MS244D

C-887 Hexapod-Controller

Benutzerhandbuch

Version: 1.5.0 Datum: 29.07.2024

Dieses Dokument beschreibt die folgenden 6-Achs-Controller für Hexapoden:

- **C-887.52**
- **C-887.521**
- **C-887.522**
- **C-887.523**
- **C-887.53**
- **C-887.531**
- **C-887.532**
- **C-887.533**

Physik Instrumente (PI) GmbH & 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

Die folgenden aufgeführten Firmennamen oder Marken sind eingetragene Warenzeichen der Physik Instrumente (PI) GmbH & Co. KG:

PI®, NanoCube®, PICMA®, PIFOC®, PILine®, NEXLINE®, PiezoWalk®, PicoCube®, PiezoMove®, PIMikroMove®, NEXACT®, Picoactuator®, PInano®, NEXSHIFT®, PITOUCH®, PIMag®, PIHera, Q-Motion®

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) GmbH & Co. KG und können Drittanbieter-Softwarekomponenten beinhalten und/oder verwenden. Weitere Informationen finden Sie in den Allgemeinen Softwarelizenzbestimmungen (https://www.physikinstrumente.com/download/EULA_PhysikInstrumenteGmbH_Co_KG.pdf) und in den Drittanbieter-Softwarehinweisen (https://www.physikinstrumente.com/download/TPSWNote_PhysikInstrumenteGmbH_Co_KG.pdf) auf unserer Webseite.

© 2024 Physik Instrumente (PI) GmbH & Co. KG, Karlsruhe, Deutschland. Die Texte, Bilder und Zeichnungen dieses Handbuchs sind urheberrechtlich geschützt. Physik Instrumente (PI) GmbH & Co. KG behält insoweit sämtliche Rechte vor. Die Verwendung dieser Texte, Bilder und Zeichnungen ist nur auszugsweise und nur unter Angabe der Quelle erlaubt.

Originalbetriebsanleitung
Erstdruck: 29.07.2024
Dokumentnummer: MS244D, BRo, Version 1.5.0

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

Inhalt

1 Über dieses Dokument 1

1.1 Ziel und Zielgruppe dieses Benutzerhandbuchs ... 1
1.2 Symbole und Kennzeichnungen ... 1
1.3 Abbildungen ... 2
1.4 Begriffserklärung ... 2
1.5 Mitgeltende Dokumente ... 6
1.6 Handbücher herunterladen ... 9

2 Sicherheit 11

2.1 Bestimmungsgemäße Verwendung ... 11
2.2 Allgemeine Sicherheitshinweise ... 11
2.2.1 Organisatorische Maßnahmen ... 12
2.2.2 Maßnahmen bei der Installation ... 12
2.2.3 Maßnahmen bei Inbetriebnahme und Betrieb ... 13
2.2.4 Maßnahmen bei der Wartung ... 15

3 Produktbeschreibung 17

3.1 Modellübersicht ... 17
3.2 Produktansicht ... 18
3.2.1 Vorderwand ... 18
3.2.2 Typenschild ... 22
3.2.3 Erdungsanschluss ... 23
3.3 Lieferumfang ... 23
3.4 Optionales Zubehör ... 24
3.4.1 Übersicht ... 24
3.4.2 Hexapod-Kabel ... 26
3.5 Kommandierbare Elemente ... 27
3.6 Wichtige Komponenten der Firmware ... 30
3.7 ID-Chip-Erkennung ... 31
3.8 Betriebsparameter der Achsen A und B ... 32
3.9 Bewegungen des Hexapods ... 34
3.9.1 Einführung ... 34
3.9.2 Stellweg und Verfahrbereichsgrenzen ... 34
3.9.3 Unterstützte Bewegungsarten ... 35
3.9.4 Profilgenerator für Punkt-zu-Punkt-Bewegungen ... 37
3.9.5 Zyklische Übertragung von Zielpositionen ... 39
3.9.6 Koordinatensysteme ... 41
3.9.7 Rotationen ... 43

3.9.8 Beispiel für die Definition eines Betriebs-Koordinatensystems mit dem Befehl KSD ... 44
3.9.9 Bewegungsstatus, Einschwingfenster, Einschwingzeit ... 46

3.10 Kommandierung über EtherCAT-Schnittstelle ... 47
3.10.1 Einführung ... 47
3.10.2 Konfiguration des C-887 für Kommandierung durch EtherCAT-Master ... 48
3.10.3 Konfiguration des EtherCAT-Masters ... 49

3.11 Kommunikationsschnittstellen ... 50
3.12 PC-Softwareübersicht ... 51
3.12.1 Ansteuerung von PI Systemen ... 51
3.12.2 PI Software Suite ... 52

4 Auspacken 55

5 Installation 57

5.1 Allgemeine Hinweise zur Installation ... 57
5.2 PC-Software installieren ... 58
5.2.1 Erstinstallation ausführen ... 58
5.2.2 Updates installieren ... 59
5.2.3 Kundenspezifische Positioniererdatenbank installieren ... 60

5.3 Belüftung sicherstellen ... 61
5.4 C-887 erden ... 61
5.5 C-887 an die Stromversorgung anschließen ... 61
5.6 Hexapod installieren ... 62
5.6.1 Zulässige Posen ermitteln ... 62
5.6.2 Hexapod erden ... 63
5.6.3 Hexapod auf Unterlage befestigen ... 63
5.6.4 Last auf Hexapod befestigen ... 63

5.7 Hexapod über Kabelsatz an C-887 anschließen ... 63
5.8 Positionierer für Achsen A und B anschließen ... 64
5.9 Analoge Signalquellen anschließen ... 65
5.10 Digitale Ein- und Ausgänge anschließen ... 66
5.11 PC anschließen ... 67
5.11.1 C-887 über die TCP/IP-Schnittstelle anschließen ... 67
5.11.2 C-887 über die RS-232-Schnittstelle anschließen ... 67

5.12 EtherCAT-Master anschließen ... 68

6 Inbetriebnahme 69

6.1 Allgemeine Hinweise zur Inbetriebnahme ... 69
6.2 C-887 einschalten ... 74
6.3 Kommunikation über TCP/IP-Schnittstelle herstellen ... 74
6.3.1 PC und C-887 für Verwendung statischer IP-Adressen vorbereiten ... 75
6.3.2 Kommunikation über TCP/IP in der PC-Software herstellen ... 79

6.4 Kommunikation über RS-232-Schnittstelle herstellen ... 82
6.4.1 Baudrate ändern ... 82
6.4.2 Kommunikation über RS-232 in der PC-Software herstellen ... 83
6.5 Bewegungen starten ... 84

7 Betrieb 89

7.1 Allgemeine Hinweise zum Betrieb ... 89
7.2 Schutzfunktionen des C-887 ... 90
7.2.1 Automatisches Ausschalten des Servomodus / Stoppen der Bewegung ... 91
7.2.2 Buchse E-Stop verwenden ... 93
7.2.3 Sicherheitsabschaltung konfigurieren ... 97
7.3 Datenrekorder ... 99
7.3.1 Eigenschaften des Datenrekorders ... 99
7.3.2 Datenrekorder einrichten ... 99
7.3.3 Aufzeichnung starten ... 101
7.3.4 Aufgezeichnete Daten auslesen ... 101
7.4 Funktionsgenerator ... 102
7.4.1 Funktionsweise des Funktionsgenerators ... 102
7.4.2 Befehle und Parameter für den Funktionsgenerator ... 103
7.4.3 Kurvenform definieren ... 105
7.4.4 Funktionsgenerator konfigurieren ... 113
7.4.5 Ausgabe starten und stoppen ... 115
7.4.6 Anwendungstipps: Kundenspezifische Daten laden ... 119
7.4.7 Anwendungstipps: Makros für Funktionsgenerator verwenden ... 124
7.5 Controllermakros ... 127
7.5.1 Übersicht: Makrofunktionalitäten und Beispielmakros ... 127
7.5.2 Befehle und Parameter für Makros ... 127
7.5.3 Mit Makros arbeiten ... 129
7.5.4 Variablen ... 135

8 GCS-Befehle 139

8.1 Schreibweise ... 139
8.2 GCS-Syntax für Syntaxversion 2.0 ... 139
8.3 Befehlsübersicht ... 142
8.4 Befehlsbeschreibungen für GCS 2.0 ... 152
8.5 Fehlercodes ... 299

9 Anpassen von Einstellungen 317

9.1 Überblick über die Einstellungen des C-887 ... 317
9.2 Parameterwerte im C-887 ändern ... 317
9.3 Parameterwerte in Textdatei sichern ... 319
9.4 Parameterübersicht ... 320

10 Wartung 339

10.1 C-887 reinigen ... 339
10.2 Firmware und Konfigurationsdateien aktualisieren ... 340
10.2.1 Allgemeine Hinweise zur Aktualisierung von Firmware und Konfigurationsdateien ... 340
10.2.2 Aktuelle Firmware und Konfigurationsdateien beziehen ... 341
10.2.3 Firmware aktualisieren ... 342
10.2.4 Konfigurationsdateien aktualisieren ... 346
10.3 Hexapod warten und prüfen ... 349
10.3.1 Wartungsfahrt durchführen ... 349
10.3.2 Beintest durchführen ... 350

11 Störungsbehebung 357

12 Kundendienst 367

13 Technische Daten 369

13.1 Spezifikationen ... 369
13.1.1 Datentabelle ... 369
13.1.2 Spezifikationen der analogen Eingänge ... 376
13.1.3 Zykluszeiten ... 376
13.1.4 Bemessungsdaten ... 376
13.1.5 Umgebungsbedingungen und Klassifizierungen ... 377
13.2 Abmessungen ... 378
13.3 Spezifikationen Kabel ... 379
13.3.1 Datenübertragungs- und Stromversorgungskabel ... 379
13.4 Pinbelegung ... 380
13.4.1 Anschluss zur Stromversorgung ... 380
13.4.2 Versorgungsspannung für Hexapod ... 381
13.4.3 E-Stop ... 381
13.4.4 IO-Anschluss ... 381
13.4.5 Hexapod ... 383
13.4.6 Motor A, Motor B ... 385
13.4.7 RS-232 ... 385
13.5 Statusregister ... 386
13.5.1 Statusregister für Hexapod-Beine und Achsen A und B ... 386
13.5.2 Systemstatusregister ... 386

14 Altgerät entsorgen 389

15 Europäische Konformitätserklärungen 391

1 Über dieses Dokument

In diesem Kapitel

Ziel und Zielgruppe dieses Benutzerhandbuchs 1
Symbole und Kennzeichnungen 1
Abbildungen 2
Begriffserklärung 2
Mitgeltende Dokumente 6
Handbücher herunterladen 9

1.1 Ziel und Zielgruppe dieses Benutzerhandbuchs

Dieses Benutzerhandbuch enthält die erforderlichen Informationen für die bestimmungsgemäße Verwendung des C-887.

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. 9) bereit.

1.2 Symbole und Kennzeichnungen

In diesem Benutzerhandbuch werden folgende Symbole und Kennzeichnungen verwendet:

VORSICHT

Gefährliche Situation
Bei Nichtbeachtung drohen leichte Verletzungen.

➢ Maßnahmen, um die Gefahr zu vermeiden.

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)
⚠	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 ***Start*** und ***Einstellungen*** 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)
5	Wert, der über die PC-Software eingegeben bzw. ausgewählt werden muss

1.3 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.

1.4 Begriffserklärung

Absolut messender Positionssensor	Sensor (Encoder) zur Erfassung von Lageänderungen oder Winkeländerungen. Die Signale des absolut messenden Positionssensors werden für die Rückmeldung der Achsenposition verwendet. Nach dem Einschalten des Controllers können sofort absolute Zielpositionen kommandiert und erreicht werden. Eine Referenzwertbestimmung ist nicht erforderlich.

Achse

Auch als "logische Achse" bezeichnet. Logische Achsen bilden die Translationen und Rotationen der Bewegungsplattform des Hexapods und die Bewegungen der optional verwendbaren Positionierer in der Firmware des C-887 ab. Jede Bewegungsrichtung entspricht einer logischen Achse. Weitere Informationen zu Translationen und Rotationen finden Sie im Handbuch des Hexapods.

Alle Bewegungsbefehle des C-887 beziehen sich auf logische Achsen.

Achsorientierungs-Koordinatensystem

Mit dem Achsorientierungs-Koordinatensystem kann die Richtung der Translationsachsen X und/oder Y und/oder Z dauerhaft geändert werden (z. B. wenn Z immer in die Richtung der werkseitigen X-Achse zeigen soll). In der Werkseinstellung ist das Achsorientierungs-Koordinatensystem PI_Base aktiv.

Arbeitsraum

Die Gesamtheit aller Posen, die der Hexapod von der aktuellen Position aus anfahren kann, wird als Arbeitsraum bezeichnet.

Der Arbeitsraum kann durch folgende externe Faktoren eingeschränkt sein:

- Vorhandener Einbauraum
- Abmessungen und Position der Last

Um eine unzulässige mechanische Belastung des Hexapods zu vermeiden, müssen zusätzlich zum Arbeitsraum auch die Kräfte betrachtet werden, die an den Posen auf die Hexapod-Beine einwirken. Zulässige Posen können mit dem Simulationsprogramm PIVirtualMove berechnet werden.

Betriebs-Koordinatensystem

Mit dem Betriebs-Koordinatensystem werden die Positionsanzeige, die Bewegungsrichtung und der Drehpunkt für die Bewegungsplattform des Hexapods an die Anwendung angepasst. Es ist auch möglich, --> Work- und Tool-Koordinatensysteme zu verwenden. In der Werkseinstellung ist das Betriebs-Koordinatensystem ZERO aktiv.

Im Handbuch des Simulationsprogramms PIVirtualMove (SM163E) werden die Betriebs-Koordinatensysteme als "Referenzkoordinatensysteme" bezeichnet.

Drehpunkt

Der Drehpunkt beschreibt das Rotationszentrum (Schnittpunkt der Rotationsachsen U, V und W).

Der Drehpunkt bewegt sich immer zusammen mit der Plattform.

In Abhängigkeit vom aktiven --> Betriebs-Koordinatensystem kann der Drehpunkt mit dem Befehl SPI aus dem Ursprung des Koordinatensystems heraus in X- und/oder Y- und/oder Z-Richtung verschoben werden. Der mit dem Befehl SPI verschiebbare Drehpunkt wird auch als "Pivotpunkt" bezeichnet.

Dynamikprofil	Umfasst die für jeden Zeitpunkt der Bewegung berechnete Zielposition, Geschwindigkeit und Beschleunigung der Achse. Die errechneten Werte werden "kommandierte Werte" genannt. Das Dynamikprofil kann vom Profilgenerator des C-887 oder von den Funktionsgeneratoren erzeugt werden, oder es kann extern erzeugt und durch zyklische Übertragung von Zielpositionen an den C-887 übergeben werden.
Fast-Alignment-Routine	Der C-887 unterstützt Routinen für die schnelle Ausrichtung eines Senders oder Empfängers. Ziel der Routinen ist es, Sender und Empfänger so auszurichten, dass auf der Empfängerseite die maximale Intensität des ausgesendeten Signals gemessen wird. Zu den Fast-Alignment-Routinen gehören alle Routinen, die mit den Befehlen FDR und FDG definiert und mit dem Befehl FRS gestartet werden.
Firmware	Software, die auf dem Controller installiert ist.
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.
GCS	PI General Command Set: Befehlssatz für Controller von PI
Hexapod-Bein	Für die Bewegung einer logischen Achse der Bewegungsplattform ist die Bewegung mehrerer Hexapod-Beine notwendig. Der C-887 berechnet aus den vorgegebenen Zielpositionen für die Translations- und Rotationsachsen die Zielpositionen für die einzelnen Beine. Die Geschwindigkeiten und Beschleunigungen der Beine werden so berechnet, dass alle Beine zur selben Zeit starten und stoppen.
Hexapod-System	Die Kombination aus Hexapod, Controller, Kabeln und Netzteil(en) wird in diesem Handbuch als "Hexapod-System" bezeichnet.
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.

Koordinatensystem	Positionsanzeige, Bewegungsrichtung und Drehpunkt für die Bewegungsplattform des Hexapods werden durch miteinander verkettete Koordinatensysteme bestimmt. Die Kette ist grundsätzlich wie folgt aufgebaut (Ausgangspunkt > Endpunkt): --> Koordinatensystem HEXAPOD, --> Korrektur-Koordinatensystem, --> Achsorientierungs-Koordinatensystem, --> Betriebs-Koordinatensystem. Die Koordinatensysteme sind immer rechtshändige Systeme. Das Koordinatensystem HEXAPOD bestimmt die grundsätzlichen Eigenschaften aller anderen Koordinatensysteme. HEXAPOD basiert auf der Konfigurationsdatei mit den Geometriedaten des Hexapods. Die Maßzeichnung im Handbuch des Hexapods zeigt jeweils die Lage des Koordinatensystems HEXAPOD. Mit dem Controller können eigene Koordinatensysteme definiert und anstelle der werkseitig eingestellten Koordinatensysteme verwendet werden. Achsorientierungs- und Korrektur-Koordinatensystem passen, basierend auf HEXAPOD, grundlegende Eigenschaften des aktiven Betriebs-Koordinatensystems an und müssen in den meisten Anwendungen gar nicht oder nur einmalig anwenderspezifisch definiert und aktiviert werden.
Korrektur-Koordinatensystem	Mit dem Korrektur-Koordinatensystem können Fehler in der Ausrichtung des Hexapods dauerhaft korrigiert werden (z. B. Einbaufehler). In der Werkseinstellung ist das Korrektur-Koordinatensystem PI_Levelling aktiv.
PC-Software	Software, die auf dem PC installiert wird.
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.
Pivotpunkt	In Abhängigkeit vom aktiven --> Betriebs-Koordinatensystem kann der Drehpunkt mit dem Befehl SPI aus dem Ursprung des Koordinatensystems heraus in X- und/oder Y- und/oder Z-Richtung verschoben werden. Der mit dem Befehl SPI verschiebbare Drehpunkt wird auch als "Pivotpunkt" bezeichnet. Im Simulationsprogramm PIVirtualMove wird "Pivot Point" als Bezeichnung für ein Betriebs-Koordinatensystem verwendet. Weitere Informationen finden Sie im PIVirtualMove-Handbuch (SM163E).

