mode (S. 136)				
TCV?	[{AxisID}]	Get Commanded Closed-Loop Velocity (S. 136)				
TMN?	[{ <axisid>}]</axisid>	Get Minimum Commandable Position (S. 136)				
TMX?	[{ <axisid>}]</axisid>	Get Maximum Commandable Position (S. 137)				
TNR?		Get Number Of Record Tables (S. 137)				
TRS?	[{ <axisid>}]</axisid>	Indicate Reference Switch (S. 137)				
TVI?		Tell Valid Character Set For Axis Identifiers (S. 138)				
VAR	<variable> <string></string></variable>	Set Variable Value (S. 138)				
VAR?	[{ <variable>}]</variable>	Get Variable Value (S. 140)				
VEL	{ <axisid> <velocity>}</velocity></axisid>	Set Closed-Loop Velocity (S. 140)				
VEL?	[{ <axisid>}]</axisid>	Get Closed-Loop Velocity (S. 141)				
VER?		Get Versions Of Firmware And Drivers (S. 141)				
WAC	<cmd?> <op> <value></value></op></cmd?>	Wait For Condition (S. 142)				
WPA	<pswd> [{<itemid> <pamid>}]</pamid></itemid></pswd>	Save Parameters To Non-Volatile Memory (S. 143)				



# 8.6 Befehlsbeschreibungen für GCS 2.0

# #4 (Request Status Register)

Beschreibung: Fragt die Systemstatusinformation 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. 132), 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-877 ist die Antwort die Summe der folgenden

Codes in Hexadezimalformat:

Bit	15	14	13	12	11	10	9	8
Beschrei- bung	On- Target- Status	Bestimmt den Referenz- wert	In Bewe- gung	Servo- modus Ein	1	1	1	Fehler- flag

Bit	7	6	5	4	3	2	1	0
Beschrei- bung	-	-	-	-	Achse ist referenzie rt	Pos. End- schal- ter	Refe- renz- schal- ter	Neg. End- schal- ter

Beispiel: Senden: #4

Empfangen: 0x900A

= Binärformat: **1**00**1**000**0**000010**1**0

Hinweis: Die Antwort wird im Hexadezimalformat

angegeben. Sie besagt (fett angezeigt):

Die Achse ist an der Zielposition (On-Target-Status; Bit 15),

der Servomodus ist eingeschaltet (Bit 12), es ist kein Fehler aufgetreten (Bit 8), die Achse wurde bereits referenziert (Bit 3), der Positionierer befindet sich an der Position des

Referenzschalters (Bit 1).



# **#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

• • •

0 gibt an, dass die Bewegung aller Achsen abgeschlossen

ist.

# **#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. 85) 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 Referenzierungsfahrt 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. 134), 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. 117), MVR (S. 118), GOH (S. 107), STE (S. 134), SMO (S. 127)), Befehle zur Referenzierung (FNL (S. 103), FPL (S. 104), FRF (S. 105)) und Makros (MAC (S. 111)) verursacht wird. Stoppt auch die Makroausführung.

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

HLT (S. 108) 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 einzeiliger

Text mit Controllername, Seriennummer und

Firmwareversion

Hinweise: Beim C-877 antwortet \*IDN? etwa Folgendes:

(c)2024 Physik Instrumente (PI) GmbH &

Co. KG, C-877, 113059603, 1.005



### ACC (Set Closed-Loop Acceleration)

Beschreibung: Setzt die Beschleunigung für die angegebenen Achsen.

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 s².

Antwort: Keine

Fehlersuche: Unzulässige Achsenkennungen

Hinweise: Die ACC-Einstellung wird nur wirksam, wenn sich die

angegebene Achse im geregelten Betrieb befindet

(Servomodus EIN).

Der kleinstmögliche Wert für < Acceleration > ist 0.

ACC ändert den Wert des Parameters Closed-Loop

Acceleration (Phys. Unit/s²) (ID 0xB) im flüchtigen Speicher des C-877. Der Parameterwert kann mit WPA (S. 143) als Standard gespeichert werden, Details siehe "Anpassen von

Einstellungen" (S. 167).

Der maximale Wert, der mit dem Befehl ACC gesetzt werden kann, wird durch den Parameter **Maximum Closed-Loop Acceleration (Phys. Unit/s**<sup>2</sup>) (ID 0x4A)

angegeben.

### ACC? (Get Closed-Loop Acceleration)

Beschreibung: Fragt den mit ACC (S. 88) gesetzten Wert der

Beschleunigung 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 ist, in physikalischen Einheiten pro s².

# **ADD (Add And Save To Variable)**

Beschreibung: Addiert zwei Werte und speichert das Ergebnis als Variable

(S. 80).

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

Hinweis: ADD kann nur in Makros verwendet werden.

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

Variable C gespeichert:

Senden: 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

## **CCL (Set Command Level)**

Beschreibung: Ändert die aktive "Befehlsebene" und bestimmt somit die

Verfügbarkeit von Befehlen und von Schreibzugriff auf

Systemparameter.

Format: CCL <Level> [<PSWD>]

Argumente: <Level> ist eine Befehlsebene des Controllers

<PSWD> ist das Kennwort, das für den Wechsel in die

entsprechende Befehlsebene erforderlich ist

Es gelten folgende Befehlsebenen und Kennwörter:

Level = 0 ist die Standardeinstellung, alle Befehle, die dem "normalen" Benutzer zur Verfügung gestellt werden, sowie der Lesezugriff auf alle Parameter sind zugänglich, kein

Kennwort erforderlich.

Level = 1 fügt zusätzliche Befehle und den Schreibzugriff für Parameter der Ebene 1 hinzu (Befehle und Parameter der Ebene 0 sind inbegriffen). Das erforderliche Kennwort

lautet "advanced".

Level > 1 ist nur für Servicepersonal von PI vorgesehen. Die Benutzer können nicht zu einer Ebene > 1 wechseln. Wenn Sie Probleme mit Parametern der Ebene 2 oder höher haben sollten, wenden Sie sich an den Kundendienst (S.

195).

Antwort: Keine

Fehlersuche: Ungültiges Kennwort

Hinweise: Beim C-877 legen die Befehlsebenen nur das Schreibrecht

auf die Parameter fest. Die Verfügbarkeit der Befehle des C-877 ist unabhängig von der aktiven Befehlsebene.

HPA? (S. 109) listet die Parameter einschließlich der Information darüber, welche Befehlsebene Schreibzugriff

auf sie erlaubt. Weitere Informationen zur

Parameterverwendung siehe "Anpassen von Einstellungen"

(S. 167).

Nach dem Einschalten oder Neustart des Controllers ist die



aktive Befehlsebene immer 0.

# **CCL? (Get Command Level)**

Beschreibung: Fragt die aktive "Befehlsebene" ab.

Format: CCL?

Argumente: Keine

Antwort: <Level> ist die aktuell aktive Befehlsebene; uint.

Hinweise: <Level> sollte 0 oder 1 sein.

<Level> = 0 ist die Werkseinstellung, auf Parameter der Ebene 0 besteht Schreibzugriff, auf alle Parameter besteht

Lesezugriff

<Level> = 1 lässt Schreibzugriff für Parameter der Ebene 1

zu (Parameter von Ebene 0 sind inbegriffen)

### **CPY (Copy Into Variable)**

Beschreibung: Kopiert eine Antwort auf einen Befehl in eine Variable (S.

80).

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

# CST? (Get Assignment Of Stages To Axes)

Beschreibung: Fragt den Namen des Positionierertyps 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 Positionierertyps ist, der der Achse

zugewiesen ist.

Hinweise: Der Positionierername wird vom Parameter **Stage Name** 

gelesen (ID 0x3C). Wenn der Parameter den Wert

NOSTAGE hat, ist die Achse "deaktiviert". Eine deaktivierte Achse ist nicht für achsenbezogene Befehle zugänglich (z.B. Bewegungsbefehle oder Positionsabfragen). Die Kennung einer deaktivierten Achse kann nur mit SAI? ALL

abgefragt werden.

Sie können mit SPA (S. 129) oder SEP (S. 125) den Wert des

Parameters 0x3C gezielt auf den Namen Ihres

Positionierers setzen. Da die PC-Software von PI den Parameterwert jedoch zur Konfiguration des C-877 für den angeschlossenen Positionierer verwendet (S. 52), wird die manuelle Änderung mit SPA oder SEP nicht empfohlen.

### **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

# **DEC (Set Closed-Loop Deceleration)**

Beschreibung: Setzt die Abbremsung für die angegebenen Achsen.

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<sup>2</sup>.

Antwort: Keine



Fehlersuche: Unzulässige Achsenkennungen

Hinweise: Die DEC-Einstellung wird nur wirksam, wenn sich die

angegebene Achse im geregelten Betrieb befindet

(Servomodus EIN).

Der kleinstmögliche Wert für < Deceleration > ist 0.

DEC ändert den Wert des Parameters *Closed Loop* 

**Deceleration (Phys. Unit/s²)** (ID 0xC) im flüchtigen Speicher des C-877. Der Parameterwert kann mit WPA (S. 143) als Standard gespeichert werden, Details siehe "Anpassen von

Einstellungen" (S. 167).

Der maximale Wert, der mit dem Befehl DEC gesetzt werden kann, wird durch den Parameter *Maximum Closed-Loop Deceleration (Phys. Unit/s*<sup>2</sup>) (ID 0x4B)

angegeben.

## **DEC? (Get Closed-Loop Deceleration)**

Beschreibung: Fragt den mit DEC (S. 92) gesetzten Wert der Abbremsung

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 ist, in

physikalischen Einheiten pro s<sup>2</sup>.

# **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. 111) und im Abschnitt "Controllermakros"

(S. 66).

## **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. 120) (Abfrage der aktuellen Position)

 TMN? (S. 136) (Abfrage der kleinsten kommandierbaren Position)

 TMX? (S. 137) (Abfrage der größten kommandierbaren Position)

DFH ändert **nicht** die Werte der Parameter für die Definition von Stellweg und Verfahrbereichsgrenzen (S.

31).

Der Offset wird in folgenden Fällen auf null zurückgesetzt:

 Beim Einschalten und Neustart des C-877: für alle Achsen

Bei der Referenzierung: 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 den Positionswert ab, der für die angegebene Achse

aktuell als Offset zum Verschieben der Nullposition

verwendet wird.

Werden alle Argumente weggelassen, wird der

Positionswert aller Achsen abgefragt.

Format: DFH? [{<AxisID>}]

Argumente: <AxisID> ist eine Achse des Controllers

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

wobei

<PositionOffset> ist die Achsenposition, die zum Zeitpunkt der Verarbeitung des letzten DFH-Befehls gültig war. Dieser Positionswert wird intern als Offset für die Berechnung der

aktuellen Achsenposition verwendet.

Fehlersuche: Unzulässige Achsenkennung



Hinweise: Die Achsenposition, 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:

loigenden i anen adi nun zuruckgesetzt.

 Beim Einschalten und Neustart des C-877: für alle Achsen

■ Bei der Referenzierung: für die betroffene Achse

Siehe DFH für ein Beispiel.

# **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 ID der Datenquelle, zum Beispiel einer Achse oder eines Kanals des Controllers. Die erforderliche Quelle ist von der ausgewählten Aufzeichnungsoption

abhängig.

<RecOption> bezeichnet die aufzuzeichnende Datenart

(Aufzeichnungsoption).

Für Details siehe die nachfolgende Liste der verfügbaren

Aufzeichnungsoptionen und der entsprechenden

Datenquellen

Antwort: Keine

Hinweise: Der C-877 hat 4 Datenrekordertabellen mit 1024 Punkten

pro Tabelle.

Mit HDR? (S. 107) erhalten Sie eine Liste aller verfügbaren Aufzeichnungs- und Triggeroptionen sowie zusätzliche Informationen über die Datenaufzeichnung. Die Anzahl verfügbarer Datenrekordertabellen kann mit TNR? (S. 137)

gelesen werden.

Nähere Informationen siehe "Datenrekorder" (S. 65).



Verfügbare Aufzeichnungsoptionen mit den entsprechenden Datenquellen: 0=Nothing is recorded

Datenquelle ist die Achse:

- 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
- 90=Active parameter set
- 91=Actual frequency

# **DRC?** (Get Data Recorder Configuration)

Beschreibung: Fragt die Einstellungen für die aufzuzeichnenden Daten 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).

Mit HDR? (S. 107) können die verfügbaren Aufzeichnungsoptionen abgefragt werden.

## **DRL? (Get Number of Recorded Points)**

Beschreibung: Fragt die Anzahl der in der letzten Aufzeichnung

enthaltenen Punkte ab.



Format: DRL? [{<RecTableID>}]

Argumente: <RecTableID> ist eine Datenrekordertabelle des Controllers

Antwort: {<RecTableID>"="<uint> LF}

wobei

<uint> die Anzahl der in der letzten Aufzeichnung

enthaltenen Punkte ist.

## **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. 107) erhalten Sie eine Liste aller verfügbaren Aufzeichnungs- und Triggeroptionen sowie zusätzliche

Informationen über die Datenaufzeichnung.

Weitere Informationen siehe die Beschreibung des Befehls

DRC (S. 96) sowie "Datenrekorder" (S. 65).

Beispiel: rtr?