Pose

Die räumliche Lage des Hexapods, d.h. die Kombination seiner Position und Orientierung im dreidimensionalen Raum, wird als "Pose" bezeichnet. Die Pose eines Hexapods wird durch sechs Koordinaten in Bezug auf ein rechtshändiges kartesisches Koordinatensystem definiert:

- Translationsachsen (auch als "lineare Achsen" bezeichnet): X, Y, Z
- Rotationsachsen: U, V, W

Eine Pose ist zulässig, wenn sie mit dem verwendeten Systemaufbau (Einbaulage des Hexapods, zu bewegende Last, externe Kräfte und Momente) und dem verwendeten Koordinatensystem von allen der sechs Achsen X, Y, Z, U, V, W erreicht werden kann **und** die zulässige Belastung der Beine **nicht** überschritten wird.

Zulässige Posen können mit dem Simulationsprogramm PIVirtualMove berechnet werden.

Work- und Tool-Koordinatensysteme

Für die Arbeit mit benutzerdefinierten Koordinatensystemen kann das Work-und-Tool-Konzept eingesetzt werden.

Das Work-und-Tool-Konzept verwendet eine Kombination aus zwei aktiven --> Betriebs-Koordinatensystemen ("Work-Koordinatensystem" und "Tool-Koordinatensystem"). Die Achsen X, Y, Z des Tool-Koordinatensystems sind immer fest mit der Bewegungsplattform des Hexapods verbunden, d.h. das Tool-Koordinatensystem bewegt sich zusammen mit der Plattform. Die Achsen X, Y, Z des Work-Koordinatensystems sind immer raumfest (in Bezug auf den Hexapod), d.h. das Work-Koordinatensystem bewegt sich **nicht** mit, wenn sich die Plattform des Hexapods bewegt.

Die aktuelle Position der Bewegungsplattform des Hexapods ist als Position des Tool-Koordinatensystems im Work-Koordinatensystem zu verstehen.

Der Drehpunkt für Rotationen liegt immer im Ursprung des Tool-Koordinatensystems und bewegt sich deshalb genau wie das Tool-Koordinatensystem zusammen mit der Plattform.

1.5 Mitgeltende Dokumente

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

Die aktuellen Versionen der Benutzerhandbücher stehen auf unserer Website zum Herunterladen (S. 9) bereit.

Ergänzende Dokumentation für den C-887:

Beschreibung	Dokument
Koordinatensysteme für Hexapod-Mikroroboter	C887T0007 Benutzerhandbuch

Beschreibung	Dokument
Bewegungen des Hexapods Position und Orientierung im Raum, Drehpunkt	C887T0021 Technical Note
Schnelle, mehrkanalige Ausrichtung in der Photonik	E712T0016 User Manual
Nur für Modelle C-887.53, .531, .532, .533: EtherCAT Interface Description	C887T0011 User Manual

Dokumentation für die verfügbare PC-Software:

Beschreibung	Dokument
Release News für die PI Software Suite: Bestandteile und ihre Kompatibilität mit PC-Betriebssystemen	C-990.CD1 Release News
PIVirtualMove: Zulässige Posen des Hexapods durch Simulation ermitteln	SM163E User Manual
GCS Array: Datenformatbeschreibung	SM146E Software Manual
GCS Treiberbibliothek zur Verwendung mit NI LabVIEW-Software	SM158E Software Manual
PI GCS 2.0 DLL	SM151E Software Manual
PI MATLAB-Treiber GCS 2.0	SM155D Softwarehandbuch
PIPython: Python-Module für den Zugriff auf Geräte von PI und die Verarbeitung von GCS-Daten	PIPython Hilfedatei (https://pipython.physikinstrumente.com/)
PIPython: Arbeiten mit Controllern, die mit PIPython ausgestattet sind	SM157D Benutzerhandbuch
PIMikroMove®	SM148E Software Manual
PIHexapodEmulator	C887T0001 User Manual
PIStages3Editor	SM156D Benutzerhandbuch
PIFirmwareManager: Updateprogramm für Controllerfirmware	SM164E User Manual
PIUpdateFinder: Updates suchen und herunterladen	A000T0028 Benutzerhandbuch

Grundsätzliche Informationen zu EtherCAT-Netzwerken und zum Antriebsprofil CiA402:

Beschreibung	Dokument
Elektrische Leistungsantriebssysteme mit einstellbarer Drehzahl - Teil 7-201: Generisches Interface und Nutzung von Profilen für Leistungsantriebssysteme (PDS) - Spezifikation von Profil-Typ 1 (IEC 61800-7-201:2015)	DIN EN 61800-7-201:2016
Elektrische Leistungsantriebssysteme mit einstellbarer Drehzahl - Teil 7-301: Generisches Interface und Nutzung von Profilen für Leistungsantriebssysteme (PDS) - Abbildung von Profil-Typ 1 auf Netzwerktechnologien (IEC 61800-7-301:2015)	DIN EN 61800-7-301:2016
EtherCAT Implementation Directive for CiA402 Drive Profile: Directive for using IEC 61800-7-201 within EtherCAT-based servo drives	ETG.6010 D (R) V1.1.0

Dokumentation für ein EtherCAT-Beispiel von PI:

Beschreibung	Dokument
Implementing a C-887 PI Controller in TwinCAT 3.1 for Motion and Activation of new Coordinate Systems	A000T0075 User Manual

Benutzerhandbücher für Hexapod-Mikroroboter, z. B.:

Modellfamilie	Dokument
H-810 Miniatur-Hexapod-Mikroroboter	MS252D
H-811 Miniatur-Hexapod-Mikroroboter	MS235D
H-820 Hexapod-Mikroroboter	MS207D
H-824 Kompakter Hexapod-Mikroroboter	MS200D
H-825 Kompakter Hexapod-Mikroroboter	MS250D
H-840 Hexapod-Mikroroboter	MS201D
H-850 Hexapod-Mikroroboter	MS202D
H-860 Hochdynamischer Hexapod-Mikroroboter	H860T0002

1.6 Handbücher herunterladen

INFORMATION

Wenn ein Handbuch fehlt oder Probleme beim Herunterladen auftreten:

➢ Wenden Sie sich an unseren Kundendienst (S. 367).

Handbücher herunterladen

1. Öffnen Sie die Website **www.pi.de**.
2. Suchen Sie auf der Website nach der Produktnummer (z. B. C-887).
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***.
6. Füllen Sie das Anfrageformular aus und wählen Sie ***ANFRAGE SENDEN***.

   Der Download-Link wird an die eingegebene E-Mail-Adresse gesendet.

2 Sicherheit

In diesem Kapitel

Bestimmungsgemäße Verwendung 11
Allgemeine Sicherheitshinweise 11

2.1 Bestimmungsgemäße Verwendung

Der C-887 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-887 vorgesehen für den geregelten Betrieb eines Hexapod-Mikroroboters von PI, der mit Antrieben mit integrierten Motortreibern ausgestattet ist. Ab der Seriennummer 121017873 unterstützt der C-887 auch Hexapod-Mikroroboter, die zur Datenübertragung das BiSS-Protokoll verwenden.

Der C-887 darf nicht für andere als die in diesem Benutzerhandbuch genannten Zwecke verwendet werden.

Der C-887 darf nur unter Einhaltung der technischen Spezifikationen und Anweisungen in diesem Benutzerhandbuch verwendet werden. Für die Prozessvalidierung ist der Betreiber verantwortlich.

2.2 Allgemeine Sicherheitshinweise

Der C-887 ist nach dem Stand der Technik und den anerkannten sicherheitstechnischen Regeln gebaut. Bei unsachgemäßer Verwendung des C-887 können Benutzer gefährdet werden und/oder Schäden am C-887 entstehen.

- Benutzen Sie den C-887 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-887 verantwortlich.

2.2.1 Organisatorische Maßnahmen

Benutzerhandbuch

- Halten Sie dieses Benutzerhandbuch ständig am C-887 verfügbar.
  Die aktuellen Versionen der Benutzerhandbücher stehen auf unserer Website zum Herunterladen (S. 9) bereit.
- Fügen Sie alle vom Hersteller bereitgestellten Informationen, z. B. Ergänzungen und Technical Notes, zum Benutzerhandbuch hinzu.
- Wenn Sie den C-887 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-887 nur, nachdem Sie dieses Benutzerhandbuch gelesen und verstanden haben.

Personalqualifikation

Nur autorisiertes und entsprechend qualifiziertes Personal darf den C-887 installieren, in Betrieb nehmen, bedienen, warten und reinigen.

2.2.2 Maßnahmen bei der Installation

- Installieren Sie den C-887 in der Nähe der Stromversorgung, damit der Netzstecker schnell und einfach vom Netz getrennt werden kann.
- Verwenden Sie zum Anschließen des C-887 an das Netz das mitgelieferte Netzkabel.
- Wenn das mitgelieferte Netzkabel ersetzt werden muss, verwenden Sie ein ausreichend bemessenes Netzkabel.

Unzulässige mechanische Belastung und Kollisionen zwischen Hexapod, zu bewegender Last und Umgebung können den Hexapod beschädigen.

- Halten Sie den Hexapod nur an der Grundplatte.
- Ermitteln Sie mit dem Simulationsprogramm PIVirtualMove vor der ersten Installation von Hexapod und Last die zulässigen Posen. Für die Definition einer zulässigen Pose siehe "Zulässige Posen ermitteln" (S. 62).
- Wiederholen Sie die Ermittlung der zulässigen Posen, wenn Sie den Hexapod-Typ wechseln und vor jeder Änderung von Einbaulage, zu bewegender Last, externen Kräften und Momenten oder verwendetem Koordinatensystem.
- Wenn Sie ein eigenes Koordinatensystem definieren und anstelle des werkseitig eingestellten Koordinatensystems verwenden: Beachten Sie, dass das

Simulationsprogramm PIVirtualMove bei der Verwendung benutzerdefinierter Koordinatensysteme engere Stellwegsgrenzen berechnet als der Controller. Wenn Sie das Kommandieren von Posen vermeiden wollen, die außerhalb der von PIVirtualMove berechneten Stellwegsgrenzen liegen:

a) Ermitteln Sie mit dem Controller durch Abfragen mit dem Befehl TRA?, ob Posen kommandiert werden können, die außerhalb der von PIVirtualMove berechneten Stellwegsgrenzen liegen.

b) Wenn notwendig, setzen Sie im Controller mit den Befehlen NLM und PLM Verfahrbereichsgrenzen entsprechend den von PIVirtualMove berechneten Stellwegsgrenzen.

- Stellen Sie bei der Installation des Hexapods und der Last sicher, dass der tatsächliche Systemaufbau dem Systemaufbau entspricht, für den Sie im Simulationsprogramm PIVirtualMove die zulässigen Posen ermittelt haben.
- Vermeiden Sie bei der Installation des Hexapods und der Last hohe Kräfte und Momente auf die Bewegungsplattform.
- Sorgen Sie für eine unterbrechungsfreie Stromversorgung, um eine ungewollte Deaktivierung des Hexapod-Systems und daraus resultierende ungewollte Positionsänderungen des Hexapods zu vermeiden.
- Stellen Sie sicher, dass im Arbeitsraum des Hexapods keine Kollisionen zwischen Hexapod, zu bewegender Last und Umgebung möglich sind.

2.2.3 Maßnahmen bei Inbetriebnahme und Betrieb

Zwischen den bewegten Teilen des Hexapods und einem feststehenden Teil oder Hindernis besteht die Gefahr von leichten Verletzungen durch Quetschung.

- Halten Sie Ihre Finger von Bereichen fern, in denen sie von bewegten Teilen erfasst werden können.

Wenn die Kommunikation zwischen C-887 und PC über TCP/IP hergestellt wird, bietet die PC-Software alle im selben Netzwerk vorhandenen Controller zur Auswahl an. Nach Auswahl eines C-887 für die Verbindung werden alle Befehle an diesen Controller geschickt. Bei Auswahl eines falschen Controllers besteht für das Bedien- und Wartungspersonal des angeschlossenen Hexapods die Gefahr von leichten Verletzungen durch Quetschung aufgrund von unerwartet kommandierten Bewegungen.

- Wenn in der PC-Software mehrere C-887 angezeigt werden, vergewissern Sie sich, dass Sie den richtigen C-887 auswählen.

Wenn die Transportsicherung des Hexapods nicht entfernt wurde und eine Bewegung kommandiert wird, können Schäden am Hexapod entstehen.

- Entfernen Sie die Transportsicherung, bevor Sie das Hexapod-System in Betrieb nehmen.

Kollisionen können den Hexapod, die zu bewegende Last und die Umgebung beschädigen.

Generelle Maßnahmen zur Vermeidung von Kollisionen:

- Stellen Sie sicher, dass im Arbeitsraum des Hexapods keine Kollisionen zwischen Hexapod, zu bewegender Last und Umgebung möglich sind.
- Platzieren Sie keine Gegenstände in Bereichen, in denen sie von bewegten Teilen erfasst werden können.
- Kommandieren Sie nur zulässige Posen. Für die Definition einer zulässigen Pose siehe "Zulässige Posen ermitteln" (S. 62).
- Halten Sie bei einer Fehlfunktion des Controllers die Bewegung sofort an.
- Beachten Sie, dass sich der Hexapod während einer Referenzierungsfahrt auf unvorhersehbare Weise bewegt. Es findet **keine** Kollisionsprüfung und -vermeidung statt. Verfahrbereichsgrenzen, die mit den Befehlen NLM und PLM für die Bewegungsplattform des Hexapods gesetzt wurden, werden während der Referenzierungsfahrt ignoriert.

In Abhängigkeit von der Quelle des Dynamikprofils kann sich die Plattform des Hexapods unter bestimmten Bedingungen auf einer undefinierten Bahn bewegen. Dadurch sind Kollisionen zwischen Hexapod, zu bewegender Last und Umgebung möglich.

Wenn das Dynamikprofil durch den Profilgenerator des C-887 festgelegt wird (Standard):

- Vermeiden Sie das Senden neuer Zielpositionen, wenn sich der Hexapod (Achsen X, Y, Z, U, V, W) noch bewegt.
- Wenn neue Zielpositionen gesendet werden müssen, während sich der Hexapod noch bewegt (Achsen X, Y, Z, U, V, W): Setzen Sie mit Bewegungsbefehlen nur Zielpositionen, die von der aktuellen Position maximal um den Wert des Parameters ***Path Control Step Size*** (ID 0x19001504) abweichen.

Wenn das Dynamikprofil durch aufeinander folgende MOV-Befehle festgelegt wird:

- Setzen Sie mit den aufeinanderfolgenden MOV-Befehlen nur Zielpositionen, deren Abstand zueinander maximal so groß wie der Wert des Parameters ***Path Control Step Size*** (ID 0x19001504) ist.

Wenn Scanprozeduren mit den Befehlen AAP, FIO, FLM, FLS, FSA, FSC, FSM ausgeführt werden, bewegt sich die Plattform des Hexapods bei zu großen Werten für Strecken oder Winkel auf einer undefinierten Bahn und kann verkippen. Dadurch sind Kollisionen zwischen Hexapod, zu bewegender Last und Umgebung möglich, und die Scanprozedur kann mit einem unbefriedigenden Ergebnis enden. Maßnahmen zur Vermeidung des Verkippens:

- Wählen Sie für Strecken und Winkel passende Werte. Für die Hexapod-Modelle H-810, H-811 und H-206 sollten 0,2 mm bzw. 0,2 Grad nicht überschritten werden; für andere Hexapod-Modelle sowie bei geänderten Einstellungen für Koordinatensysteme und Pivotpunkt müssen die idealen Werte experimentell ermittelt werden.

- Stellen Sie die Geschwindigkeit für die Bewegungsplattform des Hexapods so gering wie möglich ein (mit dem Befehl VLS).
- Richten Sie die Bewegungsplattform vor der Scanprozedur schon passend aus.
- Verwenden Sie passende Halterungen für die auszurichtenden Ein- und/oder Ausgänge des optischen Elements auf der Bewegungsplattform, so dass die Bewegung während der Scanprozedur nur über kleine Strecken oder Winkel erfolgt.

Während einer mit dem Befehl FSA gestarteten Scanprozedur kann sich der Scanbereich auf maximal das Doppelte des ursprünglichen Bereichs vergrößern. Dadurch sind Kollisionen zwischen Hexapod, zu bewegender Last und Umgebung möglich.

- Stellen Sie sicher, dass sich die Plattform auch außerhalb des ursprünglich vorgegebenen Scanbereichs gefahrlos bewegen kann.

Wenn die tatsächliche Belastung der Bewegungsplattform des Hexapods die auf der Selbsthemmung der Aktoren basierende maximale Haltekraft überschreitet, kann das Ausschalten des Servomodus für die Achsen der Bewegungsplattform des Hexapods ungewollte Positionsänderungen des Hexapods verursachen. Dadurch sind Kollisionen zwischen Hexapod, zu bewegender Last und Umgebung möglich.

- Stellen Sie sicher, dass die tatsächliche Belastung der Bewegungsplattform des Hexapod die auf der Selbsthemmung der Aktoren basierende maximale Haltekraft nicht überschreitet, bevor Sie den Servomodus ausschalten, den C-887 neu starten oder ausschalten.
- Sorgen Sie für eine unterbrechungsfreie Stromversorgung, um eine ungewollte Deaktivierung des Hexapod-Systems und daraus resultierende ungewollte Positionsänderungen des Hexapods zu vermeiden.

Unpassende Parametereinstellungen können zu unsachgemäßem Betrieb oder zur Beschädigung der angeschlossenen Mechanik führen.

- Ändern Sie Parameter nur nach sorgfältiger Überlegung.

2.2.4 Maßnahmen bei der Wartung

Der C-887 enthält elektrostatisch gefährdete Bauteile, die bei Kurzschlüssen oder Überschlägen beschädigt werden können.

- Trennen Sie vor dem Reinigen des Gehäuses den C-887 von der Stromversorgung, indem Sie den Netzstecker ziehen.

Zwischen den bewegten Teilen des Hexapods und einem feststehenden Teil oder Hindernis besteht die Gefahr von leichten Verletzungen durch Quetschung.

- Halten Sie Ihre Finger von Bereichen fern, in denen sie von bewegten Teilen erfasst werden können.

Kollisionen können den Hexapod, die zu bewegende Last und die Umgebung beschädigen.

Während eines Beintests bewegt sich der Hexapod auf unvorhersehbare Weise. Es findet **keine** Kollisionsprüfung und -vermeidung statt, selbst wenn eine Konfiguration zur Kollisionsvermeidung auf dem C-887 abgelegt wurde (z. B. mit der kostenpflichtigen PIVeriMove Hexapod Software zur Kollisionsprüfung). Verfahrbereichsgrenzen, die mit den Befehlen `NLM` (S. 248) und `PLM` (S. 251) für die Bewegungsplattform des Hexapods gesetzt wurden, werden während eines Beintests ignoriert. Dadurch sind Kollisionen zwischen Hexapod, zu bewegender Last und Umgebung möglich.

- Stellen Sie sicher, dass während eines Beintests des Hexapods keine Kollisionen zwischen Hexapod, zu bewegender Last und Umgebung möglich sind.
- Platzieren Sie keine Gegenstände in Bereichen, in denen sie während eines Beintests von bewegten Teilen erfasst werden können.
- Beaufsichtigen Sie den Hexapod während eines Beintests, um bei Störungen schnell eingreifen zu können.

Während eines Beintests kann das Hexapod-Bein auf einen Endschalter fahren. Dadurch wird automatisch der Servomodus für die Achsen der Bewegungsplattform des Hexapods ausgeschaltet. Das Ausschalten des Servomodus kann ungewollte Positionsänderungen des Hexapods verursachen. Dadurch sind Kollisionen zwischen Hexapod, zu bewegender Last und Umgebung möglich.

- Stellen Sie sicher, dass die tatsächliche Belastung der Bewegungsplattform des Hexapods die auf der Selbsthemmung der Aktoren basierende maximale Haltekraft nicht überschreitet, bevor Sie einen Beintest starten.

3 Produktbeschreibung

In diesem Kapitel

Modellübersicht 17
Produktansicht 18
Lieferumfang 23
Optionales Zubehör 24
Kommandierbare Elemente 27
Wichtige Komponenten der Firmware 30
ID-Chip-Erkennung 31
Betriebsparameter der Achsen A und B 32
Bewegungen des Hexapods 34
Kommandierung über EtherCAT-Schnittstelle 47
Kommunikationsschnittstellen 50
PC-Softwareübersicht 51

3.1 Modellübersicht

Der C-887 Hexapod-Controller ist in folgenden Ausführungen erhältlich:

Modell	Bezeichnung
C-887.52	6-Achs-Controller für Hexapoden, TCP/IP, RS-232, Tischgerät, inkl. Ansteuerung von zwei Zusatzachsen
C-887.521	6-Achs-Controller für Hexapoden, TCP/IP, RS-232, Tischgerät, inkl. Ansteuerung von zwei Zusatzachsen, Analogeingänge
C-887.522	6-Achs-Controller für Hexapoden, TCP/IP, RS-232, Tischgerät, inkl. Ansteuerung von zwei Zusatzachsen, Motion Stop
C-887.523	6-Achs-Controller für Hexapoden, TCP/IP, RS-232, Tischgerät, inkl. Ansteuerung von zwei Zusatzachsen, Motion Stop, Analogeingänge
C-887.53	6-Achs-Controller für Hexapoden, TCP/IP, RS-232, Tischgerät, inkl. Ansteuerung von zwei Zusatzachsen, EtherCAT-Schnittstelle
C-887.531	6-Achs-Controller für Hexapoden, TCP/IP, RS-232, Tischgerät, inkl. Ansteuerung von zwei Zusatzachsen, EtherCAT-Schnittstelle, Analogeingänge
C-887.532	6-Achs-Controller für Hexapoden, TCP/IP, RS-232, Tischgerät, inkl. Ansteuerung von zwei Zusatzachsen, EtherCAT-Schnittstelle, Motion Stop
C-887.533	6-Achs-Controller für Hexapoden, TCP/IP, RS-232, Tischgerät, inkl. Ansteuerung von zwei Zusatzachsen, EtherCAT-Schnittstelle, Motion Stop, Analogeingänge

INFORMATION

Ab der Seriennummer 121017873 unterstützt der C-887 auch Hexapod-Mikroroboter, die zur Datenübertragung das BiSS-Protokoll verwenden.

3.2 Produktansicht

3.2.1 Vorderwand

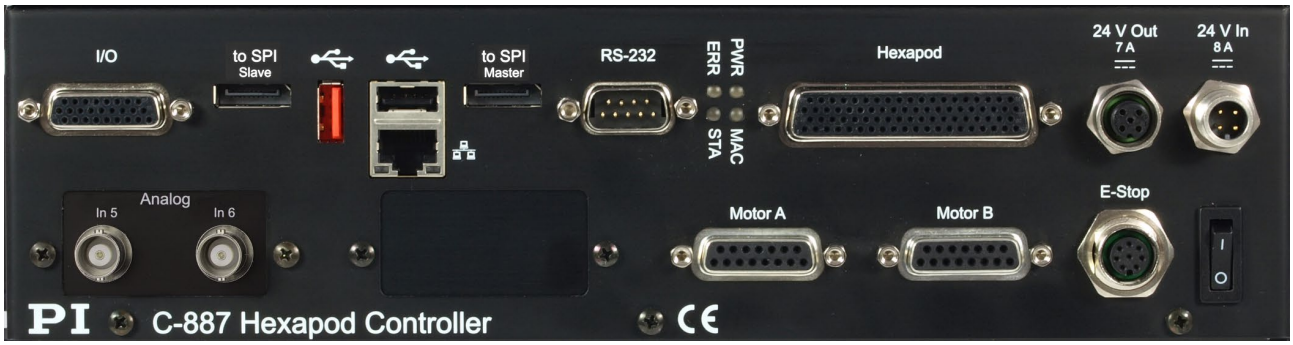


Abbildung 1: C-887.523 Vorderwand; siehe Tabelle für Vorhandensein von Bedienelementen bei anderen Modellen

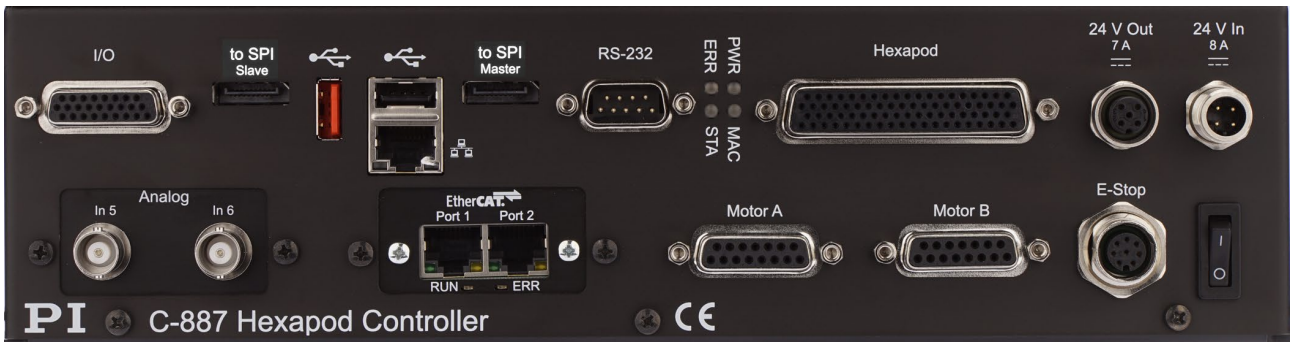
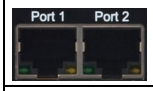


Abbildung 2: C-887.533 Vorderwand; siehe Tabelle für Vorhandensein von Bedienelementen bei anderen Modellen

Beschriftung	Typ	Funktion
I/O	HD D-Sub 26 (w) (S. 381)	Digitale Ein-/Ausgänge: ▪ Ausgänge: Triggern externer Geräte ▪ Eingänge: Verwendung in Makros Analoge Eingänge (multifunktionell)
to SPI Slave	DisplayPort	Anschluss für einen SPI-Slave. Nur für interne Zwecke.
	USB-Typ-A, für hohe Steck- und Ziehkräfte	USB-Schnittstelle zum Anschluss einer manuellen Bedieneinheit C-887.MC2 oder C-887.MC
	USB-Typ-A	USB-Schnittstelle zum Anschluss von Peripheriegeräten
	Buchse RJ45	Netzwerkverbindung über TCP/IP
to SPI Master	DisplayPort	Anschluss für einen SPI-Master. Nur für interne Zwecke.
RS-232	D-Sub 9 (m) (S. 385)	Serielle Verbindung zum PC
ERR	LED rot/aus 	Fehleranzeige: ▪ Dauerhaftes Leuchten: 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.
PWR	LED grün/aus 	Power: ▪ Dauerhaftes Leuchten: Das Booten der Firmware ist abgeschlossen, und der Controller ist bereit für den Normalbetrieb. ▪ Aus: Der Controller ist ausgeschaltet oder die Firmware bootet.

Beschriftung	Typ	Funktion
STA	LED grün/aus 	Status: ▪ Dauerhaftes Leuchten: Das Booten der Firmware ist abgeschlossen, und der Controller ist bereit für den Normalbetrieb. ▪ Aus: Der Controller ist ausgeschaltet oder die Firmware bootet.
MAC	LED grün/rot/aus 	Makro: ▪ Grünes Leuchten: Makro läuft ▪ Rotes Leuchten: Makrofehler Der Fehlercode kann mit dem Befehl `MAC ERR?` abgefragt werden. Durch die Abfrage wird der Fehlercode auf null zurückgesetzt, und die LED wird ausgeschaltet. ▪ Aus: Es läuft kein Makro und es liegt kein Makrofehler vor.
Hexapod	HD D-Sub 78 (w) (S. 383) 	Anschluss zur Datenübertragung zwischen Hexapod und Controller Der C-887 unterstützt ab der Seriennummer 121017873 auch die Datenübertragung über das BiSS-Protokoll.
24 V Out 7 A	Buchse M12 4-polig (w) (S. 381) 	Stromversorgung für Hexapod C-887.522, .523, .532, .533: Der 24-V-Ausgang für den Hexapod muss über die Buchse **E-Stop** aktiviert werden.
24 V In 8 A	Einbaustecker M12 4-polig (m) (S. 380) 	Anschluss für die Versorgungsspannung des C-887 Wenn der Hexapod an der Buchse **24 V Out 7 A** des Controllers angeschlossen ist, wird die Versorgungsspannung des Controllers auch für den Hexapod verwendet.
Analog **In 5** **In 6**	C-887.521, C-887.523, .531, .533: BNC-Buchsen 	Nur C-887.521, .523, .531, .533: Analoge Eingänge, -5 V bis 5 V Details siehe "Spezifikationen der analogen Eingänge" (S. 376).
	C-887.52, C-887.522, .53, .532: Abgedeckt mit Blindblech	

Beschriftung	Typ	Funktion
EtherCAT **Port 1** **Port 2**	C-887.53x**: Buchse RJ45 mit grüner und gelber LED  C-887.52x*: Abgedeckt mit Blindblech	Nur C-887.53x**: Port 1 (links): Anschluss für EtherCAT-Master Port 2 (rechts): Anschluss für den nächsten EtherCAT-Slave Grüne LED: ▪ Dauerhaftes Leuchten: EtherCAT-Verbindung wurde hergestellt ▪ Flackern: EtherCAT-Slave sendet oder empfängt Ethernet-Frames ▪ Aus: Keine EtherCAT-Verbindung Gelbe LED: Keine Verwendung
EtherCAT **RUN**	C-887.53x**: LED grün/aus RUN C-887.52x*: Abgedeckt mit Blindblech	Nur C-887.53x**: Kommunikationsstatus des EtherCAT-Slaves: ▪ Aus: Slave ist im Zustand INIT. ▪ Blinken (2,5 Hz): Slave ist im Zustand PRE-OPERATIONAL (vor dem Betrieb) ▪ Einzelblitz: Slave ist im Zustand SAFE-OPERATIONAL (im sicheren Betrieb) ▪ Dauerhaftes Leuchten: Slave ist im Zustand OPERATIONAL (in Betrieb)
EtherCAT **ERR**	C-887.53x**: LED rot/aus ERR C-887.52x*: Abgedeckt mit Blindblech	Nur C-887.53x**: Kommunikationsstatus des EtherCAT-Slaves: ▪ Aus: Kein Fehler, Slave kommuniziert über EtherCAT ▪ Blinken (2,5 Hz): Ungültige Konfiguration. Allgemeiner Konfigurationsfehler. Mögliche Ursache: Eine durch den Master vorgegebene Zustandsänderung ist aufgrund von Register- oder Objekteinstellungen nicht möglich. ▪ Einzelblitz: Lokaler Fehler. Die Slave-Anwendung hat den EtherCAT-Zustand eigenständig geändert. Mögliche Ursache 1: Ein Host-Watchdog-Timeout ist aufgetreten. Mögliche Ursache 2: Synchronisationsfehler, der Slave wechselt automatisch zu SAFE-OPERATIONAL. ▪ Doppelblitz: Ein Prozessdaten-Watchdog-Timeout ist aufgetreten. Mögliche Ursache: Sync-Manager-Watchdog-Timeout.
Motor A **Motor B**	D-Sub 15 (w) (S. 385) 	Zwei Anschlüsse für Positionierer. Nur für Positionierer mit DC-Motor und integriertem Motortreiber! ▪ Ausgabe der PWM-Signale für den Positionierer ▪ Eingang der Signale des Positionssensors ▪ Eingang der End- und Referenzschaltersignale

Beschriftung	Typ	Funktion
E-Stop	C-887.522, .523, .532, .533: Buchse M12 8-polig (w) (S. 381) C-887.52, .521, .53, .531: Abgedeckt mit Blindstopfen	Nur C-887.522, .523, .532, .533: Anschluss für externe Hardware (z. B. Push-Buttons oder Schalter): Der Anschluss steuert ein internes Relais mit Schließer, das den 24-V-Ausgang für den Hexapod (Buchse **24 V Out 7 A**) deaktiviert oder aktiviert. Wenn keine externe Hardware verwendet wird, muss für die Aktivierung des 24-V-Ausgangs der Kurzschlussstecker C887B0038 angeschlossen werden (im Lieferumfang (S. 23)).
-	Kippschalter 	Ein-/Ausschalter: ▪ Stellung O: Der Controller ist ausgeschaltet ▪ Stellung I : Der Controller ist eingeschaltet Wenn der Hexapod an der Buchse **24 V Out 7 A** des Controllers angeschlossen ist, wird auch der Hexapod ein-/ausgeschaltet. C-887.522, .523, .532, .533: Der 24-V-Ausgang für den Hexapod muss über die Buchse **E-Stop** aktiviert werden.

* C-887.52x steht für C-887.52, .521, .522, .523

** C-887.53x steht für C-887.53, .531, .532, .533

3.2.2 Typenschild

Beschriftung	Funktion
	DataMatrix-Code (Beispiel; enthält die Seriennummer)
C-887.521	Produktbezeichnung (Beispiel), die Stellen nach dem Punkt kennzeichnen das Modell
PI	Herstellerlogo
115040741	Seriennummer (Beispiel), individuell für jeden C-887 Bedeutung der Stellen (Zählung von links): 1 = interne Information, 2 und 3 = Herstellungsjahr, 4 bis 9 = fortlaufende Nummer
Country of origin: Germany	Herkunftsland
⚠	Warnzeichen "Handbuch beachten!"
	Altgeräteentsorgung (S. 389)
CE	Konformitätszeichen CE
WWW.PI.WS	Herstelleradresse (Website)

3.2.3 Erdungsanschluss

Abbildung 3: C-887 Controller, Erdungsanschluss

Beschriftung	Typ	Funktion
⏚	Gewindestift M4	Erdungsanschluss Wenn ein Potentialausgleich erforderlich ist, kann der C-887 mit dem Erdungssystem verbunden werden.