10

drr? 1 20

# REM C-877



```
# VERSION = 1
# TYPE = 1
# SEPARATOR = 32
# DIM = 2
# SAMPLE TIME = 0.0001000
# NDATA = 20
# NAMEO = 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, deren Aufzeichnungsption von Null verschieden ist.

Unabhängig von der eingestellten Triggeroption wird die Datenaufzeichnung immer ausgelöst, wenn eine Sprungantwortmessung mit STE (S. 134) durchgeführt wird.

Mit HDR? (S. 107) erhalten Sie eine Liste aller verfügbaren Aufzeichnungs- und Triggeroptionen sowie zusätzliche Informationen über die Datenaufzeichnung.

Weitere Informationen siehe die Beschreibung des Befehls DRC (S. 96) sowie "Datenrekorder" (S. 65).

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. 118), MOV (S. 117); <Value> muss ein

Dummy sein.

2 = next command

setzt Trigger nach Ausführung zurück; <Value> muss ein Dummy sein.

6 = any command changing target position, reset trigger after execution

z. B. MVR, MOV; setzt Trigger nach Ausführung zurück; <Value> muss ein Dummy sein.

7 = SMO command, reset trigger after execution setzt Trigger nach Ausführung 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.

Weitere Informationen finden Sie in der Beschreibung des

Befehls DRT (S. 99).

Hinweise: Da alle Datenrekordertabellen des C-877 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. 144) zu finden.

Format: ERR?

Argumente: Keine

Antwort: Der Fehlercode des zuletzt aufgetretenen Fehlers (Integer).

Fehlersuche: Kommunikationsstörung

Hinweise: Bei gleichzeitigem Zugriff mehrerer Instanzen auf den

Controller bekommt nur die erste Instanz, die den Befehl ERR? sendet, den Fehlercode geliefert. Da der Fehlercode durch die Abfrage auf 0 zurückgesetzt wird, ist der Fehler für jede weitere abfragende Instanz nicht sichtbar.



Wenn möglich, greifen Sie immer nur mit einer Instanz auf den Controller zu.

Wenn der Controller bei fehlerhaftem Systemverhalten keinen Fehlercode sendet, prüfen Sie, ob der Fehlercode durch ein Makro oder Skript oder durch PC-Software (z. B. PIMikroMove®) regelmäßig im Hintergrund abgefragt wird.

Wenn die Ursache eines Fehlers weiterhin besteht, wird der entsprechende Fehlercode sofort nach der Abfrage mit ERR? wieder neu gesetzt.

# FED (Find Edge)

Beschreibung: Bewegt die angegebene Achse zu einer angegebenen

Signalflanke.

FED setzt keinen bestimmten Positionswert an der ausgewählten Flanke (im Gegensatz zu den Befehlen FNL (S. 103), FPL (S. 104) und FRF (S. 105) zur Referenzierung), 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); SVO? (S.

136) antwortet mit dem Wert 0.

Hinweise: Der Servomodus muss mit SVO (S. 135) für die

kommandierte Achse vor dem Einsatz dieses Befehls

eingeschaltet sein (geregelter Betrieb).

Anhand von Parametern ermittelt die Firmware des C-877

Folgendes:



- Ist ein Referenzschalter vorhanden (Parameter 0x14)?
- Sind Endschalter vorhanden (Parameter 0x32)?
- Wenn der Referenzschalter durch einen Indexpuls verkörpert wird: Wie soll die Fahrt zum Indexpuls erfolgen (Parameter 0x70, 0x78, 0x79)?

Gemäß den Werten dieser Parameter aktiviert oder deaktiviert der C-877 FED-Bewegungen zu den entsprechenden Signalflanken. Passen Sie die Parameterwerte mit SPA (S. 129) oder SEP (S. 125) Ihrer Hardware entsprechend an. Weitere Informationen finden Sie in "Anpassen von Einstellungen" (S. 167).

FED kann verwendet werden, um den physikalischen Stellweg einer neuen Mechanik zu messen und so die Werte für die entsprechenden Parameter zu ermitteln:

- Abstand vom negativen zum positiven Endschalter
- Abstand zwischen dem negativen Endschalter und dem Referenzschalter (Parameter ID 0x17)
- Abstand zwischen Referenzschalter und positivem Endschalter (Parameter ID 0x2F).

Weitere Informationen siehe "Stellweg und Verfahrbereichsgrenzen" (S. 31).

Die Bewegung kann durch #24 (S. 87), STP (S. 134) und HLT (S. 108) gestoppt werden.

Vorhandene Parameter:

Die folgenden Flankenarten und deren Flankenarten und 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>}]



<AxisID> ist eine Achse des Controllers; wenn die Angabe Argumente:

weggelassen wird, sind alle Achsen betroffen.

Antwort: Keine

Fehlersuche: Unzulässige Achsenkennung

Der Servomodus muss mit SVO (S. 135) für die Hinweise:

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.

Die negative physikalische Grenze des Stellwegs wird durch den negativen Endschalter des Positionierers verkörpert. Die Differenz der Werte der Parameter 0x16 und 0x17 wird als aktuelle Position gesetzt, wenn die Achse am negativen

Endschalter ist.

Die Bewegung kann durch #24 (S. 87), STP (S. 134) und HLT

(S. 108) gestoppt werden.

Verwenden Sie FRF? (S. 106), um zu prüfen, ob die

Referenzfahrt erfolgreich war.

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.

Weitere Informationen siehe "Referenzwertbestimmung" (S. 34) und "Stellweg und Verfahrbereichsgrenzen" (S. 31).

#### **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. 135) 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.

Die positive physikalische Grenze des Stellwegs wird durch den positiven Endschalter des Positionierers verkörpert. Die Summe der Werte der Parameter 0x16 und 0x2F wird als aktuelle Position gesetzt, wenn die Achse am positiven

Endschalter ist.

Die Bewegung kann durch #24 (S. 87), STP (S. 134) und HLT

(S. 108) gestoppt werden.

Verwenden Sie FRF? (S. 106), um zu prüfen, ob die

Referenzfahrt erfolgreich war.

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.

Weitere Informationen siehe "Referenzwertbestimmung" (S. 34) und "Stellweg und Verfahrbereichsgrenzen" (S. 31).

### FRF (Fast Reference Move To Reference Switch)

Beschreibung: Startet eine Referenzierungsfahrt.

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 gleichzeitig

gestartet.

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. 135) 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.

Die Bewegung kann durch #24 (S. 87), STP (S. 134) und HLT

(S. 108) gestoppt werden.

Verwenden Sie FRF? (S. 106), um zu prüfen, ob die

Referenzfahrt erfolgreich war.

Verwenden Sie FNL (S. 103) oder FPL (S. 104) anstelle von FRF, um eine Referenzfahrt für eine Achse auszuführen, die

keinen Referenzschalter hat.

Um die beste Wiederholgenauigkeit zu erreichen, führen Sie die Referenzwertbestimmung immer auf die gleiche

Weise durch.

Weitere Informationen siehe "Referenzwertbestimmung" (S. 34) und "Stellweg und Verfahrbereichsgrenzen" (S. 31).

### 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. 103), FPL (S. 104) bzw. FRF (S. 105) oder wenn die Position direkt mit POS (S. 120) eingestellt wurde (abhängig von der mit RON (S. 121)

ausgewählten Referenzierungsmethode).

# **GOH (Go To Home Position)**

Beschreibung: Bewegt die angegebene Achse zur Nullposition.

GOH [{<AxisID>}] ist identisch mit MOV {<AxisID> 0}

Die Bewegung kann durch #24 (S. 87), STP (S. 134) und HLT

(S. 108) 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

Hinweise: Der Servomodus muss für die kommandierte Achse vor

dem Einsatz dieses Befehls eingeschaltet sein (geregelter

Betrieb).

## **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

### **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. 87) und STP (S. 134) 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

Systemabbremsung im Hinblick auf die Systemträgheit. Gilt

nicht für Trajektorien.



HLT stoppt jede Bewegung, die durch Bewegungsbefehle (z. B. MOV (S. 117), MVR (S. 118), GOH (S. 107), STE (S. 134), SMO (S. 127)), den Befehl zur Referenzwertbestimmung (FRF (S. 105)) und Makros (MAC (S. 111)) verursacht wird.

Nachdem die Achse gestoppt wurde, wird ihre Zielposition auf ihre aktuelle Position 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. 176).

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<FunctionG
roupDescription>TAB<ParameterDescription>[{TAB<Possib
leValue>"="<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-877 ist ein "Element" eine Achse oder das gesamte System.

<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. 129) beeinflusst die Parametereinstellungen im flüchtigen Speicher (RAM).

WPA (S. 143) kopiert Parametereinstellungen vom flüchtigen in den permanenten Speicher.

SEP (S. 125) schreibt die Parametereinstellungen direkt in den permanenten Speicher (ohne die Einstellungen im flüchtigen Speicher zu ändern).

RPA (S. 123) setzt den flüchtigen Speicher auf die Werte aus dem permanenten Speicher zurück.

# 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 Makroausführungszeiger zurück zur vorherigen Zeile springt, 0 bedeutet, dass der Befehl erneut ausgeführt wird, was dem Verhalten von WAC (S. 142) 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

### 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-877 das Vorhandensein oder Fehlen von

Endschaltern. Entsprechend dem Wert dieses Parameters aktiviert oder deaktiviert der C-877 das Stoppen der Bewegung an den Endschaltern und Referenzfahrten unter

Verwendung von FNL (S. 103) oder FPL (S. 104).

Passen Sie den Parameterwert Ihrer Hardware entsprechend mit SPA (S. 129) oder SEP (S. 125) an.

Weitere Informationen finden Sie in "Endschaltererkennung" (S. 30).

## **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 FREE?

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 FREE?

Fragt nach dem freien Speicherplatz für die Makroaufzeichnung

Antwort: <uint> ist die Zeichenanzahl in Bytes, für die noch freier Speicher zur Verfügung steht

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 fehlt, wenn das Makro lokale Variablen enthält, aber <String1> und <String2> im Befehl MAC NSTART weggelassen werden. Nähere Angaben siehe "Variablen" (S. 80).

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. 69) und "Empfänger- und Senderadresse" (S. 79).

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. 80).

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-877 zur Verfügung gestellten Befehle können nur in Makros verwendet werden: DEL (S. 93), JRC (S. 110), MEX (S. 116) und WAC (S. 142).

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. 87) und STP (S. 134) 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. 86) 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

### **MAN? (Get Help String For Command)**

Beschreibung: Zeigt einen detaillierten Hilfetext zu einzelnen Befehlen an.

Format: MAN? < CMD>

Argumente: <CMD> ist das Befehlskürzel des Befehls, für den der

Hilfetext angezeigt werden soll (siehe unten).

Antwort: Ein String, der den Befehl beschreibt.

### MAT (Calculate And Save To Variable)

Beschreibung: Führt eine mathematische Operation oder Bitoperation aus

und speichert das Ergebnis als Variable (S. 80).

Die Variable ist nur im flüchtigen Speicher (RAM)

vorhanden.

Format: MAT <Variable> "=" <FLOAT1> <OP> <FLOAT2>

Argumente: <Variable> ist der Name der Variable, in der das Ergebnis

gespeichert werden soll.

<FLOAT1> und <FLOAT2> sind die Größen, aus denen das Ergebnis berechnet werden soll. Sie können direkt angegeben werden oder über den Wert einer Variablen.

<OP> ist der zu verwendende Operator: Folgende Operatoren sind möglich:

<op></op>	Operation	Тур
+	Addition	Mathematische Operation
-	Subtraktion	Mathematische Operation
*	Multiplikation	Mathematische Operation
AND	UND	Bitoperation
OR	ODER	Bitoperation
XOR	XOR	Bitoperation

Wichtig: Vor und nach dem "=" und dem Operator muss jeweils ein Leerzeichen stehen!



Antwort: Keine

Hinweise: Die Verwendung von MAT zum Setzen lokaler Variablen ist

nur in Makros möglich.

Beispiel 1: Senden: MAT TARGET = \${POS} \* 2.0

Die Variable TARGET erhält den 2,0-fachen Wert der

Variable POS.

Beispiel 2: Senden: MAT TARGET = 2 \* 0x10

Senden: VAR? TARGET Empfangen: TARGET=32

Hinweis: Die Größen, aus denen das Ergebnis berechnet werden soll, können im Hexadezimal- oder Dezimalformat

geschrieben werden. Das Ergebnis wird immer im

Dezimalformat ausgegeben.

Beispiel 3: Senden: MAT INVERT = 0x45 XOR 0xFF

Senden: VAR? INVERT Empfangen: INVERT=186

Hinweis: Die Bitoperation XOR mit dem Wert 0xFF entspricht einer Inversion des Werts 0x45. Das Ergebnis

wird im Dezimalformat ausgegeben.

### **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. 142).

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

### **MOV (Set Target Position)**

Beschreibung: Setzt eine absolute Zielposition für die angegebene Achse.

Format: MOV {<AxisID> <Position>}

Argumente: <AxisID> ist eine Achse des Controllers.

<Position> ist die absolute Zielposition in physikalischen

Einheiten.

Antwort: Keine

Hinweise: Der Servomodus muss bei Verwendung dieses Befehls

eingeschaltet sein (geregelter Betrieb).

Die Zielposition muss sich innerhalb der

Verfahrbereichsgrenzen befinden. Verwenden Sie TMN? (S.

136) und TMX? (S. 137), um die aktuell gültigen

Verfahrbereichsgrenzen abzufragen.

Die Bewegung kann durch #24 (S. 87), STP (S. 134) und HLT

(S. 108) 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.

Beispiel 1: Senden: MOV 1 10

Hinweis: Achse 1 bewegt sich nach 10 (Zielposition in

mm)

Beispiel 2: Senden: MOV 1 243

Senden: ERR? 7

Hinweis: Die Achse bewegt sich nicht. Der

Fehlercode "7" in der Antwort auf den Befehl ERR? (S. 101) 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

### **MVR (Set Target Relative To Current Position)**

Beschreibung: Bewegt die angegebene Achse relativ zur letzten

kommandierten Zielposition.

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: Der Servomodus muss bei Verwendung dieses Befehls

eingeschaltet sein (geregelter Betrieb).

Die Zielposition muss sich innerhalb der

Verfahrbereichsgrenzen befinden. Verwenden Sie TMN? (S.

136) und TMX? (S. 137), um die aktuell gültigen

Verfahrbereichsgrenzen abzufragen, und MOV? (S. 118) für

die Abfrage des aktuellen Ziels.

Die Bewegung kann durch #24 (S. 87), STP (S. 134) und HLT

(S. 108) 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.

Beispiel: Senden: MOV 1 0.5