3.3 Lieferumfang

Artikelnummer	Komponenten
C-887	Hexapod-Controller gemäß Ihrer Bestellung
C-815.563	Crossover-Netzwerkkabel
C-815.553	Straight-Through-Netzwerkkabel
C-815.34	Nullmodemkabel für den Anschluss an den PC über RS-232
C-501.24180M12	Separates 24-V-Weitbereichsnetzteil (180 W / 7,5 A) zur Verwendung bei Netzspannungen von 100 bis 240 V AC und Spannungsfrequenzen von 50 oder 60 Hz, mit Stecker M12 4-polig (w)
3763	Netzkabel

Artikelnummer	Komponenten
C-990.CD1	Datenträger mit der PI Software Suite
MS247EK	Kurzanleitung für Hexapod-Systeme
Nur bei den Modellen C-887.522, C-887.523, C-887.532, C-887.533:	
C887B0038	Kurzschlussstecker für die **E-Stop**-Buchse des Controllers

3.4 Optionales Zubehör

3.4.1 Übersicht

Bestell-nummer	Beschreibung
C-887.5xxx	Verschiedene Hexapod-Kabelsätze (S. 26)
C-887.MC2	▪ Manuelle Bedieneinheit für Hexapoden, USB-Anschluss mit 3 m Anschlusskabel, Drehknöpfe für alle kartesischen Achsen, Taster für Bewegungsstopp und Referenzierung, Positionsanzeige ▪ User Manual C887T0036 Für die Verwendung der Bedieneinheit C-887.MC2 muss auf dem Controller mindestens die Version 2.7.1.1 der Firmwarekomponente FW (S. 157) vorhanden sein. Wenn notwendig, aktualisieren Sie die Firmware des Controllers (S. 342). Das Vorgängermodell C-887.MC ist nicht mehr erhältlich, wird aber vom Controller weiterhin unterstützt.

Bestell-nummer	Beschreibung
C-887.MSB	Motion-Stop-Box Für C-887-Modelle mit **E-Stop**-Buchse Details siehe "Buchse E-Stop verwenden" (S. 93).
F-712.PM1	Optischer Leistungsmesser, 400-1550 nm Wellenlängenbereich, bis 1 mA Eingangsstrom, 20 kHz Signalbandbreite, logarithmischer Ausgang ±5 V, Tischgerät, inklusive Netzteil Für C-887-Modelle mit BNC-Buchsen **Analog In 5** und **In 6** Details siehe Handbuch des optischen Leistungsmessers (MP165D).
F-712.IRP1	Hochauflösender optischer Leistungsmesser mit logarithmischem Ausgangssignal, 600 bis 1700 nm Wellenlängenbereich, Eingangsleistung von 230 pW bis 1,3 mW, 6 kHz Signalbandbreite, logarithmische Ausgangsspannung von 0,1 bis 1,6 V, einkanaliges Tischgerät. Für C-887-Modelle mit BNC-Buchsen **Analog In 5** und **In 6** Details siehe Handbuch des optischen Leistungsmessers (MP192D).
R-FMP-GSM	Gradientensuchmodus. Firmwarefunktionalität zur gleichzeitigen Ausführung von Fast-Alignment-Routinen zur Gradientensuche in mehreren Bewegungsachsen. Für alle Hexapod-Systeme mit H-811 Miniatur-Hexapod Mikroroboter ist die Gradientensuche bereits standardmäßig verfügbar. Details zur Gradientensuche siehe Handbuch "Schnelle, mehrkanalige Ausrichtung in der Photonik" (E712T0016).
L-xxx **M-xxx**	Positionierer von PI, die mit DC-Motor und PWM-Verstärker ausgerüstet sind. Informationen zu geeigneten Modellen auf Anfrage.

INFORMATION

Die kostenpflichtige PIVeriMove Hexapod Software zur Kollisionsprüfung (Bestellnummer C-887.VM1) wird mit Release 2.8.0 der PI Software Suite abgekündigt und ist nicht mehr erhältlich, wird aber vom Controller weiterhin unterstützt.

➢ Wenden Sie sich bei Bestellungen und Fragen an den Kundendienst (S. 367).

3.4.2 Hexapod-Kabel

Informationen zur Eignung der Kabel für bestimmte Hexapod-Modelle finden Sie in "Spezifikationen Kabel" (S. 379).

Bestellnummer	Datenübertragungskabel, verfügbare Längen
C-815.82D02	Datenübertragungskabel für Hexapoden, schleppkettentauglich, HD D-Sub 78 m/w, 2 m
C-815.82D03	Datenübertragungskabel für Hexapoden, schleppkettentauglich, HD D-Sub 78 m/w, 3 m
C-815.82D05	Datenübertragungskabel für Hexapoden, schleppkettentauglich, HD D-Sub 78 m/w, 5 m
C-815.82D07	Datenübertragungskabel für Hexapoden, schleppkettentauglich, HD D-Sub 78 m/w, 7,5 m
C-815.82D10	Datenübertragungskabel für Hexapoden, schleppkettentauglich, HD D-Sub 78 m/w, 10 m
C-815.82D20	Datenübertragungskabel für Hexapoden, schleppkettentauglich, HD D-Sub 78 m/w, 20 m

Bestellnummer	Stromversorgungskabel, verfügbare Längen
C-815.82P02A	Stromversorgungskabel für Hexapoden, schleppkettentauglich, M12 m/w abgewinkelt, 2 m
C-815.82P03A	Stromversorgungskabel für Hexapoden, schleppkettentauglich, M12 m/w abgewinkelt, 3 m
C-815.82P05A	Stromversorgungskabel für Hexapoden, schleppkettentauglich, M12 m/w abgewinkelt, 5 m
C-815.82P07A	Stromversorgungskabel für Hexapoden, schleppkettentauglich, M12 m/w abgewinkelt, 7,5 m
C-815.82P10A	Stromversorgungskabel für Hexapoden, schleppkettentauglich, M12 m/w abgewinkelt, 10 m
C-815.82P20A	Stromversorgungskabel für Hexapoden, schleppkettentauglich, M12 m/w abgewinkelt, 20 m

Bestellnummer	Stromversorgungskabel, verfügbare Längen
C-815.82P02E	Stromversorgungskabel für Hexapoden, schleppkettentauglich, M12 m/w gerade, 2 m
C-815.82P03E	Stromversorgungskabel für Hexapoden, schleppkettentauglich, M12 m/w gerade, 3 m
C-815.82P05E	Stromversorgungskabel für Hexapoden, schleppkettentauglich, M12 m/w gerade, 5 m

Bestellnummer	Stromversorgungskabel, verfügbare Längen
C-815.82P07E	Stromversorgungskabel für Hexapoden, schleppkettentauglich, M12 m/w gerade, 7,5 m
C-815.82P10E	Stromversorgungskabel für Hexapoden, schleppkettentauglich, M12 m/w gerade, 10 m
C-815.82P20E	Stromversorgungskabel für Hexapoden, schleppkettentauglich, M12 m/w gerade, 20 m

Wenden Sie sich für Bestellungen an den Kundendienst (S. 367).

3.5 Kommandierbare Elemente

Die folgende Tabelle enthält die mit den Befehlen des GCS (S. 139) kommandierbaren Elemente.

INFORMATION

Die Modelle C-887.53, .531, .532 und .533 sind mit einer EtherCAT-Schnittstelle ausgestattet. Bei Kommandierung über die EtherCAT-Schnittstelle wird das Hexapod-System als Mehrachsgerät entsprechend des Antriebsprofils CiA402 verwendet. Der EtherCAT-Master kommandiert die logischen Achsen X, Y, Z, U, V und W der Bewegungsplattform des Hexapods.

Weitere Informationen siehe "Kommandierung über EtherCAT-Schnittstelle" (S. 47).

Element	Anzahl	Kennung	Beschreibung
Logische Achse	6	X, Y, Z, U, V, W	Die logischen Achsen X bis W bilden die Translationen und Rotationen der Bewegungsplattform des Hexapods in der Firmware des C-887 ab. Translationsachsen: X, Y und Z Rotationsachsen: U, V und W Beachten Sie, dass die Achsen U, V, W in der Dokumentation des Hexapods auch mit θX, θY, θZ oder A (rot X), B (rot Y), C (rot Z) bezeichnet werden.
Logische Achse	2	A, B	Die logischen Achsen A und B bilden die Bewegung zusätzlicher Positionierer in der Firmware des C-887 ab.

<table>
<tr><th>Element</th><th>An-zahl</th><th>Ken-nung</th><th>Beschreibung</th></tr>
<tr><td>Hexapod-Bein</td><td>6</td><td>1 bis 6</td><td>Die Hexapod-Beine sind nicht für Bewegungsbefehle zugänglich, können aber zu Diagnosezwecken als Datenquellen für die Aufzeichnung durch den Datenrekorder ausgewählt werden, siehe auch <code>DRC</code> (S. 168) und <code>HDR?</code> (S. 200).</td></tr>
<tr><td>Eingangs-signalkanal</td><td>Maximal 6</td><td>1 bis 6</td><td>Kanäle der Analogeingänge; die Kennung ist den Eingangssignalkanälen in folgender Reihenfolge zugeordnet:
1 bis 4: analoge Eingänge auf der Buchse I/O (S. 381)
5 und 6: Nur bei den Modellen C-887.521, .523, .531, .533. Analoge Eingänge an den BNC-Buchsen Analog In 5 (Kennung 5) und In 6 (Kennung 6)
Der Befehl <code>TAC?</code> (S. 269) fragt die Anzahl der installierten analogen Eingänge ab.</td></tr>
<tr><td>Digitaler Ausgang</td><td>4</td><td>1 bis 4</td><td>1 bis 4 kennzeichnen die digitalen Ausgangsleitungen 1 bis 4 der Buchse I/O (S. 381).
Der Befehl <code>DIO</code> (S. 166) setzt den Status der digitalen Ausgangsleitungen.</td></tr>
<tr><td>Digitaler Eingang</td><td>4</td><td>1 bis 4</td><td>1 bis 4 kennzeichnen die digitalen Eingangsleitungen 1 bis 4 der Buchse I/O (S. 381).
Der Befehl <code>DIO?</code> (S. 167) fragt den Status der digitalen Eingangsleitungen ab.</td></tr>
<tr><td>Funktionsgenerator</td><td>8</td><td>1 bis 8</td><td>Jeder Funktionsgenerator (S. 102) ist einer logischen Achse fest zugeordnet:
<table>
<tr><th>Mechanik</th><th colspan="6">Hexapod</th></tr>
<tr><td>Achse</td><td>X</td><td>Y</td><td>Z</td><td>U</td><td>V</td><td>W</td></tr>
<tr><td>Funktions-generator</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td></tr>
</table>
<table>
<tr><th>Mechanik</th><th colspan="2">Zusatzachsen</th></tr>
<tr><td>Achse</td><td>A</td><td>B</td></tr>
<tr><td>Funktionsgenerator</td><td>7</td><td>8</td></tr>
</table></td></tr>
</table>

Element	Anzahl	Kennung	Beschreibung
Kurventabelle	100	1 bis 100	Die Kurventabellen enthalten die gespeicherten Daten (insgesamt 10.000.000 Punkte) für die Kurvenformen, die durch die Funktionsgeneratoren ausgegeben werden. Der Wert des Parameters ***Number Of Waves*** (ID 0x1300010A) gibt die Anzahl der Kurventabellen (S. 102) an.
Koordinatensystem	Unbegrenzt	Name wird beim Definieren des Koordinatensystems vergeben	Mit dem Controller können eigene Koordinatensysteme definiert und anstelle der werkseitig eingestellten Koordinatensysteme verwendet werden. Weitere Informationen siehe "Bewegungen des Hexapods" (S. 34). **Konventionen für Namen von Koordinatensystemen:** Zulässige Zeichen: 1234567890ABCDEFGHIJKLMNOPQRSTUVWXYZ_ Die Zeichenanzahl ist unbegrenzt. Der Name muss mit einem Buchstaben beginnen. Reservierte Namen, die nicht zum Definieren, Kopieren oder Löschen verwendet werden dürfen: HEXAPOD, PI_LEVELLING, PI_BASE, ZERO, 0, NULL, XML, KLF, KLF(USER), KLF(PI), KLD, KLD(USER), KLD(PI), KSB, KSB(USER), KSB(PI), KSD, KSF, KST, KSW Jeder Name darf nur einmal vorhanden sein. Ein vorhandenes Koordinatensystem, das nicht verwendet wird, wird beim Anlegen (Definieren, Erzeugen einer Kopie) eines Koordinatensystems mit demselben Namen überschrieben.
Fast-Alignment-Routinen	100	Name wird beim Definieren der Routine vergeben	Zu den Fast-Alignment-Routinen gehören alle Routinen, die mit den Befehlen FDR und FDG definiert werden. **Konventionen für Namen von Fast-Alignment-Routinen:** String bestehend aus alphanumerischen Zeichen. Leerzeichen und Sonderzeichen sind nicht zulässig. Weitere Informationen finden Sie im Dokument "Schnelle, mehrkanalige Ausrichtung in der Photonik" (E712T0016).

Element	Anzahl	Kennung	Beschreibung
Datenrekordertabelle	16	1 bis 16	Die Datenrekordertabellen enthalten die aufgezeichneten Daten. Der C-887 hat 16 Datenrekordertabellen (Abfrage mit `TNR?` (S. 273)) mit maximal 262144 Datenpunkten pro Tabelle (Standard: 8192 Datenpunkte pro Tabelle). Die Anzahl der Punkte pro Datenrekordertabelle kann mit dem Parameter ***Data Recorder Points Per Table*** (ID 0x16000201) eingestellt werden.
Gesamtsystem	1	1	C-887 als Gesamtsystem.

3.6 Wichtige Komponenten der Firmware

Die Firmware des C-887 stellt die folgenden funktionalen Einheiten bereit:

Firmwarekomponente	Beschreibung
ASCII-Befehle	Die Kommunikation mit dem C-887 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: ▪ Bewegungen des Hexapods starten ▪ System- und Bewegungswerte abfragen ▪ Koordinatensystem definieren ▪ Fast-Alignment-Routinen definieren Eine Liste der verfügbaren Befehle finden Sie im Abschnitt "Befehlsübersicht" (S. 142).
Konfigurationsdateien, Positionierer-datenbanken, Parameter	Die Einstellungen, mit denen der C-887 z. B. an die Eigenschaften der angeschlossenen Mechanik oder an die verwendeten Kommunikationsschnittstellen angepasst wird, werden bestimmt durch: ▪ Konfigurationsdateien: siehe "Firmware und Konfigurationsdateien aktualisieren" (S. 340) ▪ Positioniererdatenbanken: siehe "Betriebsparameter der Achsen A und B" (S. 32) ▪ Schnittstellenparameter: siehe "Kommunikation über TCP/IP-Schnittstelle herstellen" (S. 74) und "Kommunikation über die RS-232-Schnittstelle herstellen" (S. 82) ▪ Weitere Parameter siehe "Anpassen von Einstellungen" (S. 317)

Firmwarekomponente	Beschreibung
Befehlsebenen	Die Verfügbarkeit von Befehlen und das Schreibrecht auf die Parameter werden durch Befehlsebenen festgelegt. Die aktuelle Befehlsebene kann mit dem Befehl CCL geändert werden. Dazu kann die Eingabe eines Kennworts erforderlich sein. Weitere Informationen finden Sie im Abschnitt "Anpassen von Einstellungen" (S. 317).
Datenrekorder	Der C-887 besitzt einen Echtzeit-Datenrekorder (S. 99). Er kann verschiedene Ein- und Ausgangssignale (z. B. aktuelle Position, kommandierte Position) von verschiedenen Datenquellen (z. B. logische Achsen, Eingangssignalkanäle) aufzeichnen (S. 99).
Funktionsgenerator	Jede logische Achse kann von einem Funktionsgenerator gesteuert werden, der Kurvenformen ausgibt. Der Funktionsgenerator eignet sich besonders für dynamische Anwendungen, bei denen zum Beispiel periodische Bewegungen der Achse ausgeführt werden (S. 102).
Makros	Der C-887 kann Makros speichern. Über die Makrofunktion können Befehlssequenzen festgelegt und dauerhaft gespeichert werden. Ein Startup-Makro kann festgelegt werden, das bei jedem Einschalten oder Neustart des C-887 ausgeführt wird. Dies vereinfacht den Betrieb ohne Verbindung zum PC. Weitere Informationen finden Sie im Abschnitt "Controllermakros" (S. 127).

Die Firmware kann mit einem Hilfsprogramm aktualisiert werden (S. 340).

3.7 ID-Chip-Erkennung

Der Hexapod enthält einen ID-Chip, auf dem der Hexapod-Typ, die Seriennummer und das Herstelldatum gespeichert sind. Die Daten werden beim Einschalten oder Neustart des Controllers vom ID-Chip geladen. In Abhängigkeit von den geladenen Daten behält der Controller die aktuelle Konfiguration bei oder installiert eine neue Konfiguration.

Für den einfachen Austausch sind die Konfigurationsdaten aller Standard-Hexapoden werkseitig auf jedem Standardcontroller gespeichert (z. B. Geometriedaten und Regelparameter). Die Konfigurationsdaten für kundenspezifische Hexapoden sind nur dann auf dem Controller gespeichert, wenn Hexapod und Controller zusammen ausgeliefert werden, oder wenn PI vor der Auslieferung des Controllers entsprechend informiert wurde.

In Abhängigkeit von den Daten, die vom ID-Chip geladen wurden, verhält sich der Controller nach dem Einschalten oder Neustart wie folgt:

- Wenn der Hexapod-Typ und die Seriennummer, die vom ID-Chip geladen wurden, identisch sind mit den im Controller gespeicherten Daten, behält der Controller die aktuelle Konfiguration bei. Das System ist sofort betriebsbereit.
- Wenn der Hexapod-Typ und/oder die Seriennummer, die vom ID-Chip geladen wurden, von den im Controller gespeicherten Daten abweichen:
  - Wenn der Hexapod-Typ identisch ist, aber die Seriennummer abweicht, installiert der Controller die Standardkonfiguration für diesen Typ.

– Wenn der Hexapod-Typ abweicht, installiert der Controller eine neue Konfiguration, die zum neuen Hexapod-Typ passt.

Zum Aktivieren der installierten Konfiguration kann ein Neustart des Controllers erforderlich sein.

INFORMATION

Anwendungshinweise für die ID-Chip-Erkennung:

- Bevor Sie den angeschlossenen Hexapod austauschen, sichern Sie die aktuellen Parameterwerte des Controllers auf dem PC (S. 319).
- Schließen Sie den Hexapod nur an den ausgeschalteten Controller an.
- Wenn das Booten der Firmware abgeschlossen ist, senden Sie den Befehl `CST?` (S. 164), um zu prüfen, ob die installierte Konfiguration durch einen Neustart des Controllers aktiviert werden muss. Ein Neustart ist erforderlich, wenn die Antwort "NOSTAGE" lautet. Der Controller kann mit dem Befehl `RBT` (S. 253) neu gestartet werden.
- Senden Sie den Befehl `ERR?` (S. 176), um zu prüfen, ob die Konfiguration erfolgreich aktiviert wurde. Wenn die Antwort auf ERR? den Fehlercode 233 oder 211 enthält, fehlt im Controller die Konfigurationsdatei für den neuen Hexapod (möglich z. B. für kundenspezifische Hexapoden oder neue Standard-Hexapoden). Kontaktieren Sie unseren Kundendienst (S. 367), um eine geeignete Konfigurationsdatei zu erhalten. Für die Installation der neuen Konfigurationsdatei siehe "Firmware und Konfigurationsdateien aktualisieren" (S. 340).
- Senden Sie den Befehl `VER?` (S. 278), um die auf dem ID-Chip gespeicherten Angaben für Hexapod-Typ, Seriennummer und Herstelldatum zu prüfen. Beispiel für die Antwort: `IDChip: H-811.F-2 SN123456789 20/1/2016`

3.8 Betriebsparameter der Achsen A und B

Die Buchsen **Motor A** und **Motor B** (Sub-D 15 (f)) des C-887 sind für den Anschluss von Positionierern mit DC-Motor und integrierten Motortreibern vorgesehen.

Wenn Sie PI vor Auslieferung des Hexapod-Systems über die verwendeten Positionierertypen informieren, konfiguriert PI den C-887 entsprechend Ihrer Bestellung, so dass die entsprechenden Positionierertypen zu den Achsen A und B des C-887 zugewiesen sind:

- Wenn Sie nur einen Positionierer bestellen, wird der entsprechende Positionierertyp der Achse A zugewiesen.
- Wenn Sie zwei Positionierer bestellen, werden die entsprechenden Positionierertypen den Achsen A und B in aufsteigender alphabetischer Reihenfolge zugewiesen, z. B. M-403.1DG zu A und M-403.2PD zu B; M-414.1PD zu A und M-511.DD1 zu B.

Die Achsen A und B sind deaktiviert und werden in der PC-Software(z. B. in PIMikroMove®) nicht angezeigt, wenn der Positionierertyp für sie auf NOSTAGE gesetzt ist.

Sie können die Zuweisung des Positionierertyps ändern, indem Sie die PC-Software von PI verwenden (z. B. PIMikroMove®, siehe "Bewegungen starten (S. 84)") oder den Befehl CST (S. 163) senden.

Bei der Zuweisung eines neuen Positionierertyps werden dessen Betriebsparameter aus einer der beiden folgenden Positioniererdatenbanken geladen:

Dateiname	PISTAGES3.DB	PIStages2.dat
Speicherort	PC	C-887
Beschreibung	Enthält im Auslieferungszustand Parametersätze für Standardpositionierer von PI und PI miCos; wird bei der Installation der PC-Software automatisch auf dem PC gespeichert. Neue Parametersätze können angelegt, editiert und gespeichert werden.	Enthält Parametersätze für Standardpositionierer von PI und PI miCos. Wenn Sie PIStages2.dat im C-887 aktualisieren möchten, wenden Sie sich an den Kundendienst (S. 367).
Notwendige Bedingungen für die Verwendung der Datenbank	Alle Bedingungen müssen erfüllt sein: ▪ Der C-887 unterstützt den Parameter 0x3C (ab Firmwareversion 2.5.2.1). ▪ Version 3.17.0 oder höher der dynamischen Programmbibliothek für GCS (PI GCS 2.0 DLL) ist auf dem PC installiert. ▪ Die Funktionen PI_qVST() und PI_CST() der PI GCS 2.0 DLL werden verwendet. Beispiel: PIMikroMove® ruft diese Funktionen auf, wenn Sie im Fenster ***Start up controller*** einen Positionierertyp zuweisen oder im Fenster ***Command entry*** die Befehle CST und VST? senden.	Mindestens eine Bedingung muss erfüllt sein: ▪ Die Version der Firmware des C-887 ist älter als 2.5.2.1. ▪ Die Version der dynamischen Programmbibliothek für GCS (PI GCS 2.0 DLL) ist älter als 3.17.0. ▪ Sie senden die Befehle CST und VST? in einem Terminalprogramm ohne Verwendung der PI GCS 2.0 DLL.

Es wird empfohlen, die Versionen von Firmware und PC-Software so zu aktualisieren, dass die Datenbank PISTAGES3.DB verwendet wird. Die Datenbank PIStages2.dat wird nur noch aus Kompatibilitätsgründen bereitgestellt.

Die in Ihrem Hexapod-System verwendeten Versionen können Sie der Antwort auf den Befehl VER? (S. 278) entnehmen.

Weitere Informationen zu Positioniererdatenbanken finden Sie in den Handbüchern zum PIStages3Editor und zur Programmbibliothek PI GCS 2.0 DLL.

3.9 Bewegungen des Hexapods

3.9.1 Einführung

Der C-887 Hexapod-Controller wird zur Ansteuerung eines Hexapods in sechs Freiheitsgraden mit sehr hoher Positioniergenauigkeit verwendet. Ein Hexapod bietet lineare Bewegungen in Richtung der X-, Y- und Z-Achsen sowie Rotationen um jede dieser drei Achsen.

Der C-887 regelt die Motoren der sechs Hexapod-Beine. Die Hexapod-Beine tragen die Bewegungsplattform und bringen sie in die gewünschte Position.

Positionierungsbefehle verwenden kartesische Koordinaten. Der C-887 rechnet diese um in die jeweiligen Positionen und Geschwindigkeiten der Hexapod-Beine, bevor sich die Plattform in die gewünschte Position bewegt.

3.9.2 Stellweg und Verfahrbereichsgrenzen

Fragen Sie mit TRA? (S. 273) die absolute Position ab, die maximal kommandiert werden kann, wenn sich die Plattform des Hexapods entlang eines vorgegebenen Richtungsvektors bewegt. Die maximal kommandierbare Position wird ausgehend von der aktuellen Position berechnet und kann nur abgefragt werden, wenn sich die Plattform des Hexapods nicht bewegt. Die aktuellen Einstellungen für die Verfahrbereichsgrenzen (siehe NLM (S. 248), PLM (S. 251), SSL (S. 263)) und, falls vom aktiven Betriebs-Koordinatensystem verwendet, für den mit SPI definierten Pivotpunkt (siehe SPI (S. 260)) werden in die Berechnung einbezogen.

INFORMATION

Unvermeidliche Rundungsfehler bei der internen Positionsberechnung mit TRA?

Wenn der mit TRA? vorgegebene Richtungsvektor die Plattform in die Nähe einer Stellwegsgrenze bewegen würde, zeigt die Antwort rundungsbedingt möglicherweise eine Position an, die nicht erreicht werden kann. Das Kommandieren einer solchen Position schlägt fehl und erzeugt den Fehlercode 7 ("Position out of limits"). Sie können deshalb die Antwort auf TRA? mit einem Faktor so begrenzen, dass nur Positionen angezeigt werden, die auch tatsächlich kommandiert werden können:

➢ Setzen Sie mit SPA den Parameter ***Reduction Factor for TRA? Response*** (0x19006000) auf einen geeigneten Wert zwischen 0 und 1.

Wenn für das aktuell verwendete Koordinatensystem zulässig, können Sie die kleinste und die größte kommandierbare Position der einzelnen Achsen mit TMN? (S. 271) und TMX? (S. 272) abfragen. Beachten Sie dabei, dass die Stellwege in X, Y, Z, U, V, W voneinander abhängig sind. Je nach der aktuellen Position der Bewegungsplattform des Hexapods kann der tatsächlich verfügbare Stellweg für die Achsen X, Y, Z, U, V und W geringer ausfallen als in den Antworten auf TMN? und TMX? angegeben. Die Antworten auf TMN? und TMX? entsprechen dem tatsächlich verfügbaren Stellweg einer Achse nur dann, wenn die folgenden Bedingungen erfüllt sind:

- Alle anderen Achsen stehen auf Nullposition.
- Die werkseitig eingestellten Koordinatensysteme sind aktiv.
- Die Standardeinstellungen für die Pivotpunktkoordinaten gelten.

Verwenden Sie VMO? (S. 280), um abzufragen, ob eine Zielposition erreicht werden kann.

Die physikalische Einheit der Position kann mit PUN? (S. 253) abgefragt werden.

INFORMATION

Stellwegsgrenzen in PIVirtualMove
Wenn Sie ein eigenes Koordinatensystem definieren und anstelle des werkseitig eingestellten Koordinatensystems verwenden: Beachten Sie, dass das Simulationsprogramm PIVirtualMove bei der Verwendung benutzerdefinierter Koordinatensysteme engere Stellwegsgrenzen berechnet als der Controller.

3.9.3 Unterstützte Bewegungsarten

Der C-887 unterstützt die folgenden Bewegungsarten:

Bewegungsart	Auslösen der Bewegung
Referenzierungsfahrt (S. 185)	FRF Manuelle Bedieneinheit (C-887.MC2 oder C-887.MC) (S. 24) für die Achsen der Bewegungsplattform des Hexapods (X, Y, Z, U, V, W)
	Nur C-887-Modelle mit EtherCAT-Schnittstelle: Homing-Modus gemäß Antriebsprofil CiA402. Der Parameter ***Configure Command Mode*** (ID 0x19002000) muss den Wert 1 = "External: EtherCAT" haben. Weitere Informationen siehe "Kommandierung über EtherCAT-Schnittstelle" (S. 47).
Punkt-zu-Punkt-Bewegung; Profilgenerator erzeugt das Dynamikprofil (S. 37)	MOV: Bewegung zu absoluter Zielposition Der Parameter ***Trajectory Source*** (ID 0x19001900) muss den Wert 0 haben (Standard).
	STE, IMP: Sprung oder Impuls starten, mit Datenaufzeichnung MVR: Bewegung relativ zur letzten kommandierten Zielposition MRT, MRW: Bewegt die angegebene Achse relativ im Tool- bzw. Work-Koordinatensystem. Siehe "Begriffserklärung" (S. 2) für weiterführende Erläuterungen zu Koordinatensystemen
	Manuelle Bedieneinheit (C-887.MC2 oder C-887.MC) für die Achsen der Bewegungsplattform des Hexapods (X, Y, Z, U, V, W) Die Drehknöpfe der Bedieneinheit starten Bewegungen. Die Knöpfe sind schrittweise drehbar. Eine Drehung um einen Schritt startet eine Bewegung um die Schrittweite, die mit dem Befehl SST eingestellt wurde.

Bewegungsart	Auslösen der Bewegung
Zyklische Übertragung von Zielpositionen (S. 39)	Aufeinander folgende MOV-Befehle Der Parameter ***Trajectory Source*** (ID 0x19001900) muss den Wert 1 = "Dynamikprofil wird durch aufeinander folgende `MOV`-Befehle festgelegt" haben.
	Nur C-887-Modelle mit EtherCAT-Schnittstelle: Cyclic-Synchronous-Position-Modus (CSP) gemäß Antriebsprofil CiA402. Der Parameter ***Configure Command Mode*** (ID 0x19002000) muss den Wert 1 = "External: EtherCAT" haben. Weitere Informationen siehe "Kommandierung über EtherCAT-Schnittstelle" (S. 47).
Funktionsgenerator (S. 102)	WGO: Startet/stoppt die Funktionsgeneratorausgabe
Fast-Alignment-Routinen (S. 2)	Weitere Informationen finden Sie im Dokument "Schnelle, mehrkanalige Ausrichtung in der Photonik" (E712T0016).
Scanprozeduren	Befehle AAP, FIO, FLM, FLS, FSA, FSC, FSM

INFORMATION

Der C-887 kann Befehlsfolgen als Controllermakros (S. 127) speichern und abarbeiten.

Über die Kommunikationsschnittstellen des C-887 können sämtliche Befehle gesendet werden, während auf dem C-887 ein Makro läuft. Der Makroinhalt und Befehle, die über die Kommunikationsschnittstellen empfangen werden, können sich gegenseitig überschreiben.

INFORMATION

Für Achsen mit inkrementellen Sensoren können Bewegungen erst nach einer erfolgreichen Referenzierungsfahrt kommandiert werden (S. 185) (auch als "Initialisierung" bezeichnet).

Das Verhalten der Achsen des Hexapods nach der Referenzierungsfahrt wird durch die Parameter ***Behaviour After Reference Move*** (ID 0x07030401) und ***Target For Motion After Reference Move*** (ID 0x07030402) festgelegt. Entsprechend der Parameterwerte können die Achsen der Plattform z. B. nach der Referenzierungsfahrt automatisch zu einer vorgegebenen Position bewegt werden.

- Wert des Parameters 0x07030401 = 0: Achse bleibt nach der Referenzierungsfahrt in der Referenzposition.
- Wert des Parameters 0x07030401 = 1: Achse fährt nach der Referenzierungsfahrt zur absoluten Zielposition, die durch Parameter 0x07030402 vorgegeben ist.

Für Achsen mit absolut messenden Sensoren ist keine Referenzierungsfahrt erforderlich. Die Verwendung des Befehls FRF wird für diese Achsen trotzdem empfohlen. FRF startet für Achsen mit absolut messenden Sensoren **keine** Referenzierungsfahrt, sondern setzt die Zielpositionen auf die aktuellen Positionswerte. Zusätzlich werden die oben beschriebenen Parameterwerte wirksam, so dass die Achsen z. B. zu einer definierten "Ausgangsposition" bewegt werden können.

INFORMATION

Wenn eine manuelle Bedieneinheit (C-887.MC2 oder C-887.MC) (S. 24) verwendet wird: Der Wert des Parameters ***HID Device Button Mode*** (ID 0x0E001600) bestimmt das Verhalten der Drucktasten für Stop und Referenzierungsfahrt:

- Parameterwert ist 0 (Standard): Die Drucktasten lösen die entsprechenden Aktionen aus.
- Parameterwert ist 1: Die Drucktasten lösen keine Aktionen aus. Das kann z. B. sinnvoll sein, wenn der Status der Drucktasten in einem Makro ausgewertet werden soll (Abfrage mit dem Befehl HIB? (S. 202)).

Der Wert des Parameters kann mit dem Befehl SPA (S. 258) geändert und mit dem Befehl WPA (S. 293) gespeichert werden, siehe "Anpassen von Einstellungen" (S. 317).

Weitere Informationen finden Sie in der Dokumentation der manuellen Bedieneinheit.

3.9.4 Profilgenerator für Punkt-zu-Punkt-Bewegungen

Für Punkt-zu-Punkt-Bewegungen legt der Profilgenerator des C-887 das Dynamikprofil fest.

INFORMATION

Während einer Punkt-zu-Punkt-Bewegung setzt ein beliebiger neuer Bewegungsbefehl die Zielposition auf einen neuen Wert, und die Bewegungsplattform fährt auf einer undefinierten Bahn sofort die neue Zielposition an.

Parameter für den Profilgenerator

Parameter	Beschreibung und mögliche Werte
Path Control Step Size (mm) 0x19001504	Schrittweite für die Berechnung des Dynamikprofils der Plattform Dieser Parameter ist schreibgeschützt und wird vor Auslieferung auf den zum System gehörenden Hexapod abgeglichen.
Trajectory Velocity (Phys. Unit/s) 0x19001510	Geschwindigkeit für die Bewegungsplattform des Hexapods Das Ändern der Geschwindigkeit mit dem Befehl VLS überschreibt den Wert des Parameters im flüchtigen Speicher.
Trajectory Acceleration (Phys. Unit/s2) 0x19001511	Beschleunigung für die Bewegungsplattform des Hexapods
Trajectory Jerk (Phys. Unit/s3) 0x19001512	Ruck für die Bewegungsplattform des Hexapods
Trajectory Source 0x19001900	Quelle des Dynamikprofils für MOV-Befehle Für Punkt-zu-Punkt-Bewegungen, die mit dem Befehl MOV ausgelöst werden, muss der Parameter den Wert 0 haben (Standard).

Befehle für den Profilgenerator

Befehl	Syntax	Funktion
`VLS`	VLS <SystemVelocity>	Setzt die Geschwindigkeit für die Bewegungsplattform des Hexapods. Wird begrenzt durch die Parameter ***Maximum System Velocity (Phys. Unit/s)*** (ID 0x19001500) und ***Minimum System Velocity (Phys. Unit/s)*** (ID 0x19001501).