Hinweis: Dies ist eine absolute Bewegung.

Senden: POS? 1

Empfangen: 1=0.500000

Senden: MOV? 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: MOV? 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>}]

<AxisID> ist eine Achse des Controllers. Argumente:

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

wobei

<uint> = "1" wenn die angegebene Achse den Zielwert

erreicht hat, 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 0x406 und 0x407) und die Verzögerungszeit (Parameter 0x3F) beeinflusst. Details

siehe "On-Target-Status" (S. 27).

## **POS (Set Real Position)**

Beschreibung: Setzt die aktuelle Position der Achse (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 die Referenzierungsmethode "0" ausgewählt ist;

siehe RON (S. 121).

Eine Achse wird als "referenziert" bezeichnet, wenn die Position mit POS gesetzt wurde (weitere Informationen

siehe "Referenzierung" (S. 34)).

Die kleinsten und größten kommandierbaren Positionen (TMN? (S. 136), TMX? (S. 137)) werden nicht angepasst, wenn eine Position mit POS gesetzt wurde. Dies kann zu Zielpositionen führen, die vom C-877 zugelassen sind, aber von der Hardware nicht angefahren werden können. Ebenso sind Zielpositionen möglich, die von der Hardware angefahren werden können, aber vom C-877 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 ist und aktuell ausgeführt wird. Die Antwort ist eine leere Zeile, wenn kein Makro ausgeführt

wird.

## **RON (Set Reference Mode)**

Beschreibung: Wählt die Referenzierungsmethode für die angegebenen

Achsen.



Format: RON {<AxisID> <ReferenceOn>}

Argumente: <AxisID> ist eine Achse des Controllers.

<ReferenceOn> ist die Referenzierungsmethode. Kann 0

oder 1 sein. 1 ist Standard. Details siehe unten.

Antwort: Keine

Fehlersuche: Unzulässige Achsenkennung

Hinweise: <ReferenceOn> = 0: Der Achse kann ein absoluter

Positionswert mit POS (S. 120) zugewiesen werden, oder eine Referenzfahrt kann mit FRF (S. 105), FNL (S. 103) oder FPL (S. 104) gestartet werden. Relative Bewegungen mit MVR sind möglich, auch wenn die Achse noch nicht

referenziert wurde.

<ReferenceOn> = 1: Für die 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 die Achse referenziert

wurde.

Weitere Informationen siehe "Referenzierung" (S. 34) und

"Stellweg und Verfahrbereichsgrenzen" (S. 31).

#### **RON? (Get Reference Mode)**

Beschreibung: Fragt die Referenzierungsmethode der angegebenen

Achsen ab.

Format: RON? [{<AxisID>}]

Argumente: <AxisID> ist eine Achse des Controllers.

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

wobei

<ReferenceOn> die aktuell für die Achse ausgewählte

Referenzierungsmethode ist

Fehlersuche: Unzulässige Achsenkennung

Hinweis: Weitere Informationen finden Sie in der Beschreibung des

Befehls RON (S. 121).



## **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. 109) erhalten Sie eine Liste der verfügbaren

Parameter. SPA (S. 129) beeinflusst die

Parametereinstellungen im flüchtigen Speicher, WPA (S. 143) schreibt Parametereinstellungen aus dem flüchtigen in den permanenten Speicher und SEP (S. 125) 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 Hexadezimaloder Dezimalformat geschrieben werden. Nähere Angaben

siehe unten.

Antwort: Keine

Fehlersuche: Unzulässige Elementkennung, falsche Parameter-ID

Hinweise: Die Informationen aus dem ID-Chip des Positionierers und

aus Positioniererdatenbanken werden nur in den

flüchtigen Speicher des C-877 geladen. Durch RPA werden die geladenen Daten überschrieben. Wenden Sie RPA nur

an, wenn Sie sicher sind, dass der C-877 mit den

Parameterwerten aus dem permanenten Speicher korrekt

funktioniert.

Mit dem C-877 können Sie entweder alle Parameter oder gezielt einen einzelnen Parameter mit RPA zurücksetzen.

Verfügbare Element-IDs und Parameter-IDs: Ein Element ist eine Achse, die Kennung kann mit SAI (S. 125) geändert werden. Weitere Informationen siehe

"Kommandierbare Elemente" (S. 13).

Gültige Parameter-IDs finden Sie in der Parameterübersicht

(S. 176).



## 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:

Aufz.Dauer = Zykluszeit des Regelkreises \* RTR Wert

\*Anzahl der Punkte

wobei

die Zykluszeit des Regelkreises für den C-877 100 μs

beträgt

die Anzahl der Punkte für den C-877 1024 beträgt (Länge

der Datenrekordertabelle)

Weitere Informationen siehe "Datenrekorder" (S. 65).

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. 138), 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. 13).

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.

Hinweise: Wenn der Parameter **Stage Name** (0x3C) den Wert

NOSTAGE hat, ist die Achse "deaktiviert". Eine deaktivierte Achse ist nicht für achsenbezogene Befehle zugänglich (z.B. Bewegungsbefehle oder Positionsabfragen) und nur in der

Antwort auf SAI? ALL enthalten.



### 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. 123) 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. 109) gibt eine Liste verfügbarer Parameter zurück.

SPA (S. 129) schreibt Parametereinstellungen in den flüchtigen Speicher (ohne die Einstellungen im

permanenten Speicher zu ändern).

WPA (S. 143) schreibt Parametereinstellungen vom

flüchtigen in den permanenten Speicher.

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 Hexadezimaloder 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

Beachten Sie, dass die Anzahl von Schreibzyklen im Hinweise:

> permanenten Speicher begrenzt ist. Schreiben Sie Standardeinstellungen nur, wenn dies notwendig ist.

Beim C-877 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. 125) geändert werden) oder das gesamte System. Weitere Informationen siehe "Kommandierbare Elemente" (S. 13).

Gültige Parameter-IDs finden Sie in der Parameterübersicht

(S. 176).

## **SEP? (Get Non-Volatile Memory Parameters)**

Beschreibung: Fragt den Wert eines Parameters für ein angegebenes

Element aus dem permanenten Speicher ab.

Mit HPA? (S. 109) 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 Hexadezimaloder 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-877 können Sie entweder alle Parameter oder

gezielt einzelne Parameter pro Befehl SEP? abfragen.

Ein Element ist eine Achse (die Kennung kann mit SAI (S.

Verfügbare Element-IDs und Parameter-IDs:

125) geändert werden) oder das gesamte System. Weitere Informationen siehe "Kommandierbare Elemente" (S. 13).

Gültige Parameter-IDs finden Sie in der Parameterübersicht

(S. 176).



# 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

Der Servomodus ist für eine der angegebenen Achsen

eingeschaltet.

Hinweise: HINWEIS: Bei großen Stellwerten kann der Positionierer

trotz Endschalterfunktion 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 die

Piezospannung für die Achse an. Das Vorzeichen des Wertes bestimmt die Bewegungsrichtung: -32766 entspricht der maximalen Amplitude der Piezospannung in negativer Bewegungsrichtung, und 32766 entspricht der maximalen Amplitude der Piezospannung in positiver Bewegungsrichtung. Weitere Informationen siehe

"Unterstützte Motortypen" (S. 28).

Wenn ein hoher Stellwert über längere Zeit gesetzt bleibt, kann sich der angeschlossene Positionierer erwärmen. Überhitzen kann zu Schäden am Positionierer führen. Durch den Parameter *PID Maximum Output Time* (s) (ID 0x7B) wird die maximale Zeitdauer vorgegeben, für die ein hoher Stellwert im geregelten Betrieb gesetzt sein darf. Ein hoher Stellwert liegt vor, wenn gilt:

Aktueller Betrag des Stellwerts ≥ 95 % von *Maximum Motor Output* (ID 0x9). Weitere Informationen siehe "Schutz vor Überhitzung" (S. 63).



Die Parameter Range Limit Min (0x07000000) und Range

Limit Max (0x07000001) können als

Verfahrbereichsgrenzen für Bewegungen im ungeregelten 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: SMO 1 -16000

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 im ungeregelten Betrieb gesetzt wird.

Weitere Informationen siehe SMO (S. 127) und das

Blockdiagramm (S. 13).

### **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 Hexadezimaloder 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. 123) auf ihre Standardwerte

zurückgesetzt 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. 109) gibt eine Liste verfügbarer Parameter zurück.

SEP (S. 125) schreibt Parametereinstellungen direkt in den

permanenten Speicher (ohne die Einstellungen im

flüchtigen Speicher zu ändern).

WPA (S. 143) 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-877 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. 125) geändert werden) oder das gesamte System. Weitere Informationen siehe "Kommandierbare Elemente" (S. 13).

Gültige Parameter-IDs sind in der Parameterübersicht (S.

176) angegeben.



Beispiel 1: Senden: SPA 1 0x411 100

Hinweis: Setzt für Parametergruppe 1 den P-Term des Regelalgorithmus für Achse 1 auf 100, die Parameter-ID

wird im Hexadezimalformat geschrieben

Senden: SPA 1 1041 150

Hinweis: Setzt für Parametergruppe 1 den P-Term des Regelalgorithmus für Achse 1 auf 150, die Parameter-ID

wird im Dezimalformat geschrieben

Beispiel 2: Für Parametergruppe 2 müssen P-, I- und D-Parameter des

Regelalgorithmus an eine neue Last, die auf die

angeschlossene Mechanik einwirkt, angepasst werden.

**Senden:** SPA 1 0x421 150

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: WPA 100

Hinweis: Siehe die Befehlsbeschreibung für WPA (S. 143) für Details zum Umfang der gespeicherten Einstellungen.

Beispiel 3: Senden: SEP 100 LEFT 0xA 20

Hinweis: Die maximale 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. 109) 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 Hexadezimaloder 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-877 können Sie entweder alle Parameter oder

gezielt einzelne Parameter je Befehl SPA? abfragen.

Verfügbare Element -IDs und

Parameter-IDs:

Ein Element ist eine Achse (die Kennung kann mit SAI (S. 125) geändert werden) oder das gesamte System. Weitere Informationen siehe "Kommandierbare Elemente" (S. 13).

Gültige Parameter-IDs sind in der Parameterübersicht (S.

176) angegeben.

## SRG? (Query Status Register Value)

Beschreibung: Gibt Registerwerte für die abgefragten Elemente und

Register zurück.

Format: SRG? [{<|tem|D> <|Register|D>}]



Argumente: < ItemID> ist das Element, für das ein Register abgefragt

werden soll. Nähere Angaben siehe unten.

<RegisterID> ist die ID des angegebenen Registers,

verfügbare Register siehe unten.

Antwort: {<ItemID><RegisterID>"="<Value> LF}

wobei

<Value> der Wert des Registers ist, nähere Angaben siehe

unten.

Hinweis: Dieser Befehl ist funktionsgleich mit #4 (S. 85), der

bevorzugt werden sollte, wenn der Controller

zeitaufwändige Aufgaben ausführt.

Mögliche

<ItemID> ist eine Achse des Controllers.

Register-IDs und

Antwortwerte: <RegisterID> kann 1 sein.

<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
Beschrei- bung	On- Target- Status	Bestimmt den Referenz- wert	In Bewe- gung	Servo- modus Ein	1	1	1	Fehler- flag

Bit	7	6	5	4	3	2	1	0
Beschrei- bung	-	-	-	-	Achse ist referenzie rt	Pos. End- schal- ter	Refe- renz- schal- ter	Neg. End- schal- ter

Beispiel: Senden: #4

Empfangen: 0x900A

= Binärformat: **1**00**1**000**0**000010**1**0

Hinweis: Die Antwort wird im Hexadezimalformat

angegeben. Sie besagt (fett angezeigt):

Die Achse ist an der Zielposition (On-Target-Status; Bit 15),

der Servomodus ist eingeschaltet (Bit 12), es ist kein Fehler aufgetreten (Bit 8),

die Achse wurde bereits referenziert (Bit 3),

der Positionierer befindet sich an der Position des

Referenzschalters (Bit 1).



## 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. 96) gesetzt werden.

Die aufgezeichneten Daten können mit dem Befehl DRR?

(S. 98) 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. 136) und TMX? (S. 137), um die aktuell gültigen

Verfahrbereichsgrenzen abzufragen und MOV? (S. 118) für

die Abfrage des aktuellen Ziels.

Hinweise: Ein "Sprung" besteht aus einer 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.

87).

Format: STP

Argumente: Keine

Antwort: Keine

Fehlersuche: Kommunikationsstörung



Hinweise: STP stoppt jede Bewegung, die durch Bewegungsbefehle

(z. B. MOV (S. 117), MVR (S. 118), GOH (S. 107), STE (S. 134), SMO (S. 127)), Trajektorien-Ausführung (TGS), den Befehl zur Referenzwertbestimmung (FRF (S. 105)) und Makros (MAC (S. 111)) verursacht wird. Stoppt auch die

Makroausführung.

Nachdem die Achse gestoppt wurde, wird ihre Zielposition

auf ihre aktuelle Position gesetzt.

HLT (S. 108) stoppt im Gegensatz zu STP die Bewegung mit

vorgegebener Abbremsung im Hinblick auf die Systemträgheit. Gilt nicht für Trajektorien.

#### **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 ungeregelten 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 MOV (S. 117), MVR (S.

118) oder GOH (S. 107).

Servomodus aus: Verwenden Sie SMO (S. 127).

Der Servomodus muss eingeschaltet sein, bevor

Referenzfahrten mit FRF (S. 105), FNL (S. 103) oder FPL (S.

104) gestartet werden können.

Wenn der Servomodus ausgeschaltet wird, während sich

die Achse bewegt, stoppt die Achse.

Beim Auftreten eines Bewegungsfehlers (S. 63) wird der

Servomodus ausgeschaltet.



## **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

## TCV? (Get Commanded Closed-Loop Velocity)

Beschreibung: Fragt den aktuellen Wert der Geschwindigkeit ab (vom

Profilgenerator berechneter Wert).

Format: TCV? [{<AxisID>}]

Argumente: <AxisID> ist eine Achse des Controllers.

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

wobei

<float> der Geschwindigkeitswert in physikalischen

Einheiten pro Sekunde ist.

### 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. Beim Neudefinieren der Nullposition mit dem Befehl DFH (S. 94) wird die kleinste kommandierbare Position automatisch an die neue

Nullposition angepasst.

### **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. Beim Neudefinieren der Nullposition mit dem Befehl DFH (S. 94) wird die größte kommandierbare Position automatisch an die neue

Nullposition angepasst.