Prüfung des vom Profilgenerator erzeugten Dynamikprofils

Wenn das Dynamikprofil für den Hexapod durch den Profilgenerator festgelegt wird, erfolgt vor dem Start jeder Bewegung eine Prüfung, ob die Bewegungsplattform die Stützstellen des berechneten Profils und die kommandierte Zielposition tatsächlich erreichen kann. Wenn eine Stützstelle oder die Zielposition nicht erreicht werden kann, wird die Bewegung nicht ausgeführt. Geprüft wird dabei Folgendes:

- Liegen die Stützstellen und die Zielposition außerhalb der Stellwegsgrenzen, die mit `TMN?` (S. 271) und `TMX?` (S. 272) oder `TRA?` (S. 273) abgefragt werden können?
- Sind die mit `NLM` (S. 248) und `PLM` (S. 251) gesetzten Verfahrbereichsgrenzen mit `SSL` (S. 263) aktiviert, und wenn ja, liegen die Stützstellen und die Zielposition außerhalb dieser Verfahrbereichsgrenzen?
- Sind die einzelnen Antriebe in der Lage, die Plattform zu den notwendigen Stützstellen und zur vorgegebenen Zielposition zu bewegen?
- Wenn eine Konfiguration zur Kollisionsvermeidung auf dem C-887 abgelegt wurde (z. B. mit der kostenpflichtigen PIVeriMove Hexapod Software zur Kollisionsprüfung): Treten Kollisionen auf zwischen den folgenden Gruppen?
  - Umgebung inkl. Grundplatte des Hexapods
  - Antriebe des Hexapods
  - Bewegungsplattform des Hexapods inkl. Last

Der Befehl `VMO?` (S. 280) fragt ab, ob eine vorgegebene Zielposition erreicht werden kann.

INFORMATION

Wenn Bewegungsbefehle den Fehlercode 7 ("Position out of limits") erzeugen, obwohl die kommandierte Zielposition zulässig ist, befindet sich die Bewegungsplattform möglicherweise an einer Position außerhalb der Stellwegsgrenzen.

➢ Wenn der Hexapod mit absolut messenden Sensoren ausgestattet ist, deaktivieren Sie temporär die automatische Prüfung des Dynamikprofils, um die Bewegungsplattform zurück an eine zulässige Position kommandieren zu können. Details siehe "Störungsbehebung" (S. 357).

3.9.5 Zyklische Übertragung von Zielpositionen

Optionen für die Vorgabe des Dynamikprofils durch die zyklische Übertragung von Zielpositionen:

- Vorgabe durch aufeinander folgende MOV-Befehle

  Details siehe die nachfolgenden Beschreibungen von Parametern und Befehlen

- Nur C-887-Modelle mit EtherCAT-Schnittstelle: Cyclic-Synchronous-Position-Modus (CSP) gemäß Antriebsprofil CiA402.

  Weitere Informationen siehe "Kommandierung über EtherCAT-Schnittstelle" (S. 47).

HINWEIS

Zyklische Übertragung von Zielpositionen!

Beschleunigung / Verzögerung, Geschwindigkeit und Stetigkeit der Bewegung hängen während der zyklischen Übertragung von Zielpositionen von folgenden Faktoren ab:

- Zielpositionswerte
- Einhalten der Zykluszeit

Die Ausführung eines ungeeigneten Dynamikprofils kann den Hexapod verkippen. Verkippen kann den Hexapod und/oder die auf ihm angebrachte Last beschädigen.

➢ Beachten Sie deshalb bei der zyklischen Übertragung von Zielpositionen:
  – Die Bahn, die durch die Zielpositionen vorgegeben wird, muss mindestens zweimal stetig differenzierbar sein.
  – Bei der Ausführung des Dynamikprofils dürfen die maximal zulässige Geschwindigkeit und Beschleunigung des Hexapods **nicht** überschritten werden.
  – Für die Erzeugung der Zielpositionen und deren stetige Übergabe an den C-887 während der Bewegung wird die Verwendung eines geeigneten Programms empfohlen.

➢ Wenn Sie die Zielpositionen durch aufeinander folgende MOV-Befehle vorgeben: Verwenden Sie den Zwischenspeicher des C-887, um sicherzustellen, dass die Zykluszeit eingehalten wird:
  – Legen Sie das Dynamikprofil vor Ausführung im Zwischenspeicher des C-887 ab. Setzen Sie dazu mit `SPA` den Parameter ***Trajectory Execution*** (ID 0x19001901) auf den Wert 1.
  – Für eine sinnvolle Verwendung des Zwischenspeichers erhöhen Sie den Wert des Parameters ***Threshold for Trajectory Execution*** (0x19001903) mit `SPA` (Standard = 1).

INFORMATION

Empfehlung: Wenn das abzufahrende Dynamikprofil bekannt ist, sollten Sie den Funktionsgenerator (S. 102) als Alternative zu aufeinander folgenden MOV-Befehlen verwenden.

Parameter für die Vorgabe des Dynamikprofils durch aufeinander folgende MOV-Befehle

Parameter	Beschreibung und mögliche Werte
Path Control Step Size (mm) 0x19001504	Schrittweite für die Berechnung des Dynamikprofils der Plattform Dieser Parameter ist schreibgeschützt und wird vor Auslieferung auf den zum System gehörenden Hexapod abgeglichen. Der Abstand der mit aufeinander folgenden `MOV`-Befehlen gesetzten Zielpositionen zueinander darf maximal so groß sein wie der Wert des Parameters 0x19001504, um ein Verkippen des Hexapods zu vermeiden.
Trajectory Source 0x19001900	Quelle des Dynamikprofils für MOV-Befehle Für die zyklische Übertragung von Zielpositionen durch aufeinander folgende MOV-Befehle muss der Parameter den Wert 1 haben.
Trajectory Execution 0x19001901	Ausführung des Dynamikprofils Bestimmt, wie das durch aufeinander folgende `MOV`-Befehle festgelegte Dynamikprofil ausgeführt wird: 0 = Dynamikprofil wird sofort ausgeführt (Standard) 1 = Dynamikprofil wird vor Ausführung in einem Zwischenspeicher abgelegt Dieser Parameter wird nur ausgewertet, wenn Parameter 0x19001900 den Wert 1 hat.
Maximum Number of Trajectory Points 0x19001902	Maximale Anzahl der Dynamikprofilpunkte Gibt die maximale Größe des Zwischenspeichers an. Dieser Parameter ist schreibgeschützt und wird nur ausgewertet, wenn die Parameter 0x19001900 und 0x19001901 jeweils den Wert 1 haben.
Threshold for Trajectory Execution 0x19001903	Schwellenwert für Ausführung des Dynamikprofils Bestimmt, wie viele Dynamikprofilpunkte im Zwischenspeicher abgelegt sein müssen (durch aufeinander folgende `MOV`-Befehle), bis die Ausführung des Dynamikprofils beginnt. Dieser Parameter wird nur ausgewertet, wenn die Parameter 0x19001900 und 0x19001901 jeweils den Wert 1 haben.
Current Number Threshold for Trajectory Execution 0x19001904	Zeigt die aktuelle Anzahl der Dynamikprofilpunkte im Zwischenspeicher an. Der Parameterwert ist immer 0, wenn das Dynamikprofil durch den Profilgenerator festgelegt wird oder das durch aufeinander folgende `MOV`-Befehle festgelegte Dynamikprofil sofort ausgeführt wird. Dieser Parameter ist schreibgeschützt.

Befehle für die Vorgabe des Dynamikprofils durch aufeinander folgende MOV-Befehle

Befehl	Syntax	Funktion
`MOV`	MOV {<AxisID> <Position>}	Aufeinander folgende MOV-Befehle geben die einzelnen Zielpositionen vor. Andere Bewegungsbefehle und das Starten der Funktionsgeneratorausgabe sind nicht zulässig.
`SCT`	SCT "T" <CycleTime>	Legt die Zykluszeit für das Ausführen des Dynamikprofils fest. Die Zykluszeit wird verwendet, um während der Bewegung die Geschwindigkeit so zu berechnen, dass die vorgegebenen Punkte des Dynamikprofils jeweils genau am Ende des Zeitintervalls erreicht werden (sofern unter Einhaltung der Grenzwerte für Geschwindigkeit und Beschleunigung möglich).
`VLS`	VLS <SystemVelocity>	Begrenzt die Geschwindigkeiten der einzelnen Antriebe.
`#11`	#11	Fragt den freien Speicherplatz des Zwischenspeichers ab, dessen Inhalt das Dynamikprofil des Hexapods festlegt, wenn die Parameter 0x19001900 und 0x19001901 jeweils den Wert 1 haben.

3.9.6 Koordinatensysteme

Positionsanzeige, Bewegungsrichtung und Drehpunkt für die Bewegungsplattform des Hexapods werden durch miteinander verkettete Koordinatensysteme bestimmt. Die Kette ist grundsätzlich wie folgt aufgebaut (Ausgangspunkt > Endpunkt): --> Koordinatensystem HEXAPOD, --> Korrektur-Koordinatensystem, --> Achsorientierungs-Koordinatensystem, --> Betriebs-Koordinatensystem.

Die Koordinatensysteme sind immer rechtshändige Systeme.

Das Koordinatensystem HEXAPOD bestimmt die grundsätzlichen Eigenschaften aller anderen Koordinatensysteme. HEXAPOD basiert auf der Konfigurationsdatei mit den Geometriedaten des Hexapods. Die Maßzeichnung im Handbuch des Hexapods zeigt jeweils die Lage des Koordinatensystems HEXAPOD.

Mit dem Controller können eigene Koordinatensysteme definiert und anstelle der werkseitig eingestellten Koordinatensysteme verwendet werden.

Achsorientierungs- und Korrektur-Koordinatensystem passen, basierend auf HEXAPOD, grundlegende Eigenschaften des aktiven Betriebs-Koordinatensystems an und müssen in den meisten Anwendungen gar nicht oder nur einmalig anwenderspezifisch definiert und aktiviert werden.

Mit dem Betriebs-Koordinatensystem werden die Positionsanzeige, die Bewegungsrichtung und der Drehpunkt für die Bewegungsplattform des Hexapods an die Anwendung angepasst. Es ist auch möglich, --> Work- und Tool-Koordinatensysteme zu verwenden. In der Werkseinstellung ist das Betriebs-Koordinatensystem ZERO aktiv.

Die wichtigsten Befehle für die Arbeit mit benutzerdefinierten Koordinatensystemen:

- KSD (S. 229): Betriebs-Koordinatensystem durch Angabe von Offsetwerten für die Achsen X, Y, Z, U, V und W definieren
- KSF (S. 231): Betriebs-Koordinatensystem an der aktuellen Position der Bewegungsplattform des Hexapods definieren ("Homekoordinatensystem")
- KLN (S. 221): Koordinatensysteme miteinander verketten
- KEN (S. 211): Koordinatensysteme aktivieren
- WPA SKS (S. 293): Sichern der aktuell gültigen Einstellungen für Koordinatensysteme
- DPA SKS (S. 167): Einstellungen für Parameter und Koordinatensysteme auf Werkseinstellungen zurücksetzen

Für die Modelle C-887.53, .531 .532, und .533 ist nach dem Einschalten oder Neustart standardmäßig die EtherCAT-Schnittstelle (S. 47) aktiviert. Die verwendeten Koordinatensysteme (Work- und Tool-Koordinatensysteme) werden über die EtherCAT-Schnittstelle durch anwendungsspezifische Objekte (SDO) definiert und aktiviert. Die Einstellungen für Koordinatensysteme können bei aktivierter EtherCAT-Schnittstelle nicht durch GCS-Befehle geändert werden.

INFORMATION

In PIMikroMove® können Koordinatensysteme komfortabel definiert, verkettet, aktiviert und gesichert werden.

Im Fenster ***Positioner Platform*** sind dafür die Schaltflächen ***Define Home Coordinate System (KSF)*** und ***Manage Coordinate Systems ...*** verfügbar. Auf der Registerkarte ***Positioner 3D View*** werden der Hexapod und das aktive Koordinatensystem grafisch abgebildet.

Wenn im Hauptfenster **nicht** das Fenster ***Positioner Platform*** und die Registerkarte ***Positioner 3D View*** angezeigt werden:

- ➢ Blenden Sie das Fenster ***Positioner Platform*** mit dem Menüeintrag ***C-887 > Show Positioner Platform Settings*** ein.
- ➢ Blenden Sie die Registerkarte ***Positioner 3D View*** mit dem Menüeintrag ***C-887 > Positioner 3D View > Show*** ein.

Weitere Informationen zu Koordinatensystemen und zum Work-und-Tool-Konzept finden Sie im Dokument "Koordinatensysteme für Hexapod-Mikroroboter" (C887T0007) und in "Begriffserklärung" (S. 2).

Ein Beispiel für die Verwendung benutzerdefinierter Koordinatensysteme siehe "Beispiel für die Definition eines Betriebs-Koordinatensystems mit dem Befehl KSD" (S. 44).

3.9.7 Rotationen

Rotationen erfolgen um den Drehpunkt. Eine beliebige Rotation im Raum wird aus den Einzelrotationen in der Reihenfolge U-> V -> W berechnet. Dies geschieht unabhängig davon, ob die Werte für U, V und W mit dem aktuellen Befehl explizit gegeben wurden oder aus den vorangegangenen Befehlen resultieren.

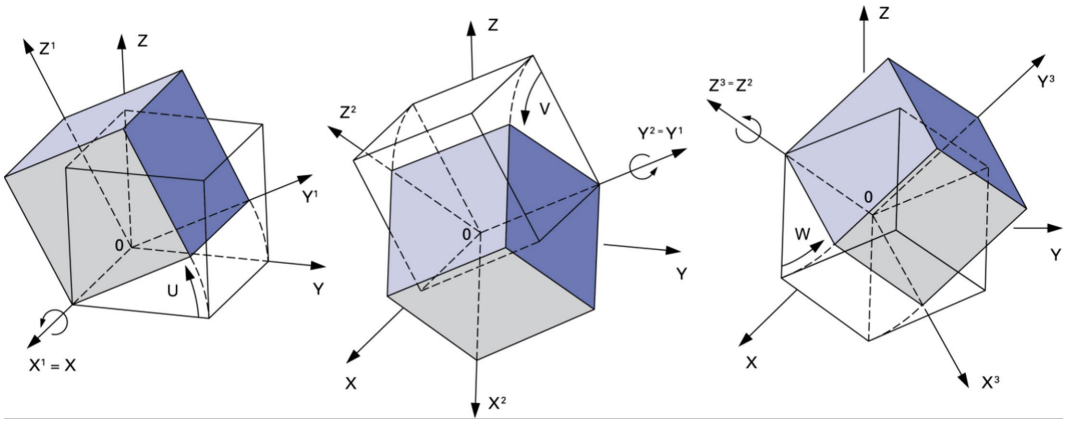


Abbildung 4: Reihenfolge der Elementardrehungen der Hexapod-Plattform beim Erreichen einer Position (von links nach rechts)

Der Drehpunkt beschreibt das Rotationszentrum (Schnittpunkt der Rotationsachsen U, V und W).

Der Drehpunkt bewegt sich immer zusammen mit der Plattform.

Der Drehpunkt kann durch die Definition und Aktivierung eines Betriebs-Koordinatensystems vom Typ KSD oder von Work- und Tool-Koordinatensystemen (Typen KSW und KST) jederzeit geändert werden.

Der Drehpunkt kann mit dem Befehl SPI (S. 260) aus dem Ursprung des Koordinatensystems heraus in X- und/oder Y- und/oder Z-Richtung verschoben werden, wenn als Betriebs-Koordinatensystem das Koordinatensystem ZERO oder ein Koordinatensystem vom Typ KSF aktiv ist und für die Rotationskoordinaten der Bewegungsplattform U = V = W = 0 gilt. Der mit dem Befehl SPI verschiebbare Drehpunkt wird auch als "Pivotpunkt" bezeichnet.

Weitere Informationen zur Berechnung von Translationen und Rotationen durch den C-887 finden Sie im Dokument "Bewegungen des Hexapods - Position und Orientierung im Raum, Drehpunkt" (C887T0021).