# **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-877 hat vier Datenrekordertabellen mit 1024

Datenpunkten pro Tabelle.

Weitere Informationen siehe "Datenrekorder" (S. 65).



# **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-877 das Vorhandensein oder Fehlen eines

Referenzschalters. Gemäß dem Wert dieses Parameters aktiviert oder deaktiviert der C-877 Referenzfahrten zum

Referenzschalter (Befehl FRF (S. 105)).

Passen Sie den Parameterwert entsprechend Ihrer Hardware mit SPA (S. 129) oder SEP (S. 125) an.

Weitere Informationen siehe "Referenzschalterkennung"

(S. 29).

## TVI? (Tell Valid Character Set For Axis Identifiers)

Beschreibung: Gibt einen String mit den für Achsenkennungen

verwendbaren Zeichen zurück.

Verwenden Sie SAI (S. 125), um die Achsenkennungen zu

ändern und SAI? (S. 125), um die aktuell gültigen

Achsenkennungen abzufragen.

Format: TVI?

Argumente: Keine

Antwort: <string> ist eine Liste von Zeichen.

Hinweise: Beim C-877 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. 80).

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. 80).

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. 89).

# **VAR? (Get Variable Values)**

Beschreibung: Gibt Variablenwerte zurück.

Wird VAR? mit CPY (S. 91), JRC (S. 110), MEX (S. 116) oder WAC (S. 142) kombiniert, muss die Antwort auf VAR? ein

einzelner Wert sein (und nicht mehr).

Nähere Angaben zu lokalen und globalen Variablen siehe

"Variablen" (S. 80).

Format: VAR? [{<Variable>}]

Argumente: <Variable> ist der Name der abzufragenden Variablen.

Nähere Angaben zu Namenskonventionen siehe

"Variablen" (S. 80).

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. Weitere Informationen zu lokalen und globalen Variablen siehe

"Variablen" (S. 80).

Beispiel: Beispiel siehe ADD (S. 89).

#### VEL (Set Closed-Loop Velocity)

Beschreibung: Setzt die Geschwindigkeit für die angegebenen Achsen.

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.

Die Geschwindigkeit kann mit VEL verändert werden, während die Achse sich bewegt.

VEL ändert den Wert des Parameters *Closed-Loop Velocity* (*Phys. Unit/s*) (ID 0x49) im flüchtigen Speicher des C-877. Der Parameterwert kann mit WPA (S. 143) als Standard gespeichert werden, Details siehe "Anpassen von Einstellungen" (S. 167).

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.

# **VEL? (Get Closed-Loop Velocity)**

Beschreibung: Fragt die kommandierte Geschwindigkeit ab.

Werden keine Argumente angegeben, wird der Wert aller

Achsen abgefragt.

Format: VEL? [{<AxisID>}]

Argumente: <AxisID> ist eine Achse des Controllers.

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

wobei

<float> der aktuell gültige kommandierte

Geschwindigkeitswert in physikalischen Einheiten pro

Sekunde ist.

Hinweise: VEL? fragt den mit VEL kommandierten Wert der

Geschwindigkeit für den geregelten Betrieb ab (Wert des

Parameters Closed Loop Velocity (Phys. Unit/s) (ID 0x49)

im flüchtigen Speicher).

#### **VER? (Get Versions Of Firmware And Drivers)**

Beschreibung: Fragt die Versionen der Firmware des C-877 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. 116).

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:

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

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. 123) wiederhergestellt werden.

Mit HPA? (S. 109) erhalten Sie eine Liste aller verfügbaren Parameter.

Prüfen Sie die aktuellen Parametereinstellungen im flüchtigen Speicher mit SPA? (S. 129).

Ein Beispiel finden Sie in der Beschreibung des Befehls SPA

(S. 129).

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 Hexadezimaloder Dezimalformat geschrieben werden. Nähere Angaben

siehe unten.

Antwort: Keine

Fehlersuche: Unzulässige Elementkennung, falsche Parameter-ID,

ungültiges Kennwort

Beachten Sie, dass die Anzahl von Schreibzyklen im permanenten Speicher begrenzt ist. Schreiben Sie Standardeinstellungen nur, wenn dies notwendig ist.



Hinweise: Parameterwerte im flüchtigen Speicher können mit SPA (S.

129) geändert werden.

Gültige Das Kennwort zum Schreiben in den permanenten

Kennwörter: Speicher ist "100".

Element- und Die gezielte Auswahl einzelner Elemente und Parameter Für die Speicherung ist beim C-877 nicht möglich, d. h.

<ItemID> und <PamID> werden ignoriert.

# 8.7 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_STA GE	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_RAN GE	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_FAILE D	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_CONTRO L	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_DAT A	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		•
01	PI_CNTR_MACRO_UNKNOWN_COMMAN D	Not macro command
82		_
	D	Not macro command
82	D PI_CNTR_MACRO_PC_ERROR	Not macro command  Macro counter out of range
82 83	D PI_CNTR_MACRO_PC_ERROR PI_CNTR_JOYSTICK_ACTIVE	Not macro command  Macro counter out of range Joystick is active
82 83 84	D PI_CNTR_MACRO_PC_ERROR PI_CNTR_JOYSTICK_ACTIVE PI_CNTR_MOTOR_IS_OFF	Not macro command  Macro counter out of range Joystick is active  Motor is off
82 83 84 85	D PI_CNTR_MACRO_PC_ERROR PI_CNTR_JOYSTICK_ACTIVE PI_CNTR_MOTOR_IS_OFF PI_CNTR_ONLY_IN_MACRO	Not macro command  Macro counter out of range Joystick is active Motor is off Macro-only command
82 83 84 85 86	D PI_CNTR_MACRO_PC_ERROR PI_CNTR_JOYSTICK_ACTIVE PI_CNTR_MOTOR_IS_OFF PI_CNTR_ONLY_IN_MACRO PI_CNTR_JOYSTICK_UNKNOWN_AXIS	Not macro command  Macro counter out of range Joystick is active  Motor is off  Macro-only command Invalid joystick axis
82 83 84 85 86 87	D PI_CNTR_MACRO_PC_ERROR PI_CNTR_JOYSTICK_ACTIVE PI_CNTR_MOTOR_IS_OFF PI_CNTR_ONLY_IN_MACRO PI_CNTR_JOYSTICK_UNKNOWN_AXIS PI_CNTR_JOYSTICK_UNKNOWN_ID	Not macro command  Macro counter out of range Joystick is active Motor is off Macro-only command Invalid joystick axis Joystick unknown Move without referenced stage
82 83 84 85 86 87	PI_CNTR_MACRO_PC_ERROR PI_CNTR_JOYSTICK_ACTIVE PI_CNTR_MOTOR_IS_OFF PI_CNTR_ONLY_IN_MACRO PI_CNTR_JOYSTICK_UNKNOWN_AXIS PI_CNTR_JOYSTICK_UNKNOWN_ID PI_CNTR_REF_MODE_IS_ON PI_CNTR_NOT_ALLOWED_IN_CURRENT_M	Not macro command  Macro counter out of range Joystick is active Motor is off Macro-only command Invalid joystick axis Joystick unknown Move without referenced stage Command not allowed in



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_A XIS_IN_MOTION	This command is not allowed while the affected axis or its master is in motion.
94	PI_CNTR_OPEN_LOOP_JOYSTICK_ENABLE D	Servo cannot be switched on when open-loop joystick control is activated.
95	PI_CNTR_INVALID_SERVO_STATE_FOR_PA RAMETER	This parameter cannot be changed in current servo mode.
96	PI_CNTR_UNKNOWN_STAGE_NAME	Unknown stage name
97	PI_CNTR_INVALID_VALUE_LENGTH	Invalid length of value (too much characters)
98	PI_CNTR_AUTOZERO_FAILED	AutoZero procedure was not successful
99	PI_CNTR_SENSOR_VOLTAGE_OFF	Sensor voltage is off
100	PI_LABVIEW_ERROR	PI driver for use with NI LabVIEW 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_HEAD ER	uudecode: incomplete header
207	PI_CNTR_UUDECODE_NOTHING_TO_DEC ODE	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_CHAN GE	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_ESTIMATI ON	While MOV! is running position can only be estimated!
219	PI_CNTR_POSITION_BASED_ON_INTERPOL ATION	Position was calculated during MOV motion
220	PI_CNTR_INTERPOLATION_FIFO_UNDERR UN	FIFO buffer underrun during interpolation
221	PI_CNTR_INTERPOLATION_FIFO_OVERFLO W	FIFO buffer overflow during interpolation
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_WH EN_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_LI MIT	Wave parameter out of range
406	PI_CNTR_WGO_BIT_NOT_SUPPORTED	WGO command bit not supported
500	PI_CNTR_EMERGENCY_STOP_BUTTON_AC TIVATED	The \"red knob\" is still set and disables system
501	PI_CNTR_EMERGENCY_STOP_BUTTON_W AS_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.
507	PI_CNTR_LOST_SPI_SLAVE_CONNECTION	Digital connection to external SPI slave device is lost
508	PI_CNTR_MOVE_ATTEMPT_NOT_PERMITT ED	Move attempt not permitted due to customer or limit settings
509	PI_CNTR_TRIGGER_EMERGENCY_STOP	Emergency stop caused by trigger input
530	PI_CNTR_NODE_DOES_NOT_EXIST	A command refers to a node that does not exist
531	PI_CNTR_PARENT_NODE_DOES_NOT_EXIS T	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
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



539	PI_CNTR_NODE_PARENT_IDENTICAL_TO_ CHILD	A node cannot be linked to itself
540	PI_CNTR_NODE_DEFINITION_INCONSISTE NT	Node definition is erroneous or not complete (replace or delete it)
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_S UPPORTED	With some transformations pivot point usage is not supported
545	PI_CNTR_SOFTLIMITS_INVALID	Soft limits invalid due to changes in coordinate system
546	PI_CNTR_CS_WRITE_PROTECTED	Coordinate system is write protected
547	PI_CNTR_CS_CONTENT_FROM_CONFIG_FI LE	Coordinate system cannot be changed because its content is loaded from a configuration file
548	PI_CNTR_CS_CANNOT_BE_LINKED	Coordinate system may not be linked
549	PI_CNTR_KSB_CS_ROTATION_ONLY	A KSB-type coordinate system can only be rotated by multiples of 90 degrees
551	PI_CNTR_CS_DATA_CANNOT_BE_QUERIE D	This query is not supported for this coordinate system type
552	PI_CNTR_CS_COMBINATION_DOES_NOT_ EXIST	This combination of work- and-tool coordinate systems does not exist
553	PI_CNTR_CS_COMBINATION_INVALID	The combination must consist of one work and one tool coordinate system
554	PI_CNTR_CS_TYPE_DOES_NOT_EXIST	This coordinate system type does not exist
555	PI_CNTR_UNKNOWN_ERROR	BasMac: unknown controller error
556	PI_CNTR_CS_TYPE_NOT_ACTIVATED	No coordinate system of this type is activated
557	PI_CNTR_CS_NAME_INVALID	Name of coordinate system is invalid
558	PI_CNTR_CS_GENERAL_FILE_MISSING	File with stored CS systems is missing or erroneous



559	PI_CNTR_CS_LEVELING_FILE_MISSING	File with leveling CS is missing or erroneous
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
607	PI_CNTR_NO_INTEGER	Value is not integer
608	PI_CNTR_FAST_ALIGNMENT_PROCESS_IS_ NOT_RUNNING	Fast alignment process cannot be paused because it is not running
609	PI_CNTR_FAST_ALIGNMENT_PROCESS_IS_ NOT_PAUSED	Fast alignment process cannot be restarted/resumed because it is not paused
650	PI_CNTR_UNABLE_TO_SET_PARAM_WITH _SPA	Parameter could not be set with SPA - SEP needed?
651	PI_CNTR_PHASE_FINDING_ERROR	Phase finding error
652	PI_CNTR_SENSOR_SETUP_ERROR	Sensor setup error
653	PI_CNTR_SENSOR_COMM_ERROR	Sensor communication error
654	PI_CNTR_MOTOR_AMPLIFIER_ERROR	Motor amplifier error
655	PI_CNTR_OVER_CURR_PROTEC_TRIGGERE D_BY_I2T	Overcurrent protection triggered by I2T-module
656	PI_CNTR_OVER_CURR_PROTEC_TRIGGERE D_BY_AMP_MODULE	Overcurrent protection triggered by amplifier module
657	PI_CNTR_SAFETY_STOP_TRIGGERED	Safety stop triggered
658	PI_SENSOR_OFF	Sensor off?
659	PI_CNTR_PARAM_CONFLICT	Parameter could not be set. Conflict with another parameter.
700	PI_CNTR_COMMAND_NOT_ALLOWED_IN _EXTERNAL_MODE	Command not allowed in external mode
710	PI_CNTR_EXTERNAL_MODE_ERROR	External mode communication error



715	PI_CNTR_INVALID_MODE_OF_OPERATION	Invalid mode of operation
716	PI_CNTR_FIRMWARE_STOPPED_BY_CMD	Firmware stopped by command (#27)
717	PI_CNTR_EXTERNAL_MODE_DRIVER_MISS ING	External mode driver missing
718	PI_CNTR_CONFIGURATION_FAILURE_EXTE RNAL_MODE	Missing or incorrect configuration of external mode
719	PI_CNTR_EXTERNAL_MODE_CYCLETIME_I NVALID	External mode cycletime invalid
720	PI_CNTR_BRAKE_ACTIVATED	Brake is activated
725	PI_CNTR_DRIVE_STATE_TRANSITION_ERR OR	Drive state transition error
731	PI_CNTR_SURFACEDETECTION_RUNNING	Command not allowed while surface detection is running
732	PI_CNTR_SURFACEDETECTION_FAILED	Last surface detection failed
733	PI_CNTR_FIELDBUS_IS_ACTIVE	Fieldbus is active and is blocking GCS control commands
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_ARG UMENT	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 response was received while executing WAC/MEX/JRC/
1011	PI_CMD_NOT_VALID_IN_MACRO_MODE	Command not valid during macro execution
1012	PI_CNTR_ERROR_IN_MACRO	Error occured during macro execution
1013	PI_CNTR_NO_MACRO_OR_EMPTY	No macro with given name on controller, or macro is empty
1015	PI_CNTR_INVALID_ARGUMENT	One or more arguments given to function is invalid (empty string, index out of range,)
1024	PI_CNTR_MOTION_ERROR	Motion error: position error too large, servo is switched off automatically
1025	PI_CNTR_MAX_MOTOR_OUTPUT_REACHE D	Maximum motor output reached
1028	PI_CNTR_UNKNOWN_CHANNEL_IDENTIFIER	Unknown channel identifier
1063	PI_CNTR_EXT_PROFILE_UNALLOWED_CM D	User Profile Mode: Command is not allowed, check for required preparatory commands
1064	PI_CNTR_EXT_PROFILE_EXPECTING_MOTI ON_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_RANG E	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_I DENTIFIER	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



2100	PI_CNTR_FEATURE_LICENSE_INVALID	Entered license is invalid
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
4008	PI_CNTR_FW_CRC_PAR_ERROR	FW Parameter CRC wrong
4009	PI_CNTR_FW_CRC_FW_ERROR	FW CRC wrong
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 cannot 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
5300	PI_CNTR_FREQU_ANALYSIS_FAILED	Frequency analysis failed
5301	PI_CNTR_FREQU_ANALYSIS_RUNNING	Another frequency analysis is running
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
Schnitt	stellenfehler	
_		
0	COM_NO_ERROR	No error occurred during function call
-1	COM_NO_ERROR COM_ERROR	
		function call  Error during com operation
-1	COM_ERROR	function call Error during com operation (could not be specified)
-1 -2	COM_ERROR SEND_ERROR	function call Error during com operation (could not be specified) Error while sending data
-1 -2 -3	COM_ERROR SEND_ERROR REC_ERROR	function call  Error during com operation (could not be specified)  Error while sending data  Error while receiving data  Not connected (no port with
-1 -2 -3 -4	COM_ERROR  SEND_ERROR  REC_ERROR  NOT_CONNECTED_ERROR	function call  Error during com operation (could not be specified)  Error while sending data  Error while receiving data  Not connected (no port with given ID open)
-1 -2 -3 -4	COM_ERROR  SEND_ERROR  REC_ERROR  NOT_CONNECTED_ERROR  COM_BUFFER_OVERFLOW	function call  Error during com operation (could not be specified)  Error while sending data  Error while receiving data  Not connected (no port with given ID open)  Buffer overflow
-1 -2 -3 -4 -5 -6	COM_ERROR  SEND_ERROR  REC_ERROR  NOT_CONNECTED_ERROR  COM_BUFFER_OVERFLOW  CONNECTION_FAILED	function call  Error during com operation (could not be specified)  Error while sending data  Error while receiving data  Not connected (no port with given ID open)  Buffer overflow  Error while opening port
-1 -2 -3 -4 -5 -6 -7	COM_ERROR  SEND_ERROR  REC_ERROR  NOT_CONNECTED_ERROR  COM_BUFFER_OVERFLOW  CONNECTION_FAILED  COM_TIMEOUT	function call  Error during com operation (could not be specified)  Error while sending data  Error while receiving data  Not connected (no port with given ID open)  Buffer overflow  Error while opening port  Timeout error  There are more lines waiting
-1 -2 -3 -4 -5 -6 -7 -8	COM_ERROR  SEND_ERROR  REC_ERROR  NOT_CONNECTED_ERROR  COM_BUFFER_OVERFLOW  CONNECTION_FAILED  COM_TIMEOUT  COM_MULTILINE_RESPONSE	function call  Error during com operation (could not be specified)  Error while sending data  Error while receiving data  Not connected (no port with given ID open)  Buffer overflow  Error while opening port  Timeout error  There are more lines waiting in buffer  There is no interface or DLL
-1 -2 -3 -4 -5 -6 -7 -8	COM_ERROR  SEND_ERROR  REC_ERROR  NOT_CONNECTED_ERROR  COM_BUFFER_OVERFLOW  CONNECTION_FAILED  COM_TIMEOUT  COM_MULTILINE_RESPONSE  COM_INVALID_ID	function call  Error during com operation (could not be specified)  Error while sending data  Error while receiving data  Not connected (no port with given ID open)  Buffer overflow  Error while opening port  Timeout error  There are more lines waiting in buffer  There is no interface or DLL handle with the given ID  Event/message for notification could not be

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_FO R_ERASE	FTDIUSB: device not opened for erase
-42	COM_FTDIUSB_DEVICE_NOT_OPENED_FO R_WRITE	FTDIUSB: device not opened for write
-43	COM_FTDIUSB_FAILED_TO_WRITE_DEVIC E	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_PROGRAM MED	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!)

-62



-02	COM_30CKL1_HO31_NO1_HOUND	Host flot fourid
-63	COM_DEVICE_CONNECTED	Device already connected
-64	COM_INVALID_COM_PORT	Invalid COM port
-65	COM_USB_DEVICE_NOT_FOUND	USB device not found
-66	COM_NO_USB_DRIVER	No USB driver installed
-67	COM_USB_NOT_SUPPORTED	USB is not supported
DLL-Feh	ller	
-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 SGAmust 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 errorcould not start thread
-1010	PI_IN_MACRO_MODE	Controller is (already) in macro modecommand not valid in macro mode
-1011	PI_NOT_IN_MACRO_MODE	Controller not in macro modecommand 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

COM\_SOCKET\_HOST\_NOT\_FOUND Host not found



-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 SPAparameter not defined for this controller!
-1042	PI_NUMBER_OF_POSSIBLE_WAVES_EXCEE DED	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_COM PONENT	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_ER ROR	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_PAR AMETER	ProfileGenerator: Unknown parameter ID in Set/Get Parameter command
-1070	PI_PROFILE_GENERATOR_PAR_OUT_OF_R ANGE	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_FIL E	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_REVIS ION	User stages DAT file does not have correct revision.
-1082	PI_SOFTWARE_TIMEOUT	Timeout Error. Some lengthy operation did not finish within expected time.
-1083	PI_WRONG_DATA_TYPE	A function argument has an unexpected data type.
-1084	PI_DIFFERENT_ARRAY_SIZES	Length of data arrays is different.
-1085	PI_PARAM_NOT_FOUND_IN_PARAM_DAT _FILE	Parameter value not found in parameter DAT file.
-1086	PI_MACRO_RECORDING_NOT_ALLOWED_I N_THIS_MODE	Macro recording is not allowed in this mode of operation.
-1087	PI_USER_CANCELLED_COMMAND	Command cancelled by user input.
-1088	PI_TOO_FEW_GCS_DATA	Controller sent too few GCS data sets
-1089	PI_TOO_MANY_GCS_DATA	Controller sent too many GCS data sets
-1090	PI_GCS_DATA_READ_ERROR	Communication error while reading GCS data
-1091	PI_WRONG_NUMBER_OF_INPUT_ARGUM ENTS	Wrong number of input arguments.
-1092	PI_FAILED_TO_CHANGE_CCL_LEVEL	Change of command level has failed.



-1093	PI_FAILED_TO_SWITCH_OFF_SERVO	Switching off the servo mode has failed.
-1094	PI_FAILED_TO_SET_SINGLE_PARAMETER_ WHILE_PERFORMING_CST	A parameter could not be set while performing CST: CST was not performed (parameters remain unchanged).
-1095	PI_ERROR_CONTROLLER_REBOOT	Connection could not be reestablished after reboot.
-1096	PI_ERROR_AT_QHPA	Sending HPA? or receiving the response has failed.
-1097	PI_QHPA_NONCOMPLIANT_WITH_GCS	HPA? response does not comply with GCS2 syntax.
-1098	PI_FAILED_TO_READ_QSPA	Response to SPA? could not be received.
-1099	PI_PAM_FILE_WRONG_VERSION	Version of PAM file cannot be handled (too old or too new)
-1100	PI_PAM_FILE_INVALID_FORMAT	PAM file does not contain required data in PAM-file format
-1101	PI_INCOMPLETE_INFORMATION	Information does not contain all required data
-1102	PI_NO_VALUE_AVAILABLE	No value for parameter available
-1103	PI_NO_PAM_FILE_OPEN	No PAM file is open
-1104	PI_INVALID_VALUE	Invalid value
-1105	PI_UNKNOWN_PARAMETER	Unknown parameter
-1106	PI_RESPONSE_TO_QSEP_FAILED	Response to SEP? could not be received.
-1107	PI_RESPONSE_TO_QSPA_FAILED	Response to SPA? could not be received.
-1108	PI_ERROR_IN_CST_VALIDATION	Error while performing CST: One or more parameters were not set correctly.
-1109	PI_ERROR_PAM_FILE_HAS_DUPLICATE_EN TRY_WITH_DIFFERENT_VALUES	PAM file has duplicate entry with different values.
-1110	PI_ERROR_FILE_NO_SIGNATURE	File has no signature
-1111	PI_ERROR_FILE_INVALID_SIGNATURE	File has invalid signature
-1112	PI_ERROR_CANNOT_DETERMINE_ACTUAL _END_OF_TRAVEL_WHILE_PLATFORM_IS_ MOVING	Cannot determine actual end of travel range while platform is moving.
-1113	PI_ERROR_AT_QIDN	Sending IDN? or receiving the response has failed.



-1114	PI_ERROR_AT_MAC_DEF	Sending MAC_DEF or receiving the response has failed.
-1115	PI_CONTROLLER_OR_CONTROLLER_VERSI ON_DOES_NOT_EXIST_IN_PISTAGES_DAT ABASE	Sending Controller or controller version does not exist in PIStages database.
-1116	PI_NOT_ENOUGH_MEMORY	Not enough memory
-1117	PI_ERROR_AXIS_RUNTIME_ERROR	Runtime error indicated for axis, check error log with \"LOG?\" to find more details.
-1118	PI_ERROR_SYSTEM_RUNTIME_CRITICAL_E RROR	Critical error indicated for system, check error log with \"LOG?\" to find more details.
-1119	PI_ERROR_CANNOT_START_EMULATOR	Cannot start emulation software.
-1120	COM_DEVICE_NOT_SUPPORTED	Device is not supported
-10000	PI_PARAMETER_DB_INVALID_STAGE_TYPE _FORMAT	PI stage database: String containing stage type and description has invalid format.
-10001	PI_PARAMETER_DB_SYSTEM_NOT_AVAIL ABLE	PI stage database: Database does not contain the selected stage type for the connected controller.
-10002	PI_PARAMETER_DB_FAILED_TO_ESTABLIS H_CONNECTION	PI stage database: Establishing the connection has failed.
-10003	PI_PARAMETER_DB_COMMUNICATION_E RROR	PI stage database: Communication was interrupted (e.g. because database was deleted).
-10004	PI_PARAMETER_DB_ERROR_WHILE_QUER YING_PARAMETERS	PI stage database: Querying data failed.
-10005	PI_PARAMETER_DB_SYSTEM_ALREADY_EX ISTS	PI stage database: System already exists. Rename stage and try again.
-10006	PI_PARAMETER_DB_QHPA_CONTANS_UN KNOWN_PAM_IDS	PI stage database: Response to HPA? contains unknown parameter IDs.
-10007	PI_PARAMETER_DB_AND_QHPA_ARE_INC ONSISTENT	PI stage database: Inconsistency between database and response to HPA?.
-10008	PI_PARAMETER_DB_SYSTEM_COULD_NOT _BE_ADDED	PI stage database: Stage has not been added.
-10009	PI_PARAMETER_DB_SYSTEM_COULD_NOT _BE_REMOVED	PI stage database: Stage has not been removed.



-10010	PI_PARAMETER_DB_CONTROLLER_DB_PA RAMETERS_MISMATCH	Controller does not support all stage parameters stored in PI stage database. No parameters were set.
-10011	PI_PARAMETER_DB_DATABASE_IS_OUTD ATED	The version of PISTAGES3.DB stage database is out of date. Please update via PIUpdateFinder. No parameters were set.
-10012	PI_PARAMETER_DB_AND_HPA_MISMATC H_STRICT	Mismatch between number of parameters present in stage database and available in controller interface. No parameters were set.
-10013	PI_PARAMETER_DB_AND_HPA_MISMATC H_LOOSE	Mismatch between number of parameters present in stage database and available in controller interface. Some parameters were ignored.
-10014	PI_PARAMETER_DB_FAILED_TO_SET_PAR AMETERS_CORRECTLY	One or more parameters could not be set correctly on the controller.
-10015	PI_PARAMETER_DB_MISSING_PARAMETE R_DEFINITIONS_IN_DATABASE	One or more parameter definitions are not present in stage database. Please update PISTAGES3.DB via PIUpdateFinder. Missing parameters were ignored.
-10016	PI_PARAMETER_DB_MISSING_FIRMWARE _FEATURE_ON_CONTROLLER	Parameters could not be set on controller because the corresponding firmware feature is missing



# 9 Anpassen von Einstellungen

# 9.1 Einstellungen des C-877

Die Eigenschaften des C-877 und des angeschlossenen Positionierers sind im C-877 als Parameterwerte hinterlegt (z. B. Einstellungen für den Regelalgorithmus (S. 21)).

Die Parameter können in folgende Kategorien eingeteilt werden:

- Geschützte Parameter, deren Werkseinstellung nicht geändert werden kann
- Parameter, die zur Anpassung an die Anwendung vom Benutzer eingestellt werden können

Das Schreibrecht für die Parameter ist durch Befehlsebenen festgelegt.

Jeder Parameter ist sowohl im flüchtigen als auch im permanenten Speicher des C-877 vorhanden. Die Werte im permanenten Speicher werden als Standardwerte beim Einschalten oder Neustart des C-877 in den flüchtigen Speicher geladen. Die Werte im flüchtigen Speicher bestimmen das aktuelle Verhalten des Systems.

In der PC-Software von PI werden auch die Bezeichnungen "Active Values" für die Parameterwerte im flüchtigen Speicher und "Startup Values" für die Parameterwerte im permanenten Speicher verwendet.

# 9.2 Parameterwerte im C-877 ändern

#### **HINWEIS**



#### **Unpassende Parametereinstellungen!**

Die Werte im permanenten Speicher werden als Standardwerte beim Einschalten oder Neustart des C-877 in den flüchtigen Speicher geladen und sind sofort gültig. Unpassende Parametereinstellungen können zur Beschädigung der angeschlossenen Mechanik führen.

- Ändern Sie Parameterwerte nur nach sorgfältiger Überlegung.
- Sichern Sie die aktuellen Parameterwerte auf dem PC (S. 169), bevor Sie Änderungen im permanenten Speicher durchführen.

#### **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. 169), bevor Sie Änderungen im permanenten Speicher durchführen.
- Wenden Sie sich an unseren Kundendienst (S. 195), wenn der C-877 ein unerwartetes Verhalten zeigt.



### **INFORMATION**

Wenn der angeschlossene Positionierer einen ID-Chip (S. 12) enthält, werden beim Einschalten oder Neustart des C-877 Daten aus dem ID-Chip in den flüchtigen Speicher des C-877 geladen.

Der ID-Chip enthält nur einen Teil der Informationen, die zum Betrieb des Positionierers mit dem C-877 erforderlich sind. Wenn Sie die PC-Software von PI verwenden, werden weitere Informationen als Parameterwerte aus einer Positioniererdatenbank (S. 11) in den flüchtigen Speicher des C-877 geladen.

Parameter, die vom ID-Chip oder aus einer Positioniererdatenbank geladen werden, sind in der Parameterübersicht (S. 176) farbig markiert.

# 9.2.1 Allgemeine Befehle für Parameter

Für Parameter stehen folgende allgemeine Befehle zur Verfügung:

Befehl	Funktion
CCL	Auf eine höhere Befehlsebene wechseln, um z. B. Schreibrecht auf bestimmte Parameter zu erhalten.
CCL?	Aktive Befehlsebene abfragen.
HPA?	Antwortet mit einem Hilfetext, der alle verfügbaren Parameter mit Kurzbeschreibungen enthält.
RPA	Parameterwert vom permanenten in den flüchtigen Speicher kopieren.
SEP	Parameter im permanenten Speicher ändern.
SEP?	Parameterwerte aus dem permanenten Speicher abfragen.
SPA	Parameter im flüchtigen Speicher ändern.
SPA?	Parameterwerte aus dem flüchtigen Speicher abfragen.
WPA	Aktuellen Parameterwert vom flüchtigen in den permanenten Speicher kopieren. Dort wird er als Standardwert verwendet.

Einzelheiten finden Sie in den Befehlsbeschreibungen (S. 84).

# 9.2.2 Befehle für Schnellzugriff auf einzelne Parameter

Die folgenden, speziellen Befehle ändern die zugehörigen Parameter nur im flüchtigen Speicher. Die geänderten Werte müssen bei Bedarf mit dem Befehl WPA (S. 143) 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 (0xB)
DEC	Abbremsung im geregelten Betrieb (0xC)
VEL	Geschwindigkeit im geregelten Betrieb (0x49)

Einzelheiten finden Sie in den Befehlsbeschreibungen (S. 84).

#### 9.2.3 Parameterwerte in Textdatei sichern

#### **INFORMATION**

Der C-877 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-877 eine Sicherungskopie auf dem PC an. Sie können dann jederzeit die Originaleinstellungen wiederherstellen.
- Erstellen Sie nach jeder Optimierung der Parameterwerte oder Anpassung des C-877 an eine bestimmte Mechanik eine weitere Sicherungskopie mit neuem Dateinamen.