3.9.8 Beispiel für die Definition eines Betriebs-Koordinatensystems mit dem Befehl KSD

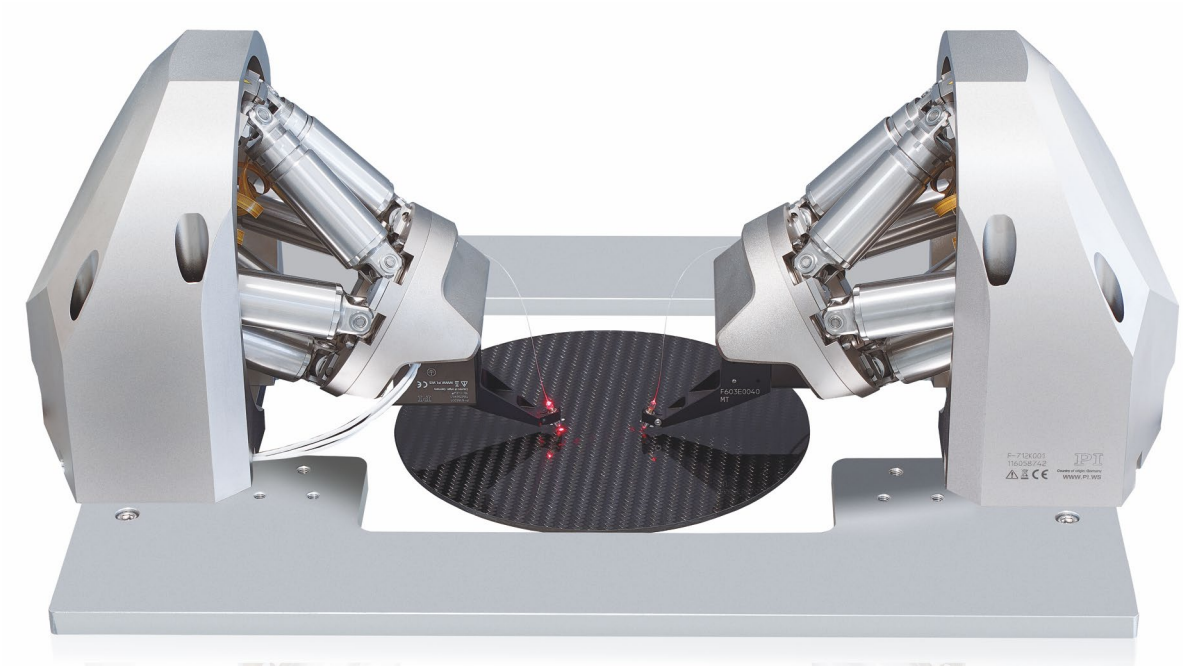


Abbildung 5: F-712.HA2 Hochpräzises Faserausrichtungssystem: Anwendungsbeispiel

In einem hochpräzisen Faserausrichtungssystem F-712.HA2, das auf Sender- und Empfängerseite jeweils einen Hexapod H-811 enthält, werden kundenspezifische Faserhalter verwendet. Bei Auslieferung des Faserausrichtungssystems sind die werkseitig eingestellten Koordinatensysteme für Sender und Empfänger aktiv (Koordinatensysteme sender, receiver). Der Drehpunkt für Bewegungen des Hexapods soll aber jeweils in der Faserspitze liegen. Deshalb werden in den Hexapod-Controllern für Sender- und Empfängerseite kundenspezifische Koordinatensysteme definiert, mit denen der Drehpunkt aus der werkseitigen Position in die Faserspitze verschoben wird. Der Abstand zwischen dem werkseitigen Drehpunkt und der Faserspitze beträgt im nachfolgenden Beispiel 85,37 mm in X-Richtung und 71,88 mm in Z-Richtung, siehe auch die Abbildung unten. Die neuen Koordinatensysteme sollen fibertip1s (Sender) und fibertip1r (Empfänger) heißen und beim Einschalten oder Neustart des Faserausrichtungssystems sofort aktiv sein.

Für den Hexapod-Controller auf der Senderseite werden folgende Befehle gesendet:

Befehl	Funktion
KSD fibertip1s X 85.37 Z 71.88	Definiert das neue Koordinatensystem
KLN fibertip1s sender	Hängt das neue Koordinatensystem als Nachfolger an das Koordinatensystem sender an
KEN fibertip1s	Aktiviert das neue Koordinatensystem
WPA SKS	Sichert die Einstellungen im permanenten Speicher

Das werkseitig eingestellte Koordinatensystem receiver für die Empfängerseite ist so gedreht, dass die Richtung der X-Achse in Bezug auf die Senderseite invertiert ist. Deshalb muss bei der Definition des neuen Koordinatensystems für die Empfängerseite ein negativer Offsetwert für die X-Achse angegeben werden. Für den Hexapod-Controller auf der Empfängerseite werden folgende Befehle gesendet:

Befehl	Funktion
KSD fibertip1r X -85.37 Z 71.88	Definiert das neue Koordinatensystem
KLN fibertZp1r receiver	Hängt das neue Koordinatensystem als Nachfolger an das Koordinatensystem receiver an
KEN fibertip1r	Aktiviert das neue Koordinatensystem
WPA SKS	Sichert die Einstellungen im permanenten Speicher

Die nachfolgende Abbildung zeigt die Koordinatensysteme und ihre Definition am Beispiel der Senderseite. Zum besseren Verständnis wird nicht nur das resultierende Koordinatensystem fibertip1s dargestellt, sondern auch die anderen Betriebs-Koordinatensysteme, von denen fibertip1s durch Verkettung seine Eigenschaften übernimmt: ZERO und sender (basiert auf ZERO). Durch die Umsetzung einzelner Anpassungsschritte als separate Koordinatensystem-Definitionen und die anschließende Verkettung dieser Koordinatensysteme werden die komplexen Berechnungen für das resultierende Koordinatensystem in den C-887 verlagert. Außerdem lassen sich die einzelnen Anpassungsschritte flexibel kombinieren.

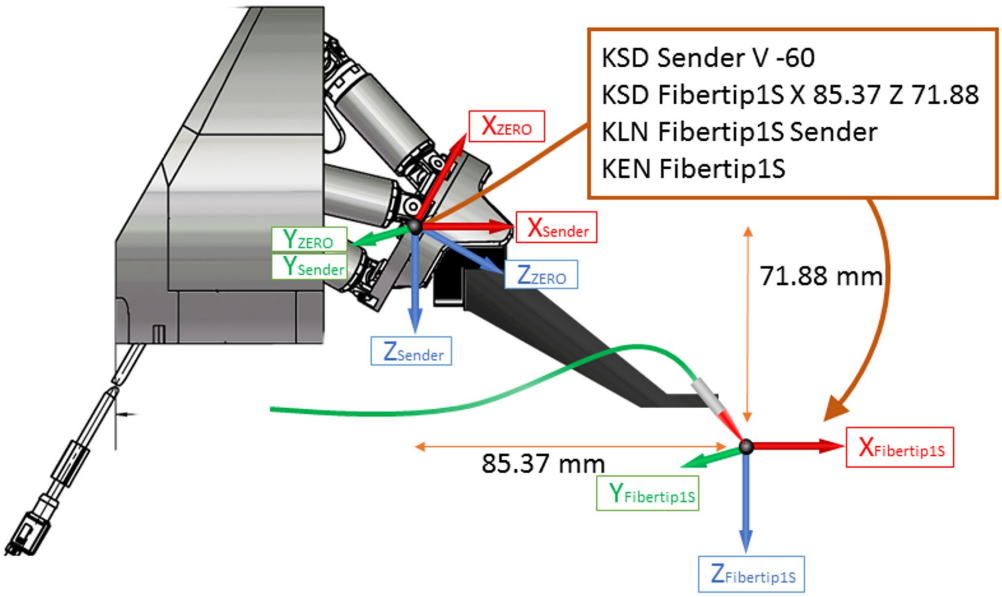


Abbildung 6: F-712.HA2: Drehpunkt für Senderseite in Faserspitze verschieben

3.9.9 Bewegungsstatus, Einschwingfenster, Einschwingzeit

Bewegungsstatus

Der C-887 zeigt den Bewegungsstatus und damit das Erreichen der Zielposition anhand von Statusregisterbits an. Statusregister sind verfügbar für die Hexapod-Beine und die Achsen A und B.

Registerbits für die Anzeige des Bewegungsstatus:

- Statusregister für jedes Hexapod-Bein und die Achsen A und B (S. 386):
  - Bit 13 = 1: Hexapod-Bein / Achse A / B ist in Bewegung
  - Bit 13 = 0: Hexapod-Bein / Achse A / B ist nicht in Bewegung
  - Bit 15 = 1: Zielposition ist erreicht
  - Bit 15 = 0: Zielposition ist nicht erreicht

  Sie können die Bits der Register mit dem Befehl SRG? (S. 262) abfragen. Zusätzlich können Sie diese Bits mit dem Datenrekorder des C-887 (S. 99) aufzeichnen, Aufzeichnungsoption 80 (Status register of axis).
- Gemeinsames Systemstatusregister (S. 386):
  - Bits 8 bis 15 = 1: Zielposition ist nicht erreicht (entsprechende/s Hexapod-Bein / Achse A / B ist in Bewegung)
  - Bits 8 bis 15 = 0: Zielposition ist erreicht (entsprechende/s Hexapod-Bein / Achse A / B ist nicht in Bewegung)

  Sie können die Bits des Systemstatusregisters mit den Befehlen STA? (S. 266) und #4 (S. 152) abfragen.

Für achsenbezogene Abfragen sind außerdem folgende Befehle verfügbar:

- Befehl #5 (S. 153): Fragt den Bewegungsstatus der Achsen ab.

  Der Bewegungsstatus *aller* Achsen des Hexapods (X bis W) lautet "in Bewegung", solange mindestens *ein* Hexapod-Bein in Bewegung ist (Beinstatus gemäß den oben genannten Registerbits).
- Befehl ONT? (S. 250): Fragt den On-Target-Status der angegebenen Achse ab.

  Für die Achsen des Hexapods (X bis W) prüft ONT? nur, ob das Ende des Dynamikprofils erreicht ist, aber **nicht** den tatsächlichen Bewegungsstatus.

Einschwingfenster, Einschwingzeit

Der Bewegungsstatus der Hexapod-Beine bezieht sich auf die Zielpositionen, die der C-887 für die Beine aus den Zielpositionen der Achsen des Hexapods errechnet hat. Der C-887 ermittelt den Bewegungsstatus der Hexapod-Beine und der Achsen A und B anhand folgender Kriterien:

- Einschwingfenster um die Zielposition:

  Vorgabe durch Parameter ***Settling Window (encoder counts)***, ID 0x36. Der Parameterwert entspricht der Hälfte der Fensterbreite. Einheit: Impulse des Encoders.

- Verzögerungszeit für das Setzen des Statusbits auf "Zielposition ist erreicht":

  Vorgabe durch Parameter ***Settle Time***, ID 0x38. Einheit: Anzahl Servozyklen.
- Die Zielposition gilt als erreicht, wenn die aktuelle Position im Einschwingfenster ist und dort mindestens für die Dauer der Verzögerungszeit bleibt.

3.10 Kommandierung über EtherCAT-Schnittstelle

3.10.1 Einführung

Der EtherCAT-Master gibt die Zielpositionen für die Achsen der Bewegungsplattform des Hexapods in kartesischen Koordinaten vor und wertet die entsprechende Positionsrückmeldung der Achsen aus. Der C-887.53x (.53x steht für C-887.53, .531, .532, .533) wandelt die Zielpositionen der Achsen in Bewegungsbefehle für die sechs Antriebe des Hexapods um. Das Hexapod-System (C-887.53x plus Hexapod) verhält sich wie ein intelligenter Mehrachsantrieb gemäß dem Antriebsprofil CiA402.

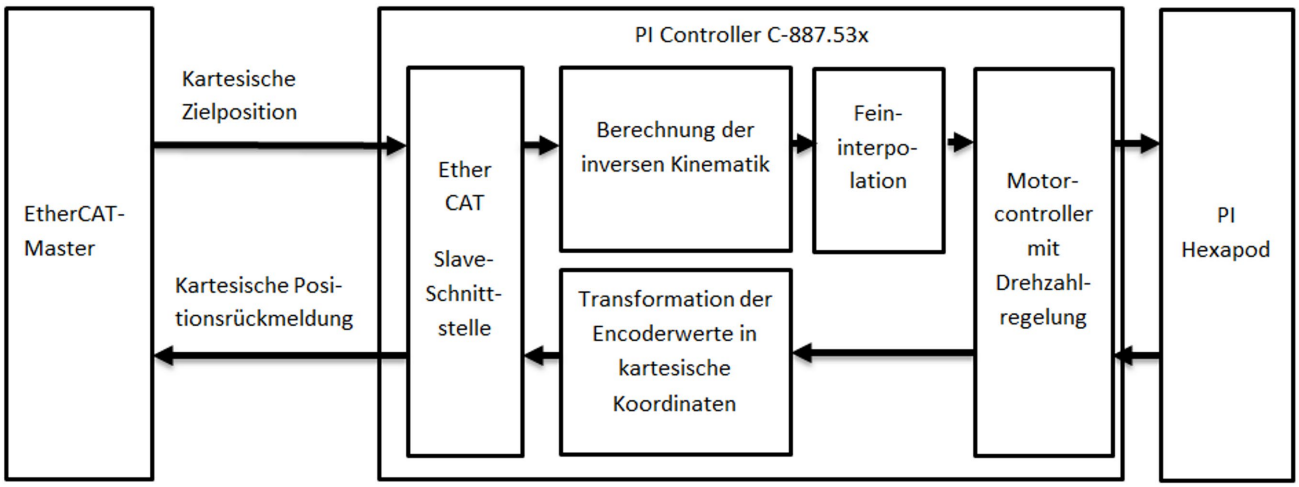


Abbildung 7: Vom EtherCAT-Master kommandiertes Hexapod-System

Die nachfolgende Tabelle gibt einen Überblick über die wichtigsten Merkmale der Kommandierung des C-887.53x durch einen EtherCAT-Master:

Feldbus-Protokoll:	EtherCAT (CoE = CANopen over EtherCAT)
Antriebsprofil:	CiA402 (IEC 61800-7-201)
Typ des kommandierten Geräts, Anzahl der Achsen:	Mehrachsgerät mit 6 einzelnen kartesischen Achsen
Zykluszeit für Vorgabe von Zielpositionen, Signalverarbeitung und Synchronisation:	≥1 ms

Unterstützte Betriebsmodi gemäß CiA402:	**Modus** / **Objekt 0x6060** / **Anmerkung** Keine Modus-änderung / kein Modus ausge-wählt / 0 / Sicherer Grundzustand, Zielpositionen werden ignoriert; erforderlich für das Aktivieren von Koordinatensystemen Homing Modus / 6 / Referenzierungsfahrt ausführen Cyclic-Synchro-nous-Position-Modus (CSP) / 8 / Zyklische Vorgabe von Zielpositionen durch den EtherCAT-Master
Unterstützter Synchronisationsmodus:	Distributed Clocks, Synchron mit SYNC0 Event
Anschluss an den Feldbus:	Buchse RJ45

Die EtherCAT-Schnittstelle wird im C-887.53x durch eine Parametereinstellung aktiviert, Standardeinstellung: Nach dem Einschalten oder Neustart des C-887.53x ist die EtherCAT-Schnittstelle aktiviert. Details siehe "Konfiguration des C-887 für Kommandierung durch EtherCAT-Master" (S. 48).

Die Integration des C-887.53x in das EtherCAT-Netzwerk erfolgt über eine von PI bereitgestellte XML-Datei, Details siehe "Konfiguration des EtherCAT-Masters" (S. 49).

Die LED **RUN** des C-887.53x (S. 18) zeigt den aktuellen Zustand der EtherCAT-Kommunikationszustandsmaschine an.

Weitere Informationen finden Sie im Dokument "EtherCAT Interface Description" (C887T0011).

3.10.2 Konfiguration des C-887 für Kommandierung durch EtherCAT-Master

Aktivierung der EtherCAT-Schnittstelle

Bewegungen der Achsen X, Y, U, V, W und Z können bei Modellen mit EtherCAT-Schnittstelle durch einen EtherCAT-Master ausgelöst werden (Details siehe "Bewegungen des Hexapods" (S. 34)). Die EtherCAT-Schnittstelle des C-887 wird durch den Wert des Parameters ***Configure Command Mode*** (ID 0x19002000) aktiviert. Mögliche Werte des Parameters ***Configure Command Mode***:

- 0 = EtherCAT-Schnittstelle deaktiviert ("GCS")
- 1 = EtherCAT-Schnittstelle aktiviert ("External: EtherCAT"); Standardeinstellung für die Modelle C-887.53, .531, .532, .533

Der Wert des Parameters ***Configure Command Mode*** kann mit GCS-Befehlen über die TCP/IP- oder RS-232-Schnittstelle des C-887.53x geändert werden.

1. Gewünschten Aktivierungszustand wählen:

   `SPA 1 0x19002000 1`

   oder

   `SPA 1 0x19002000 0`

2. Geänderten Parameterwert aktivieren:

   a) Parameterwert im permanenten Speicher des C-887 speichern:
   `WPA 101 1 0x19002000`

   b) C-887 neu starten durch Senden von `RBT` oder Aus- und Einschalten des C-887.

EtherCAT-Schnittstelle und GCS-Befehle

Wenn die EtherCAT-Schnittstelle durch den Parameter ***Configure Command Mode*** aktiviert ist, können immer noch GCS-Befehle über die TCP/IP- oder RS-232-Schnittstelle des C-887.53x gesendet werden. Die Ausführung der folgenden GCS-Befehle wird jedoch vom C-887.53x verweigert, wenn die Hexapod-Achsen im Zustand ***Operation enabled*** der CiA402-Zustandsmaschine sind:

- Befehle, die Bewegungen auslösen (z. B. MOV)
- Befehle, die Bewegungen stoppen (z. B. HLT und STP)
- Befehl WPA

Der Zustand ***Operation enabled*** kann nicht durch GCS-Befehle verlassen werden.

Wenn die EtherCAT-Schnittstelle aktiviert ist, werden die verwendeten Koordinatensysteme (S. 41) durch anwendungsspezifische Objekte (SDO) definiert und aktiviert. Die Einstellungen für Koordinatensysteme können bei aktivierter EtherCAT-Schnittstelle nicht durch GCS-Befehle geändert werden.

Die GCS-Kommunikation kann die Leistungsfähigkeit der EtherCAT-Schnittstelle des C-887.53x reduzieren. Deshalb sollte die GCS-Kommunikation vermieden werden (z. B. das Senden von GCS-Befehlen oder die Verwendung von PC-Software wie PIMikroMove®), wenn die EtherCAT-Schnittstelle verwendet wird. Verwenden Sie die GCS-Kommunikation nur für die erste Inbetriebnahme, in Supportfällen und zur Fehlersuche.

Wenn die EtherCAT-Schnittstelle deaktiviert ist, ist keine Kommandierung durch den EtherCAT-Master möglich, sondern nur durch GCS-Befehle.

3.10.3 Konfiguration des EtherCAT-Masters

Die Schritte für Konfiguration, Inbetriebnahme und Betrieb des EtherCAT-Masters hängen vom verwendeten Gerät ab. Details finden Sie in der Dokumentation Ihres EtherCAT-Masters.