#### **INFORMATION**

Parameterwerte, die in einer Textdatei auf dem PC gesichert wurden, können in PIMikroMove® oder PITerminal zurück auf den C-877 geladen werden. Im Fenster zum Senden von Befehlen ist dazu die Schaltfläche **Send file...** verfügbar. Vor dem Laden in den C-877 müssen die einzelnen Zeilen der Textdatei in Befehlszeilen umgewandelt werden, die entsprechende SPA-oder SEP-Befehle enthalten.

### Voraussetzungen

✓ Sie haben die Kommunikation zwischen dem C-877 und dem PC mit PIMikroMove® oder PITerminal hergestellt (S. 50).

### 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-877 sichern möchten: Senden Sie den Befehl SPA?.
  - Wenn Sie die Parameterwerte aus dem permanenten Speicher des C-877 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.

# 9.2.4 Parameterwerte ändern: Generelle Vorgehensweise

Für die Arbeit mit Parametern können Sie die allgemeinen Befehle (S. 168) und die Befehle für den Schnellzugriff (S. 168) verwenden.

Für den einfacheren Zugang zu Parametern wird im Folgenden PIMikroMove® verwendet, so dass Sie sich nicht mit den entsprechenden Befehlen auseinandersetzen müssen.

#### **HINWEIS**



#### Unpassende Parametereinstellungen!

Die Werte im permanenten Speicher werden als Standardwerte beim Einschalten oder Neustart des C-877 in den flüchtigen Speicher geladen und sind sofort gültig. Unpassende Parametereinstellungen können zur Beschädigung der angeschlossenen Mechanik führen.

- Ändern Sie Parameterwerte nur nach sorgfältiger Überlegung.
- Sichern Sie die aktuellen Parameterwerte auf dem PC (S. 169), bevor Sie Änderungen im permanenten Speicher durchführen.

# **INFORMATION**

Für das Ändern von Parameterwerten wird generell folgendes Vorgehen empfohlen:

- 1. Ändern Sie die Parameterwerte im flüchtigen Speicher.
- 2. Prüfen Sie, ob der C-877 mit den geänderten Parameterwerten korrekt funktioniert.

#### Wenn ja:

Schreiben Sie die geänderten Parameterwerte in den permanenten Speicher.

#### Wenn nein:

Ändern und prüfen Sie die Parameterwerte im flüchtigen Speicher erneut.

# **INFORMATION**

Der Schreibzugriff auf die Parameter des C-877 ist durch Befehlsebenen festgelegt. Nach dem Einschalten oder Neustart des Controllers ist die aktive Befehlsebene immer 0. Auf Befehlsebenen > 1 besteht Schreibzugriff nur für PI-Servicepersonal.

Wenn Sie Probleme mit Parametern der Befehlsebene 2 oder höher haben, wenden Sie sich an den Kundendienst (S. 195).

### Voraussetzungen

✓ Wenn Sie Parameterwerte im permanenten Speicher des C-877 ändern wollen: Sie haben die Parameterwerte des C-877 in einer Textdatei auf dem PC gesichert (S. 169).



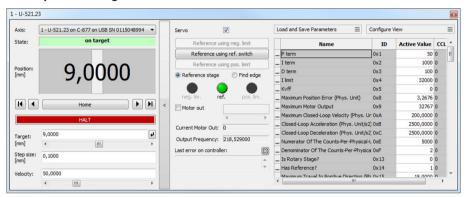
✓ Sie haben die Kommunikation zwischen dem C-877 und dem PC mit PIMikroMove® hergestellt (S. 50).

### Parameterwerte ändern: Generelle Vorgehensweise

1. Zeigen Sie in PIMikroMove® die Parameterliste an.

Wenn Sie die achsbezogenen Parameter des C-877 ändern wollen:

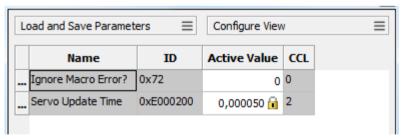
a) Öffnen Sie im Hauptfenster von PIMikroMove® das erweiterte Einzelachsen-Fenster für den angeschlossenen Positionierer, indem Sie mit der rechten Maustaste auf die entsprechende Zeile der Registerkarte *Axes* klicken und im Kontextmenü *Show Expanded Single Axis Window* auswählen.



b) Wenn der zu ändernde Parameter nicht in der Liste auf der rechten Seite des Fensters enthalten ist, klicken Sie auf *Configure View > Select parameters...* und fügen ihn zur Liste hinzu. Sie können auch bestimmte Gruppen von Parametern oder alle achsbezogenen Parameter einblenden.

Wenn Sie die systembezogenen Parameter des C-877 ändern wollen:

 Öffnen Sie im Hauptfenster von PIMikroMove® das Fenster für die systembezogenen Parameter des C-877, indem Sie im Menü C-877 > Show system parameters auswählen.



2. Ändern Sie in der entsprechenden Parameterliste die gewünschten Parameterwerte im flüchtigen oder permanenten Speicher des C-877.

Wenn Sie Parameterwerte im flüchtigen Speicher ändern wollen, haben Sie folgende Optionen:

 Tippen Sie den neuen Parameterwert in das entsprechende Eingabefeld in der Spalte Active Value der Liste ein. Drücken Sie auf der Tastatur des PC die Enter-Taste oder klicken Sie mit der Maus außerhalb des Eingabefeldes, um den Parameterwert in den flüchtigen Speicher des C-877 zu übertragen.



- Klicken Sie auf Load and Save Parameters -> Load all startup parameters of the axis / system from controller, um die Werte aller achsenbezogenen / systembezogenen Parameter aus dem permanenten Speicher des C-877 zu laden.
- Klicken Sie im erweiterten Einzelachsen-Fenster auf Load and Save Parameters > Load parameters from stage database..., um für die Achse einen ausgewählten Parametersatz aus der Positioniererdatenbank zu laden. Mit Load and Save Parameters > Reload parameters from stage database... können Sie den aktuell geladenen Parametersatz erneut laden.

Wenn Sie Parameterwerte im permanenten Speicher ändern wollen, haben Sie folgende Optionen:

- Tippen Sie den neuen Parameterwert in das entsprechende Eingabefeld in der Spalte Startup Value der Liste ein. Drücken Sie auf der Tastatur des PC die Enter-Taste oder klicken Sie mit der Maus außerhalb des Eingabefeldes, um den Parameterwert in den permanenten Speicher des C-877 zu übertragen.
- Klicken Sie auf Load and Save Parameters -> Save all currently active axis / system
  parameters as startup parameters to controller, um die Werte aller
  achsenbezogenen / systembezogenen Parameter aus dem flüchtigen in den
  permanenten Speicher des C-877 zu schreiben. Sie können Parameter
  überspringen, für die auf der aktuellen Befehlsebene kein Schreibzugriff besteht.

Wenn ein Parameterwert im flüchtigen Speicher (Spalte *Active Value*) vom Parameterwert im permanenten Speicher (Spalte *Startup Value*) abweicht, ist die Zeile in der Liste farbig markiert.

# 9.3 Positionierertyp anlegen oder ändern

Sie können in der PC-Software von PI den für Ihren Positionierer geeigneten Parametersatz aus einer Positioniererdatenbank auswählen. Die Software überträgt die Werte des ausgewählten Parametersatzes in den flüchtigen oder permanenten Speicher des Controllers. Weitere Informationen siehe "Positioniererdatenbanken" (S. 11).

In der Positioniererdatenbank PI\_UserStages2.dat können Sie neue Parametersätze anlegen und bearbeiten. Dies kann z. B. in folgenden Fällen erforderlich sein:

- Sie möchten einen Positionierer mit anderen Regelparameter-Einstellungen als denjenigen aus dem Standard-Parametersatz betreiben.
- Sie möchten die Verfahrbereichsgrenzen des Positionierers an Ihre Anwendung anpassen.
- Sie haben einen kundenspezifischen Positionierer.

#### **INFORMATION**

Einen neuen Positionierertyp können Sie am einfachsten anlegen, indem Sie in PIMikroMove® einen vorhandenen Positionierertyp ändern und ihn unter einer neuen Benennung abspeichern.



#### **INFORMATION**

Wenn in der Standard-Positioniererdatenbank (PIStages2.dat oder PIMicosStages2.dat) und in der Datenbank PI\_UserStages2.dat ein Positionierertyp mit gleicher Benennung vorhanden ist, werden bei der Auswahl dieses Positionierertyps in der PC-Software immer die Parametereinstellungen aus der Standard-Positioniererdatenbank geladen. Die Parametereinstellungen aus PI\_UserStages2.dat werden in diesem Fall nicht verwendet.

Vergeben Sie beim Speichern von Positionierertypen nur Benennungen, die nicht bereits in der Positioniererdatenbank PIStages2.dat oder PIMicosStages2.dat verwendet werden.

Im Folgenden wird PIMikroMove® zum Anlegen eines neuen Positionierertyps und zum Ändern eines vorhandenen Positionierertyps verwendet.

#### Voraussetzungen

- ✓ Sie haben die neueste Version der Positioniererdatenbanken pistages2.dat und pimicosstages2.dat auf dem PC installiert (S. 45).
- ✓ Wenn Sie für Ihren Positionierer eine kundenspezifische Positioniererdatenbank von PI erhalten haben, dann haben Sie diese Datenbank auf Ihrem PC installiert (S. 46).
- ✓ Sie haben die Kommunikation zwischen dem C-877 und dem PC mit PIMikroMove® hergestellt (S. 50).

# Positionierertyp in Positioniererdatenbank anlegen

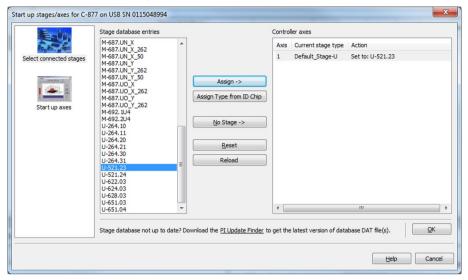
- 1. Wählen Sie im Hauptfenster von PIMikroMove® den Menüeintrag *C-877 > Select connected stages...*.
  - Das Fenster **Start up stages/axes for C-877** öffnet sich, der Schritt **Select connected stages** ist aktiv.
- 2. Wählen Sie im Schritt Select connected stages einen passenden Positionierertyp aus:
  - Klicken Sie auf Assign Type from ID Chip.

oder

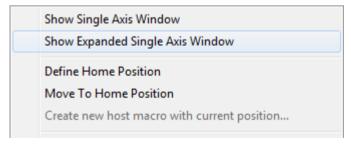
a) Markieren Sie den Positionierertyp in der Liste Stage database entries.



b) Klicken Sie auf Assign.



- c) Bestätigen Sie die Auswahl mit OK.
- 3. Klicken Sie im Dialog **Save all changes permanently?** auf **Keep the changes temporarily**, um die Parametereinstellungen in den flüchtigen Speicher des C-877 zu laden.
  - Das Fenster Start up stages/axes wechselt zum Schritt Start up axes.
- 4. Klicken Sie im Schritt **Start up axes** auf **Close**, um das Fenster **Start up stages/axes** zu schließen.
- 5. Öffnen Sie im Hauptfenster von PIMikroMove® das erweiterte Einzelachsen-Fenster für den ausgewählten Positionierertyp, indem Sie mit der rechten Maustaste auf die entsprechende Zeile der Registerkarte Axes klicken und im Kontextmenü Show Expanded Single Axis Window auswählen.

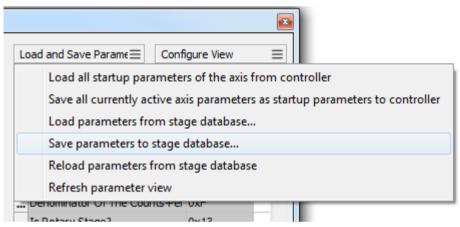


6. Geben Sie neue Werte für die zu ändernden Parameter ein:





- a) Wenn der zu ändernde Parameter nicht in der Liste auf der rechten Seite des Fensters enthalten ist, klicken Sie auf *Configure view > Select parameters...* und fügen ihn zur Liste hinzu. Sie können auch bestimmte Gruppen von Parametern oder alle achsenbezogenen Parameter einblenden.
- b) Tippen Sie den neuen Parameterwert in das entsprechende Eingabefeld in der Spalte *Active Value* der Liste ein.
- c) Drücken Sie auf der Tastatur des PC die Enter-Taste oder klicken Sie mit der Maus außerhalb des Eingabefeldes, um den Parameterwert in den flüchtigen Speicher des Controllers zu übertragen. Anmerkung: Wenn ein Parameterwert im flüchtigen Speicher (Spalte Active Value) vom Parameterwert im permanenten Speicher (Spalte Startup Value) abweicht, ist die Zeile in der Liste farbig markiert.
- 7. Klicken Sie auf Load and Save Parameters -> Save parameters to stage database....



Der Dialog Save Parameters as User Stage Type öffnet sich.

- 8. Speichern Sie im Dialog *Save Parameters as User Stage Type* die geänderten Parameterwerte als neuen Positionierertyp:
  - a) Lassen Sie den Eintrag im Feld **Parameters of axis** unverändert.
  - b) Tragen Sie im Feld *Save as* die Benennung für den neuen Positionierertyp ein.
  - c) Klicken Sie auf OK.

Der neue Positionierertyp wurde in der Positioniererdatenbank PI\_UserStages2.dat gespeichert. Die Anzeige des angeschlossenen Positionierertyps wurde im Einzelachsen-Fenster und im Hauptfenster von PIMikroMove® aktualisiert. Der neue Positionierertyp steht ab sofort auch für die Auswahl im Schritt **Select connected stages** zur Verfügung.

#### Positionierertyp in Positioniererdatenbank ändern

- 1. Wählen Sie im Hauptfenster von PIMikroMove® den Menüeintrag *C-877 > Select connected stages...*.
  - Das Fenster **Start up stages/axes for C-877** öffnet sich, der Schritt **Select connected stages** ist aktiv.
- Wählen Sie im Schritt Select connected stages einen Positionierertyp aus, den Sie zuvor wie oben beschrieben (S. 173) neu angelegt haben. Gehen Sie bei der Auswahl vor wie in Schritt 2 der Anleitung Positionierertyp in Positioniererdatenbank anlegen beschrieben.

C-877 PILine® Controller MS232D Version: 2.0.0 175



- 3. Führen Sie die Schritte 3 bis 7 aus **Positionierertyp in Positioniererdatenbank anlegen** aus.
- 4. Speichern Sie im Dialog *Save Parameters as User Stage Type* die geänderten Parameterwerte des Positionierertyps:
  - 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 Positionierertyps wurden in der Positioniererdatenbank PI UserStages2.dat sowie im Hauptfenster von PIMikroMove® aktualisiert.

#### 9.4 Parameterübersicht

#### **INFORMATION**

Der Schreibzugriff auf die Parameter des C-877 ist durch Befehlsebenen festgelegt. Nach dem Einschalten oder Neustart des Controllers ist die aktive Befehlsebene immer 0. Auf Befehlsebenen > 1 besteht Schreibzugriff nur für PI-Servicepersonal.

Wenn Sie Probleme mit Parametern der Befehlsebene 2 oder höher haben, wenden Sie sich an den Kundendienst (S. 195).

#### **INFORMATION**

Das Kennwort zum Speichern der Parameterwerte im permanenten Speicher lautet 100.

Bedeutung der farblichen Unterlegung in der Parametertabelle:

Dunkelgrau:	Der Wert des Parameters wird aus dem ID-Chip des Positionierers (S. 12) geladen.
Hellgrau:	Der Wert des Parameters kann aus einer Positioniererdatenbank (S. 11) geladen werden.
Farblos:	Der Wert des Parameter kann nur per Befehl (SPA, SEP, spezielle Befehle (S. 168)) oder durch die Verwendung entsprechender Bedienelemente der PC-Software geändert werden (S. 170).

Bezeichnungen in der Kopfzeile der nachfolgenden Tabelle:

- ID = Parameter-ID, Hexadezimal-Format
- Typ = Datentyp:
  - INT = ganzzahliger Wert, inkl. Boolesche Werte
  - FLOAT = Gleitkommazahl



- CHAR = Stringformat
- CL = Befehlsebene (Command Level) für Schreibzugriff
- Element = Elementtyp, auf den sich der Parameter bezieht, weitere Informationen siehe "Kommandierbare Elemente" (S. 13)
- Parametername = Name des Parameters
- Beschreibung = Erläuterungen zum Parameter

ID	Тур	CL	Element	Parametername	Beschreibung
0x1	INT	0	Achse	P Term	Nur aus Kompatibilitätsgründen
0x2	INT	0	Achse	I Term	vorhanden Die Werte dieser Parameter werden
0x3	INT	0	Achse	D Term	automatisch auf die Werte der
0x4	INT	0	Achse	I Limit	Parameter 0x411, 0x412, 0x413, 0x414 und 0x415 eingestellt.
0x5	INT	0	Achse	Kvff	OATT UND OATTS EMIGESTEIN.
0x8	FLOAT	0	Achse	Maximum Position Error (Phys. Unit)	Maximaler Positionsfehler Wird für die Erkennung von Bewegungsfehlern verwendet. Details siehe "Verhalten bei Bewegungsfehler" (S. 63).
0x9	INT	0	Achse	Maximum Motor Output	Maximal zulässiger Betrag des Stellwerts (dimensionslos) Details siehe "Unterstützte Motortypen" (S. 28).
0xA	FLOAT	0	Achse	Maximum Closed Loop Velocity (Phys. Unit/s)	Maximale Geschwindigkeit im geregelten Betrieb Gibt den Maximalwert für Parameter 0x49 an.
OxB	FLOAT	0	Achse	Closed Loop Acceleration (Phys. Unit/s²)	Beschleunigung im geregelten Betrieb Details siehe "Erzeugung des Dynamikprofils" (S. 18). Wird begrenzt durch Parameter 0x4A.
0xC	FLOAT	0	Achse	Closed-Loop Deceleration (Phys. Unit/s²)	Abbremsung im geregelten Betrieb Details siehe "Erzeugung des Dynamikprofils" (S. 18). Wird begrenzt durch Parameter 0x4B.
0xE	INT	0	Achse	Numerator Of The Counts- Per-Physical-Unit-Factor	Zähler und Nenner des Faktors für Impulse pro physikalische



ID	Тур	CL	Element	Parametername	Beschreibung
0xF	INT	0	Achse	Denominator Of The Counts- Per-Physical-Unit-Factor	Längeneinheit Details siehe "Physikalische Einheiten" (S. 16).
0x13	INT	0	Achse	Is Rotary Stage?	Handelt es sich um einen Rotationstisch?  0 = Kein Rotationstisch 1 = Rotationstisch Keine Auswertung durch den C-877, sondern nur durch die PC-Software: PIMikroMove® entscheidet anhand dieses Wertes, welche Bewegungen zulässig sind.
0x14	INT	0	Achse	Has Reference?	Hat der Positionierer einen Referenzschalter? Details siehe "Referenzschaltererkennung" (S. 29).
0x15	FLOAT	0	Achse	Maximum Travel In Positive Direction (Phys. Unit)	Verfahrbereichsgrenze in positiver Richtung Siehe Beispiele in "Stellweg und Verfahrbereichsgrenzen" (S. 32).
0x16	FLOAT	0	Achse	Value At Reference Position (Phys. Unit)	Positionswert am Referenzschalter Siehe Beispiele in "Stellweg und Verfahrbereichsgrenzen" (S. 32).
0x17	FLOAT	0	Achse	Distance From Negative Limit To Reference Position (Phys. Unit)	Abstand zwischen Referenzschalter und negativem Endschalter Siehe Beispiele in "Stellweg und Verfahrbereichsgrenzen" (S. 32).
0x18	INT	0	Achse	Limit Mode	Signallogik der Endschalter Details siehe "Endschaltererkennung" (S. 30).
0x2F	FLOAT	0	Achse	Distance From Reference Position To Positive Limit (Phys. Unit)	Abstand zwischen Referenzschalter und positivem Endschalter Siehe Beispiele in "Stellweg und Verfahrbereichsgrenzen" (S. 32).
0x30	FLOAT	0	Achse	Maximum Travel In Negative Direction (Phys. Unit)	Verfahrbereichsgrenze in negativer Richtung Siehe Beispiele in "Stellweg und Verfahrbereichsgrenzen" (S. 32).



ID	Тур	CL	Element	Parametername	Beschreibung
0x31	INT	0	Achse	Invert Reference?	Soll das Referenzsignal invertiert werden? Details siehe "Referenzschaltererkennung" (S. 29).
0x32	INT	0	Achse	Has No Limit Switches?	Hat der Positionierer keine Endschalter? Details siehe "Endschaltererkennung" (S. 30).
0x33	INT	0	Achse	Motor Offset Positive	Offset für die positive Bewegungsrichtung Details siehe "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 21).
0x34	INT	0	Achse	Motor Offset Negative	Offset für die negative Bewegungsrichtung Details siehe "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 21).
0x36	INT	0	Achse	Settling Window (encoder counts)	Nur aus Kompatibilitätsgründen vorhanden Der Wert dieses Parameters wird automatisch auf den Werte des Parameters 0x406 eingestellt.
0x3C	CHAR	0	Achse	Stage Name	Positionierername Maximal 20 Zeichen; Standardwert: NOSTAGE Der Wert NOSTAGE "deaktiviert" die Achse. Eine deaktivierte Achse ist nicht für achsenbezogene Befehle zugänglich (z.B. Bewegungsbefehle oder Positionsabfragen).
0x3F	FLOAT	0	Achse	Settling Time (s)	Verzögerungszeit für das Setzen des On-Target-Status. Details siehe "On-Target-Status" (S. 27).
0x47	INT	0	Achse	Reference Travel Direction	Standardrichtung für die Referenzfahrt Details siehe "Referenzwertbestimmung" (S. 34).
0x48	INT	0	Achse	Motor Drive Offset	Geschwindigkeitsabhängiger Offset Details siehe "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 21).



ID	Тур	CL	Element	Parametername	Beschreibung
0x49	FLOAT	0	Achse	Closed-Loop Velocity (Phys. Unit/s)	Geschwindigkeit im geregelten Betrieb Details siehe "Erzeugung des Dynamikprofils" (S. 18). Wird begrenzt durch Parameter 0xA
0x4A	FLOAT	0	Achse	Maximum Closed-Loop Acceleration (Phys. Unit/s²)	Maximale Beschleunigung im geregelten Betrieb Gibt den Höchstwert für Parameter OxB an.
0x4B	FLOAT	0	Achse	Maximum Closed-Loop Deceleration (Phys. Unit/s²)	Maximale Abbremsung im geregelten Betrieb Gibt den Höchstwert für Parameter 0xC an.
0x4D	INT	0	Achse	Servo Window Mode	Bezugsgröße für Positionsfenster zum Umschalten zwischen Parametergruppen Details siehe "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 21).
0x50	FLOAT	0	Achse	Velocity For Reference Moves (Phys. Unit/s)	Höchstgeschwindigkeit für Referenzfahrt Details siehe "Referenzwertbestimmung" (S. 34).
0x51	FLOAT	0	Achse	Output Frequency (kHz)	Frequenz der Piezospannung Details siehe "Unterstützte Motortypen" (S. 28).
0x52	INT	0	Achse	Frequency Control	Zustand der Frequenzregelung Details siehe "Frequenzregelung" (S. 28).
0x53	FLOAT	0	Achse	Minimum Output Frequency (kHz)	Minimale Frequenz der Piezospannung Details siehe "Frequenzregelung" (S. 28).
0x54	FLOAT	0	Achse	Maximum Output Frequency (kHz)	Maximale Frequenz der Piezospannung Details siehe "Frequenzregelung" (S. 28).
0x55	INT	0	Achse	Minimum Motor Output For Frequency Control	Minimaler Stellwert für Aktivierung der Frequenzregelung Details siehe "Frequenzregelung" (S. 28).
0x5A	INT	0	Achse	Numerator Of The Servo-Loop Input Factor	Zähler und Nenner des Eingangsfaktors des Regelkreises



ID	Тур	CL	Element	Parametername	Beschreibung
0x5B	INT	0	Achse	Denominator Of The Servo- Loop Input Factor	Details siehe "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 21).
0x5C	INT	0	Achse	Source Of Reference Signal	Referenzsignalquelle für die Befehle FRF (S. 105) oder FED (S. 102)
0x5D	INT	0	Achse	Source Of Negative Limit Signal	Referenzsignalquelle für die Befehle FNL (S. 103) oder FED (S. 102)
0x5E	INT	0	Achse	Source of Positive Limit Signal	Referenzsignalquelle für die Befehle FPL (S. 104) oder FED (S. 102)
0x5F	INT	0	Achse	Invert Digital Input Used For Negative Limit	Invertiert die Polarität der digitalen Eingänge, die als Quellen des negativen Endschaltersignals dienen.
0x60	INT	0	Achse	Invert Digital Input Used For Positive Limit	Invertiert die Polarität der digitalen Eingänge, die als Quellen des positiven Endschaltersignals dienen.
0x63	FLOAT	0	Achse	Distance Between Limit And Hard Stop (Phys. Unit)	Abstand zwischen eingebautem Endschalter und mechanischem Anschlag Details siehe "Referenzwertbestimmung" (S. 34).
0x64	INT	0	Achse	Phase Shift	Phasenverschiebung zwischen Strom und Spannung am Antrieb
0x70	INT	0	Achse	Reference Signal Mode	Art des Referenzsignals Details siehe "Referenzschaltererkennung" (S. 29).
0x71	INT	0	Achse	D-Term Delay (No. Of Servo Cycles)	D-Term-Verzögerung Details siehe "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 21).
0x72	INT	0	System	Ignore Macro Error?	Makrofehler ignorieren? Details siehe "Befehle und Parameter für Makros" (S. 67).
0x77	INT	0	Achse	Use Limit Switches Only For Reference Moves	Sollen die Endschalter nur für Referenzfahrten verwendet werden? Details siehe "Endschaltererkennung" (S. 30).
0x78	FLOAT	0	Achse	Distance From Limit To Start Of Ref Search (Phys. Unit)	Abstand zwischen Endschalter und der Startposition für die Referenzfahrt zum Indexpuls Details siehe "Referenzwertbestimmung" (S. 34).



ID	Тур	CL	Element	Parametername	Beschreibung
0x79	FLOAT	0	Achse	Distance For Reference Search (Phys. Unit)	Maximale Strecke für die Referenzfahrt zum Indexpuls Details siehe "Referenzwertbestimmung" (S. 34).
0x7B	FLOAT	0	Achse	PID Maximum Output Time (s)	Maximale Zeitdauer, für die ein hoher Stellwert im geregelten Betrieb gesetzt sein darf. Details siehe "Schutz vor Überhitzung" (S. 63).
0x7C	FLOAT	0	Achse	Maximum Motor Output (V)	Maximal zulässige Piezospannung Details siehe "Unterstützte Motortypen" (S. 28).
0x400	INT	0	Achse	Number of Servo Parameter Groups	Maximale Anzahl der verwendeten Parametergruppen Details siehe "Regelalgorithmus und weitere Stellwertkorrekturen" (S. 21).
0x401	INT	0	Achse	P Term 0	Proportionalkonstante der Parametergruppe 0
0x402	INT	0	Achse	I Term 0	Integralkonstante der Parametergruppe 0
0x403	INT	0	Achse	D Term 0	Differenzialkonstante der Parametergruppe 0
0x404	INT	0	Achse	I Limit 0	Begrenzung der Integralkonstante der Parametergruppe 0
0x405	INT	0	Achse	Kvff 0	Vorwärtsregelung der kommandierten Geschwindigkeit für Parametergruppe 0
0x406	INT	0	Achse	Window Enter 0	Positionsfenster für das Aktivieren der Parametergruppe 0 Weitere Details siehe "On-Target-Status" (S. 27)
0x407	INT	0	Achse	Window Exit 0	Positionsfenster für das Deaktivieren der Parametergruppe 0 Weitere Details siehe "On-Target-Status" (S. 27)
0x409	INT	0	Achse	2nd Phase On 0 (No. Of Servo Cycles)	Einschaltdauer der parallelen Ansteuerung beider Piezosegmente für Parametergruppe 0 Details siehe "Optionale Zweiphasenansteuerung" (S. 26)



ID	Тур	CL	Element	Parametername	Beschreibung
0x40A	INT	0	Achse	2nd Phase Off 0 (No. Of Servo Cycles)	Pausendauer der parallelen Ansteuerung beider Piezosegmente für Parametergruppe 0 Details siehe "Optionale Zweiphasenansteuerung" (S. 26)
0x411	INT	0	Achse	P Term 1	Proportionalkonstante der Parametergruppe 1
0x412	INT	0	Achse	I Term 1	Integralkonstante der Parametergruppe 1
0x413	INT	0	Achse	D Term 1	Differenzialkonstante der Parametergruppe 1
0x414	INT	0	Achse	I Limit 1	Begrenzung der Integralkonstante der Parametergruppe 1
0x415	INT	0	Achse	Kvff 1	Vorwärtsregelung der kommandierten Geschwindigkeit für Parametergruppe 1
0x416	INT	0	Achse	Window Enter 1	Positionsfenster für das Aktivieren der Parametergruppe 1 Weitere Details siehe "On-Target- Status" (S. 27)
0x417	INT	0	Achse	Window Exit 1	Positionsfenster für das Deaktivieren der Parametergruppe 1 Weitere Details siehe "On-Target-Status" (S. 27)
0x419	INT	0	Achse	2nd Phase On 1 (No. Of Servo Cycles)	Einschaltdauer der parallelen Ansteuerung beider Piezosegmente für Parametergruppe 1 Details siehe "Optionale Zweiphasenansteuerung" (S. 26)
0x41A	INT	0	Achse	2nd Phase Off 1 (No. Of Servo Cycles)	Pausendauer der parallelen Ansteuerung beider Piezosegmente für Parametergruppe 1 Details siehe "Optionale Zweiphasenansteuerung" (S. 26)
0x421	INT	0	Achse	P Term 2	Proportionalkonstante der Parametergruppe 2
0x422	INT	0	Achse	I Term 2	Integralkonstante der Parametergruppe 2
0x423	INT	0	Achse	D Term 2	Differenzialkonstante der Parametergruppe 2
0x424	INT	0	Achse	I Limit 2	Begrenzung der Integralkonstante der Parametergruppe 2



ID	Тур	CL	Element	Parametername	Beschreibung	
0x425	INT	0	Achse	Kvff 2	Vorwärtsregelung der kommandierten Geschwindigkeit für Parametergruppe 2	
0x426	INT	0	Achse	Window Enter 2	Positionsfenster für das Aktivieren der Parametergruppe 2 Weitere Details siehe "On-Target- Status" (S. 27)	
0x427	INT	0	Achse	Window Exit 2	Positionsfenster für das Deaktivieren der Parametergruppe 2 Weitere Details siehe "On-Target-Status" (S. 27)	
0x429	INT	0	Achse	2nd Phase On 2 (No. Of Servo Cycles)	Einschaltdauer der parallelen Ansteuerung beider Piezosegmente für Parametergruppe 2 Details siehe "Optionale Zweiphasenansteuerung" (S. 26)	
0x42A	INT	0	Achse	2nd Phase Off 2 (No. Of Servo Cycles)	Pausendauer der parallelen Ansteuerung beider Piezosegmente für Parametergruppe 2 Details siehe "Optionale Zweiphasenansteuerung" (S. 26)	
0x431	INT	0	Achse	P Term 3	Proportionalkonstante der Parametergruppe 3	
0x432	INT	0	Achse	I Term 3	Integralkonstante der Parametergruppe 3	
0x433	INT	0	Achse	D Term 3	Differenzialkonstante der Parametergruppe 3	
0x434	INT	0	Achse	I Limit 3	Begrenzung der Integralkonstante der Parametergruppe 3	
0x435	INT	0	Achse	Kvff 3	Vorwärtsregelung der kommandierten Geschwindigkeit für Parametergruppe 3	
0x436	INT	0	Achse	Window Enter 3	Positionsfenster für das Aktivieren der Parametergruppe 3	
0x437	INT	0	Achse	Window Exit 3	Positionsfenster für das Deaktivieren der Parametergruppe 3	
0x439	INT	0	Achse	2nd Phase On 3 (No. Of Servo Cycles)	Einschaltdauer der parallelen Ansteuerung beider Piezosegmente für Parametergruppe 3	
0x43A	INT	0	Achse	2nd Phase Off 3 (No. Of Servo Cycles)	Pausendauer der parallelen Ansteuerung beider Piezosegmente für Parametergruppe 3	
0x441	INT	0	Achse	P Term 4	Proportionalkonstante der Parametergruppe 4	



ID	Тур	CL	Element	Parametername	Beschreibung
0x442	INT	0	Achse	I Term 4	Integralkonstante der Parametergruppe 4
0x443	INT	0	Achse	D Term 4	Differenzialkonstante der Parametergruppe 4
0x444	INT	0	Achse	I Limit 4	Begrenzung der Integralkonstante der Parametergruppe 4
0x445	INT	0	Achse	Kvff 4	Vorwärtsregelung der kommandierten Geschwindigkeit für Parametergruppe 4
0x446	INT	0	Achse	Window Enter 4	Positionsfenster für das Aktivieren der Parametergruppe 4
0x447	INT	0	Achse	Window Exit 4	Positionsfenster für das Deaktivieren der Parametergruppe 4
0x449	INT	0	Achse	2nd Phase On 4 (No. Of Servo Cycles)	Einschaltdauer der parallelen Ansteuerung beider Piezosegmente für Parametergruppe 4
0x44A	INT	0	Achse	2nd Phase Off 4 (No. Of Servo Cycles)	Pausendauer der parallelen Ansteuerung beider Piezosegmente für Parametergruppe 4
0x7000000	FLOAT	0	Achse	Range Limit Min	Zusätzliche Verfahrbereichsgrenze für die negative Bewegungsrichtung (physikalische Einheit) Details siehe "Stellweg und Verfahrbereichsgrenzen" (S. 31).
0x7000001	FLOAT	0	Achse	Range Limit Max	Zusätzliche Verfahrbereichsgrenze für die positive Bewegungsrichtung (physikalische Einheit) Details siehe "Stellweg und Verfahrbereichsgrenzen" (S. 31).
0x7000601	CHAR	0	Achse	Axis Unit	Einheitenzeichen Details siehe "Physikalische Einheiten" (S. 16).
0xE000200	FLOAT	0	System	Servo Update Time	Servo-Zykluszeit in Sekunden
0xF000100	CHAR	2	Achse	Stage Type	Positionierertyp Form für Standardpositionierer: x-xxx Form für kundenspezifische Positionierer: x-xxxKxxx
0xF000200	CHAR	2	Achse	Stage Serial Number	Seriennummer des Positionierers 9-stellige Nummer
0xF000300	CHAR	2	Achse	Stage Assembly Date	Herstellungsdatum des Positionierers Datumsformat: TTMMJJ



ID	Тур	CL	Element	Parametername	Beschreibung
0xF000400	INT	2	Achse		Versionsnummer der Positioniererhardware



## 10 Wartung

### **10.1 C-877** reinigen

#### **HINWEIS**



#### Kurzschlüsse oder Überschläge!

Der C-877 enthält elektrostatisch gefährdete Bauteile, die beim Eindringen von Reinigungsflüssigkeiten in das Gehäuse durch Kurzschlüsse oder Überschläge beschädigt werden können.

- Trennen Sie vor dem Reinigen den C-877 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-877 mit einem Tuch, das leicht mit einem milden Reinigungs- oder Desinfektionsmittel angefeuchtet wurde.

#### 10.2 Firmware aktualisieren

#### **INFORMATION**

Der Befehl \*IDN? liest unter anderem die Versionsnummer der Firmware aus. Beispiel für eine Antwort des C-877:

Beispiel für eine Antwort des C-8//:

(c)2024 Physik Instrumente (PI) GmbH & Co. KG, C-877.1U11, 117048994, 01.400

- C-877.1U11: Gerätebezeichnung
- 117048994: Seriennummer des Geräts
- 01.400: Firmware-Version

#### **INFORMATION**

Wenn sich der C-877 im Firmware-Update-Modus befindet, ist die LED **STA** aus. Der C-877 verlässt den Firmware-Update-Modus erst, wenn er nach einer **erfolgreichen** Aktualisierung der Firmware **neu gestartet** wird. Wenn die Aktualisierung der Firmware nicht erfolgreich war oder abgebrochen wurde, bleibt der C-877 nach einem Neustart im Firmware-Update-Modus. Wenn die LED **STA** noch aus ist, obwohl der C-877 nach der Firmware-Aktualisierung neu gestartet wurde:

Wiederholen Sie die Aktualisierung der Firmware.



Wenn die erneute Aktualisierung der Firmware fehlschlägt, kontaktieren Sie unseren Kundendienst (S. 195).

#### **INFORMATION**

Wenn mit der Aktualisierung der Firmware neue Parameter eingeführt werden oder das Speichermanagement des C-877 verändert wird, ist nach der Aktualisierung der Firmware eine Initialisierung des C-877 erforderlich.

#### Voraussetzungen

- ✓ Sie haben den C-877 über die USB-Schnittstelle (S. 47) an den PC angeschlossen.
- ✓ Das Programm *PIFirmwareManager* ist auf dem PC installiert (S. 44).
- ✓ Sie haben die neue Firmwaredatei, die Sie von unserem Kundendienst erhalten haben, in ein Verzeichnis auf dem PC kopiert.
- ✓ Sie haben die Dokumentation gelesen und verstanden, die Sie mit der neuen Firmware von unserem Kundendienst erhalten haben. Der Dokumentation haben Sie entnommen, ob mit der Aktualisierung der Firmware neue Parameter eingeführt werden oder das Speichermanagement des C-877 verändert wird.
- ✓ Sie haben die Parameterwerte des C-877 in einer Textdatei auf dem PC gesichert (S. 169).
- ✓ Sie haben die Controllermakros des C-877 in Dateien auf dem PC gesichert (S. 75).
- ✓ Sie haben die Kommunikation zwischen dem C-877 und dem PC mit PIMikroMove® oder PITerminal hergestellt (S. 50).

#### Firmware des C-877 aktualisieren

1. Starten Sie am PC das Programm *PIFirmwareManager* und führen Sie die Aktualisierung der Firmware des Controllers durch.

Gehen Sie dazu vor wie im Benutzerhandbuch SM164E (S. 3) beschrieben.

#### Neustart des C-877 durchführen

- 1. Schalten Sie den C-877 aus.
- 2. Schalten Sie den C-877 wieder ein.

Wenn die Aktualisierung der Firmware erfolgreich war, hat der C-877 den Firmware-Update-Modus verlassen, und die LED **STA** leuchtet grün.

Wurden mit der Aktualisierung der Firmware neue Parameter eingeführt, oder wurde das Speichermanagement des C-877 verändert?

- Wenn nein: Die Aktualisierung der Firmware ist beendet.
- Wenn ja: Eine Initialisierung des C-877 ist erforderlich, siehe unten.

#### C-877 nach Firmware-Aktualisierung initialisieren

Die Initialisierung des C-877 setzt **alle** Parameter auf ihre Werkseinstellung zurück und löscht alle Controllermakros. Ungesicherte Parameterwerte und Controllermakros gehen somit bei der Initialisierung verloren.



- 1. Stellen Sie sicher, dass die aktuellen Parameterwerte und Controllermakros des C-877 auf dem PC gesichert wurden.
- 2. Starten Sie am PC PITerminal oder PIMikroMove®, stellen Sie die Verbindung zum C-877 her, und öffnen Sie gegebenenfalls das Fenster zum Senden von Befehlen.
- 3. Initialisieren Sie den C-877, indem Sie nacheinander folgende Befehle senden:

ZZZ 100 parameter ZZZ 100 macros

Nach erfolgreicher Initialisierung gibt der Controller jeweils eine entsprechende Meldung aus.

4. Passen Sie die Parameterwerte des C-877 an.

Eine Anleitung zum allgemeinen Vorgehen finden Sie in "Parameterwerte ändern: Generelle Vorgehensweise" (S. 170).

- 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.
- 5. Wenn Sie Controllermakros auf dem PC gesichert haben: Laden Sie die Controllermakros zurück in den C-877, siehe "Controllermakros sichern und laden" (S. 75).



# 11 Störungsbehebung

Störung: Positionierer bewegt sich nicht		
Mögliche Ursachen	Maßnahmen zur Behebung	
Kabel nicht korrekt angeschlossen	Prüfen Sie die Kabelanschlüsse.	
Positionierer wurde an den eingeschalteten C-877 angeschlossen	Die Sensorelektronik im Positionierer wurde nicht initialisiert, und der ID-Chip des Positionierers (S. 12) wurde nicht ausgelesen.  ➤ Schalten Sie den C-877 aus und wieder ein, oder starten Sie den C-877 mit dem Befehl RBT oder mit den entsprechenden Funktionen der PC-Software neu.	
Ungeeignetes Positioniererkabel verwendet	Bei der Verwendung ungeeigneter Kabel können Störungen in der Signalübertragung zwischen Positionierer und C-877 auftreten.  ➤ Wenn Positionierer, Kabel und C-877 als zusammengehörendes System gekennzeichnet sind, tauschen Sie die Systemkomponenten nur nach Rücksprache mit PI gegen andere Komponenten aus.  ➤ Wenn Sie Verlängerungskabel benötigen, kontaktieren Sie unseren Kundendienst (S. 195).	
Positionierer oder Positioniererkabel defekt	Wenn vorhanden, tauschen Sie den defekten Positionierer gegen einen anderen Positionierer aus und testen Sie die neue Kombination.	
Falsche Konfiguration	Prüfen Sie die Parametereinstellungen des C-877 mit den Befehlen SPA? (flüchtiger Speicher) und SEP? (permanenter Speicher). Details zu Parametereinstellungen siehe "Anpassen von Einstellungen" (S. 167).	
Falscher Befehl oder falsche Syntax	Senden Sie den Befehl ERR? und prüfen Sie den zurückgemeldeten Fehlercode.	
Falsche Achse kommandiert	Stellen Sie sicher, dass die richtige Achsenkennung verwendet wird und dass die kommandierte Achse zum richtigen Positionierer gehört.	

Störung: Positionierer führt unbeabsichtigte Bewegung aus		
Mögliche Ursachen	Maßnahmen zur Behebung	
Startup-Makro wird ausgeführt	Prüfen Sie, ob ein Makro als Startup-Makro festgelegt ist, und heben Sie gegebenenfalls die Auswahl des Startup- Makros auf (S. 69).	



Störung: Positionierer schwingt oder wird ungenau positioniert		
Mögliche Ursachen	Maßnahmen zur Behebung	
Die Last wurde geändert.	Stellen Sie das System der Laständerung entsprechend neu ein (S. 56).	

Störung: Positionierer schwingt bereits während der Referenzfahrt		
Mögliche Ursachen	Maßnahmen zur Behebung	
Sehr hohe Last auf dem Positionierer	Gehen Sie bei einer sehr hohen Last während der Referenzfahrt mit PIMikroMove® wie folgt vor:	
	<ol> <li>Starten Sie die Referenzfahrt nicht im Schritt Start up axes, sondern klicken Sie auf Close, um das Fenster Start up controller zu schließen.</li> </ol>	
	<ol> <li>Öffnen Sie im Hauptfenster das Einzelachsen-Fenster für den angeschlossenen Positionierer, indem Sie den Positionierer im Menü View &gt; Single Axis Window auswählen.</li> </ol>	
	3. Erweitern Sie die Ansicht des Einzelachsen-Fensters durch Anklicken der Schaltfläche > am rechten Fensterrand.	
	4. Stellen Sie mit dem Kontrollkästchen <b>Servo</b> sicher, dass der Servomodus eingeschaltet ist.	
	5. Starten Sie die Referenzfahrt durch Klicken auf eine der Schaltflächen <i>Reference</i> .	
	6. Wenn der Positionierer schwingt: Halten Sie die Referenzfahrt im Dialog <b>Reference Axes</b> 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. 56).	
	8. Starten Sie die Referenzfahrt erneut.	
	9. Wenn der Positionierer immer noch schwingt, wiederholen Sie die Schritte 6 bis 8 so lange, bis die Referenzfahrt ohne Schwingungen erfolgreich beendet wird.	



Störung: Kommunikation zwischen Controller und PC funktioniert nicht		
Mögliche Ursachen	Maßnahmen zur Behebung	
Falsches Kommunikationskabel wird verwendet oder es ist defekt	Prüfen Sie, ob das USB-Kabel an einem fehlerfreien System funktioniert.	
Ein anderes Programm greift auf die Schnittstelle zu.	Schließen Sie das andere Programm.	
Probleme mit spezieller Software	<ul> <li>Prüfen Sie, ob das System mit einer anderen Software, wie z. B. einem Terminal-Programm oder einer Entwicklungsumgebung, funktioniert.</li> <li>Sie können die Kommunikation testen, indem Sie ein Terminal-Programm (z. B. PITerminal) starten und *IDN? oder HLP? eingeben.</li> <li>Achten Sie darauf, dass Sie Befehle mit einem LF (line feed) abschließen.</li> <li>Ein Befehl wird erst ausgeführt, wenn der LF empfangen wurde.</li> </ul>	

Störung: Kunden-Software funktioniert nicht mit den PI-Treibern		
Mögliche Ursachen	Maßnahmen zur Behebung	
Falsche Kombination der Treiberroutinen/VIs	<ul> <li>Prüfen Sie, ob das System mit einem Terminal-Programm (z. B. PITerminal) funktioniert.</li> </ul>	
	<ul> <li>Wenn ja:</li> <li>Lesen Sie die Angaben im zugehörigen Softwarehandbuch und vergleichen Sie Ihren Programmcode mit dem</li> </ul>	
	Beispielcode auf dem Datenträger mit der PI Software Suite.	

Wenn die Störung Ihres Systems nicht in der Liste angeführt ist oder wenn sie nicht wie beschrieben behoben werden kann, kontaktieren Sie unseren Kundendienst (S. 195).



### 12 Kundendienst

Wenden Sie sich bei Fragen und Bestellungen an Ihre PI Vertretung oder schreiben Sie uns eine E-Mail (mailto:service@pi.de).

- Geben Sie bei Fragen zu Ihrem System folgende Systeminformationen an:
  - Produkt- und Seriennummern von allen Produkten im System
  - Firmwareversion des Controllers (sofern vorhanden)
  - Version des Treibers oder der Software (sofern vorhanden)
  - PC-Betriebssystem (sofern vorhanden)
- Wenn möglich: Fertigen Sie Fotografien oder Videoaufnahmen Ihres Systems an, die Sie unserem Kundendienst auf Anfrage senden können.

Die aktuellen Versionen der Benutzerhandbücher stehen auf unserer Website zum Herunterladen (S. 4) bereit.



## 13 Technische Daten

Änderungen vorbehalten. Die aktuellen Produktspezifikationen finden Sie auf der Seite des Produkts unter www.pi.de (https://www.physikinstrumente.de/de/).

## 13.1 Spezifikationen

#### 13.1.1 Datentabelle

Bewegung und Regler	C-877.1U11
Achsen	1
Servozykluszeit	100 μs
Profilgenerator	Trapezförmiges Geschwindigkeitsprofil, Punkt-zu-Punkt- Bewegung
Encodereingang	A/B-Quadratur TTL-Pegel, differenziell gem. RS-422
Blockiererkennung	Automatischer Motorstopp bei Überschreitung eines programmierbaren Positionsfehlers
Endschalter	2 × TTL (Polarität programmierbar)
Referenzschalter	1×TTL

Elektrische Eigenschaften	C-877.1U11
Max. Ausgangsleistung pro Kanal	15 W
Max. Ausgangsspannung pro Kanal	200 V <sub>pp</sub> , 71 V <sub>eff</sub>

Schnittstellen und Bedienung	C-877.1U11
Kommunikationsschnittstellen	USB
Motoranschluss	D-Sub 15-pol. (w)
Befehlssatz	PI General Command Set (GCS)
Bedienersoftware	PIMikroMove®, PITerminal
Softwaretreiber	NI LabVIEW-Treiber, dynamische Bibliotheken für Windows und Linux
Unterstützte Funktionen	Startup-Makro; Datenrekorder zur Aufnahme von Betriebsgrößen wie Motorspannung, Geschwindigkeit, Position oder Positionsfehler; interne Sicherheitsschaltung: Watchdog Timer; ID-Chip
Manuelle Bedienhilfe	-

C-877 PILine® Controller MS232D Version: 2.0.0 197



Umgebung	C-877.1U11
Betriebsspannung	24 VDC von externem Netzteil (im Lieferumfang)
Max. Stromaufnahme	300 mA plus Motorstrom (max. 0,8 A)
Betriebstemperaturbereich	5 bis 40 °C
Masse	0,13 kg
Abmessungen	95 mm × 71 mm × 24 mm (inkl. Montageschienen)

### 13.1.2 Bemessungsdaten

Der C-877 ist für folgende Betriebsgrößen ausgelegt:

Eingang an:	Maximale Betriebsspannung	Betriebsfrequenz	Maximale Stromaufnahme
			<u>^</u>
Hohlstecker-Buchse (m)	30 V	<u> </u>	0,8 A

Aus	gang an:	Maximale Piezospannung	Maximale Frequenz der Piezospannung	Maximaler Ausgangsstrom
D-S	ub 15 (f)	200 V <sub>pp</sub>	500 kHz	280 mA <sub>pp</sub>

### 13.1.3 Umgebungsbedingungen und Klassifizierungen

Folgende Umgebungsbedingungen und Klassifizierungen sind für den C-877 zu beachten:

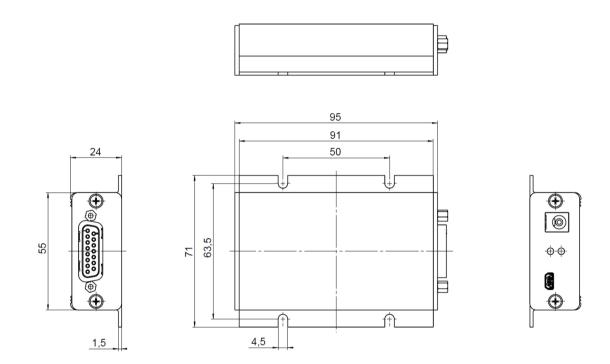
Einsatzbereich	Nur zur Verwendung in Innenräumen
Maximale Höhe	2000 m
Luftdruck	1100 hPa bis 0.1 hPa (entspricht etwa 825 torr bis 0.075 torr)
Relative Luftfeuchte	Höchste relative Luftfeuchte 80 % für Temperaturen bis 31 °C Linear abnehmend bis 50 % relativer Luftfeuchte bei 40 °C
Lagertemperatur	0 °C bis 70 °C
Transporttemperatur	−25 °C bis +85 °C
Überspannungskategorie	II



Schutzklasse	I
Verschmutzungsgrad	2
Messkategorie	
Schutzart gemäß IEC 60529	IP20

## 13.2 Abmessungen

Abmessungen in mm.





## 13.3 Pinbelegung

### 13.3.1 Motor und Sensor

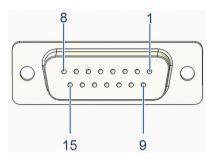
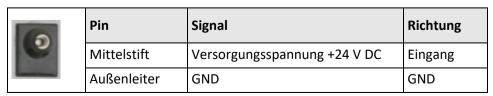


Abbildung 15: Buchse D-Sub 15

Pin		Signal	Funktion
1		NC	nicht angeschlossen
	9	MOTOR GND	Ausgang: Piezo
2		MOTOR GND	Ausgang: Piezo
	10	GND	0 V
3		MOTOR OUT 1	Ausgang: Piezo
	11	MOTOR OUT 1	Ausgang: Piezo
4		VDD	Ausgang: +5 V
	12	NLIMIT	Eingang: negativer Endschalter, TTL
5		PLIMIT	Eingang: positiver Endschalter, TTL
	13	REFSWITCH	Eingang: Referenzschalter, TTL
6		ID CHIP	Bidirektional: ID-Chip
	14	ENCA+	Eingang: Encoderkanal A, RS-422
7		ENCA-	Eingang: Encoderkanal A (invertiert), RS-422
	15	ENCB+	Eingang: Encoderkanal B, RS-422
8		ENCB-	Eingang: Encoderkanal B (invertiert), RS-422

### 13.3.2 Netzteilanschlus

### Hohlstecker-Buchse (m)





## 14 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) SE & Co. KG kostenfrei die umweltgerechte Entsorgung eines Altgeräts von PI, 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) SE & Co. KG Auf der Römerstr. 1 D-76228 Karlsruhe