Für die Integration des C-887 in das EtherCAT-Netzwerk muss die von PI bereitgestellte XML-Datei Physik_Instrumente_Hexapod.xml auf dem EtherCAT-Master gespeichert werden. Die XML-Datei finden Sie im Verzeichnis EDS auf dem Datenträger aus dem Lieferumfang des C-887.

Zusätzlich müssen die folgenden Einstellungen des EtherCAT-Masters geändert werden, um ihn an den C-887 anzupassen.

Einzustellen pro kartesischer Achse des Hexapod-Systems:

- Scaling Factor Numerator: 1
  Scaling Factor Denominator: 100000
- Reference System: Absolut (der C-887 liefert absolute Positionen an den EtherCAT-Master)
- Totzeitkompensation: Hängt von der Zykluszeit ab. Berechnung und Beispiele:
  Totzeit = 4 * Zykluszeit + 5 * Interpolationspufferzeit (2 ms)
  1 ms Zykluszeit: 14 ms Totzeit
  2 ms Zykluszeit: 18 ms Totzeit
  Empirische Optimierung empfohlen
- Beschleunigung, Geschwindigkeit und Ruck sollten an den Hexapod angepasst werden, um Bewegungsfehler zu vermeiden (technische Daten siehe Dokumentation des Hexapods)

Einzustellen für das gesamte Hexapod-System:

- Synchronisationsmodus: Distributed Clocks, Synchron mit SYNC0 Event

Während der Kommandierung durch den EtherCAT-Master muss die minimale Zykluszeit des Hexapod-Systems (1 ms) eingehalten werden. Wenn die tatsächliche Zykluszeit kürzer ist als die minimale Zykluszeit des Hexapod-Systems, bewegt sich der Hexapod nicht.

Während der Kommandierung durch den EtherCAT-Master kann die tatsächliche Zykluszeit des C-887 über den Wert des Parameters ***Cycletime For Interpolation In External Mode*** (ID 0x19002010) ausgelesen werden. Der Parameterwert kann mit dem GCS-Befehl SPA? (S. 259) über die TCP/IP- oder RS-232-Schnittstelle des C-887 ausgelesen werden.

3.11 Kommunikationsschnittstellen

Der C-887 kann über folgende Schnittstellen mit ASCII-Befehlen gesteuert werden:

- TCP/IP
- Serielle RS-232-Verbindung

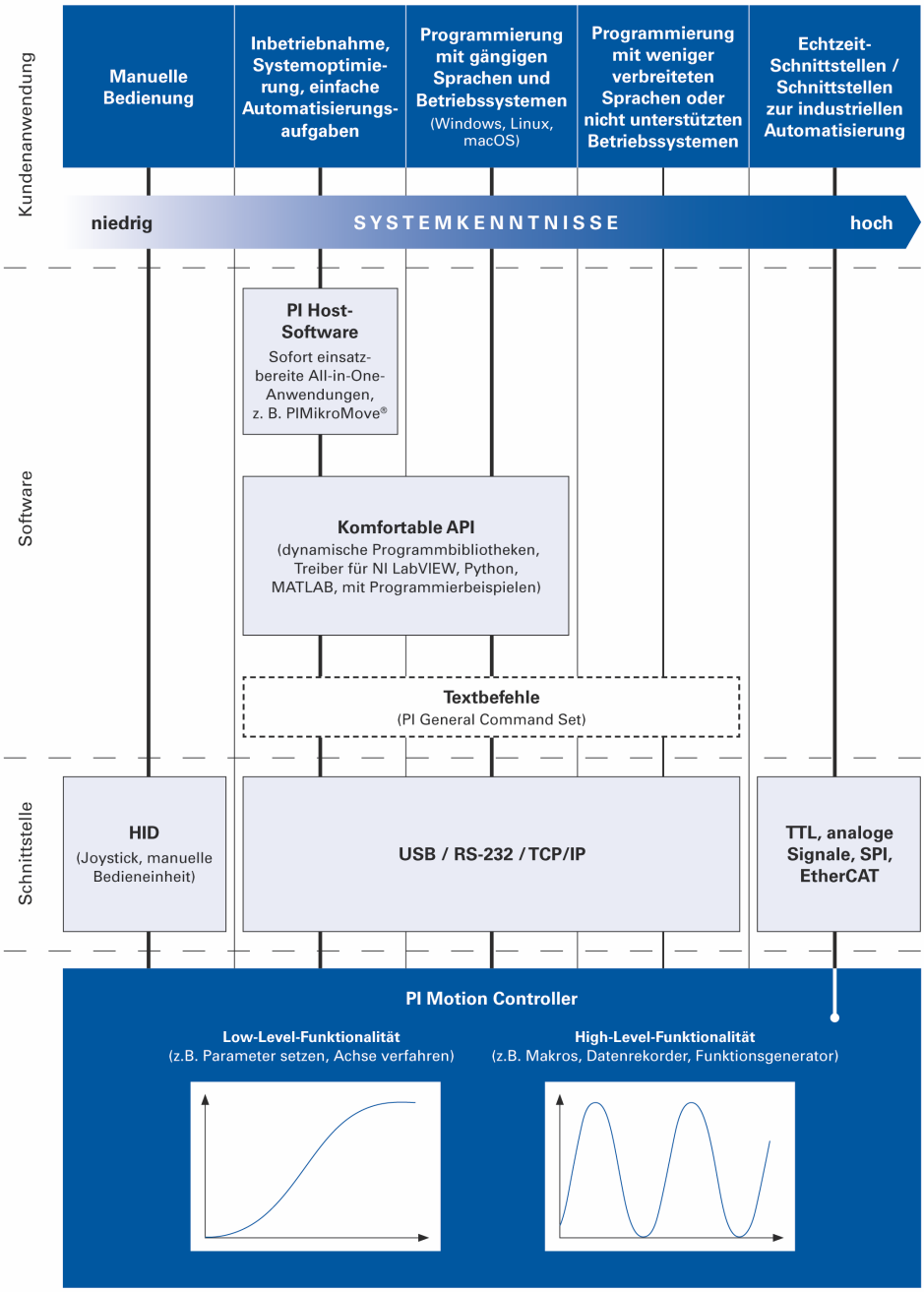
Die Modelle C-887.53, .531, .532 und .533 sind mit einer EtherCAT-Schnittstelle ausgestattet. Bei Kommandierung über die EtherCAT-Schnittstelle wird das Hexapod-System als Mehrachsgerät entsprechend des Antriebsprofils CiA402 verwendet. Der EtherCAT-Master kommandiert die logischen Achsen X, Y, Z, U, V und W der Bewegungsplattform des Hexapods.

Weitere Informationen siehe "Kommandierung über EtherCAT-Schnittstelle" (S. 47).

3.12 PC-Softwareübersicht

3.12.1 Ansteuerung von PI Systemen

Systeme von PI können grundsätzlich wie folgt angesteuert werden:



3.12.2 PI Software Suite

Im Lieferumfang (S. 23) des C-887 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 Softwareprogrammierung für den C-887 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.
PI Hexapod 3D Library	Programmbibliothek, die folgende Funktionen für PIMikroMove® bereitstellt: ▪ 3D-Darstellung der Bewegungen des Hexapods und des aktiven Koordinatensystems ▪ Konfiguration und Verwaltung von benutzerdefinierten Koordinatensystemen	Wird für PIMikroMove® benötigt.
PI Hexapod DataFiles	Konfigurationsdateien für die grafische Darstellung der Hexapoden in der PC-Software	Werden für die PI Hexapod 3D Library 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 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.	Für Anwender, die MATLAB zur Programmierung ihrer Anwendung verwenden möchten.

PC-Software	Kurzbeschreibung	Empfohlene Verwendung
PIPython	Eine Sammlung von Python-Modulen für die komfortable Nutzung von Elektroniken und GCS-Daten. Die Module können mit Python 3.6+ eingesetzt werden. Über Sockets ist auch der Einsatz auf weiteren Betriebssystemen möglich.	Für Anwender, die Python zur Programmierung von Skripten für ihre Anwendung verwenden möchten. Die Verwendung von Python erweitert den Funktions-umfang des GCS-Befehlssatzes beträchtlich.

Anwendersoftware

PC-Software	Kurzbeschreibung	Empfohlene Verwendung
PIVirtualMove	Simulationsprogramm für parallelkinematische Mehrachssysteme von PI. Bestimmte Posen können für einen bestimmten Aufbau des Systems (Einbaulage, zu bewegende Last, externe Kräfte und Momente, verwendetes Koordinatensystem) simuliert und geprüft werden hinsichtlich: ▪ Maximale und tatsächlich verfügbare Stellwege ▪ Kräfte, die aktuell auf die Beine oder Aktoren der Mechanik einwirken	Für alle Anwender.
PIHexapod Emulator	Programm, mit dem der C-887 und der angeschlossene Hexapod sowie die Achsen A und B emuliert werden können. Der PIHexapodEmulator kann direkt gestartet werden (verbinden über TCP/IP mit der Adresse *localhost* und dem Port *50000*) oder aus PIMikroMove® (im Dialog ***Start up controller*** den C-887 auswählen und auf der Registerkarte ***PIHexapodEmulator*** die Verbindung herstellen).	Für Anwender, die das Verhalten des Hexapod-Systems testen möchten, wenn Controller und/oder Hexapod nicht zur Verfügung stehen.

PC-Software	Kurzbeschreibung	Empfohlene Verwendung
PIMikroMove®	Grafische Benutzerschnittstelle für Windows, mit der der C-887 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 Automatisierungsaufgaben ausführen oder ihre Ausrüstung vor oder anstelle der Programmierung einer Anwendung testen möchten. Ein Logfenster mit Anzeige der gesendeten Befehle ermöglicht auch das Erlernen der Befehls-verwendung.
PITerminal	Terminalprogramm, das für nahezu alle Controller von PI 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 Positioniererdaten-banken auseinandersetzen möchten.
PIUpdateFinder	Überprüft die auf dem PC installierte Software von PI. Wenn auf dem Server von PI aktuellere Versionen der PC-Software vorhanden sind, wird das Herunterladen angeboten.	Für Anwender, die die PC-Software aktualisieren möchten.
PIFirmware Manager	Programm zur Aktualisierung der Firmware des C-887.	Für Anwender, die die Firmware aktualisieren möchten.

4 Auspacken

1. Packen Sie den C-887 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. 367).
4. Bewahren Sie das komplette Verpackungsmaterial auf für den Fall, dass das Produkt zurückgeschickt werden muss.

5 Installation

In diesem Kapitel

Allgemeine Hinweise zur Installation 57
PC-Software installieren 58
Belüftung sicherstellen 61
C-887 erden 61
C-887 an die Stromversorgung anschließen 61
Hexapod installieren 62
Hexapod über Kabelsatz an C-887 anschließen 63
Positionierer für Achsen A und B anschließen 64
Analoge Signalquellen anschließen 65
Digitale Ein- und Ausgänge anschließen 66
PC anschließen 67
EtherCAT-Master anschließen 68

5.1 Allgemeine Hinweise zur Installation

Der Hexapod kann in beliebiger Orientierung montiert werden.

HINWEIS

Unzulässige mechanische Belastung und Kollisionen!

Unzulässige mechanische Belastung und Kollisionen zwischen Hexapod, zu bewegender Last und Umgebung können den Hexapod beschädigen.

- Halten Sie den Hexapod nur an der Grundplatte.
- Ermitteln Sie mit dem Simulationsprogramm PIVirtualMove vor der ersten Installation von Hexapod und Last die zulässigen Posen. Für die Definition einer zulässigen Pose siehe "Zulässige Posen ermitteln" (S. 62).
- Wiederholen Sie die Ermittlung der zulässigen Posen, wenn Sie den Hexapod-Typ wechseln und vor jeder Änderung von Einbaulage, zu bewegender Last, externen Kräften und Momenten oder verwendetem Koordinatensystem.
- Wenn Sie ein eigenes Koordinatensystem definieren und anstelle des werkseitig eingestellten Koordinatensystems verwenden: Beachten Sie, dass das Simulationsprogramm PIVirtualMove bei der Verwendung benutzerdefinierter Koordinatensysteme engere Stellwegsgrenzen berechnet als der Controller. Wenn Sie das Kommandieren von Posen vermeiden wollen, die außerhalb der von PIVirtualMove berechneten Stellwegsgrenzen liegen:
  - Ermitteln Sie mit dem Controller durch Abfragen mit dem Befehl `TRA?`, ob Posen kommandiert werden können, die außerhalb der von PIVirtualMove berechneten Stellwegsgrenzen liegen.

- Wenn notwendig, setzen Sie im Controller mit den Befehlen `NLM` und `PLM` Verfahrbereichsgrenzen entsprechend den von PIVirtualMove berechneten Stellwegsgrenzen.

➢ Stellen Sie bei der Installation des Hexapods und der Last sicher, dass der tatsächliche Systemaufbau dem Systemaufbau entspricht, für den Sie im Simulationsprogramm PIVirtualMove die zulässigen Posen ermittelt haben.

➢ Vermeiden Sie bei der Installation hohe Kräfte und Momente auf die Bewegungsplattform.

➢ Sorgen Sie für eine unterbrechungsfreie Stromversorgung, um eine ungewollte Deaktivierung des Hexapod-Systems und daraus resultierende ungewollte Positionsänderungen des Hexapods zu vermeiden.

➢ Stellen Sie sicher, dass im Arbeitsraum des Hexapods keine Kollisionen zwischen Hexapod, zu bewegender Last und Umgebung möglich sind.

5.2 PC-Software installieren

Die Kommunikation zwischen dem C-887 und einem PC ist zur Konfiguration des C-887 und zur Bewegungskommandierung mit den Befehlen des GCS notwendig. Dafür stehen verschiedene PC-Softwareanwendungen zur Verfügung.

5.2.1 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 für GCS
   - PIMikroMove®
   - PC-Software zum Aktualisieren der Firmware des C-887
   - 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.
4. 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.

5.2.2 Updates installieren

Die PI Software Suite wird von PI ständig verbessert.

➢ Installieren Sie immer die neueste Version der PI Software Suite 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 PISTAGES3.DB 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***
5. 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. 367).

PISTAGES3.DB auf Linux aktualisieren

1. Wenden Sie sich an den Kundendienst (S. 367), um die neueste Version der Positioniererdatenbank PISTAGES3.DB zu erhalten.
2. Melden Sie sich am PC als Superuser (Root-Rechte) an.
3. Installieren Sie das Update, das Sie von unserem Kundendienst erhalten haben, auf dem PC.

5.2.3 Kundenspezifische Positioniererdatenbank installieren

Mit einem kundenspezifischen Positionierer erhalten Sie von PI einen Datenträger mit folgendem Inhalt:

- Programm Import PI CustomStage
- Kundenspezifische Positioniererdatenbank mit dem Parametersatz für den Positionierer

Damit der Parametersatz in der PC-Software ausgewählt werden kann, muss er zuvor mit dem Programm Import PI Custom Stage in die Positioniererdatenbank PIStages3 eingefügt werden.

➢ Installieren Sie die kundenspezifische Positioniererdatenbank, indem Sie im Hauptverzeichnis des Datenträgers auf die Datei ***Import_PI_CustomStage.exe*** doppelklicken.

Der Parametersatz aus der kundenspezifischen Positioniererdatenbank wird in PIStages3 eingefügt.

Wenn eine Meldung erscheint, dass die Installation der kundenspezifischen Positioniererdatenbank fehlgeschlagen ist:

a) Aktualisieren Sie die Positioniererdatenbank PIStages3 auf Ihrem PC, siehe "Updates installieren" (S. 59).

b) Wiederholen Sie die Installation der kundenspezifischen Positioniererdatenbank.

5.3 Belüftung sicherstellen

Hohe Temperaturen können den C-887 überhitzen.

- Installieren Sie den C-887 mit einem Abstand von mindestens 10 cm zur Oberseite und mindestens 5 cm zu dessen Seiten. Wenn dies nicht möglich ist, kühlen Sie die Umgebung ausreichend.
- Sorgen Sie für ausreichende Belüftung am Aufstellungsort.
- Halten Sie die Umgebungstemperatur auf einem unkritischen Wert (zwischen 5 °C und 40 °C).

5.4 C-887 erden

Der C-887 ist nicht über den Netzteilanschluss geerdet.

- Schließen Sie den mit dem Schutzleitersymbol gekennzeichneten Gewindestift (siehe Abbildung) am Gehäuse des C-887 an den Schutzleiter an.

5.5 C-887 an die Stromversorgung anschließen

INFORMATION

Wenn der Hexapod an der Buchse **24 V Out 7 A** des Controllers angeschlossen ist, wird die Versorgungsspannung des Controllers auch für den Hexapod verwendet.

Nur Modelle C-887.522, .523, .532, .533:

Die Buchse **E-Stop** steuert ein internes Relais mit Schließer, das den 24-V-Ausgang für den Hexapod (Buchse **24 V Out 7 A**) deaktiviert oder aktiviert.

Voraussetzungen

- ✓ Der C-887 ist ausgeschaltet, d. h. der Ein-/Ausschalter befindet sich in der Stellung O.
- ✓ Der C-887 ist in der Nähe der Stromversorgung installiert, damit der Netzstecker schnell und einfach vom Netz getrennt werden kann.

Werkzeug und Zubehör

- Netzteil: Mitgeliefertes 24-V-Weitbereichsnetzteil (für Netzspannungen zwischen 100 und 240 Volt Wechselspannung bei 50 oder 60 Hz)
- Mitgeliefertes Netzkabel
- Alternativ: ausreichend bemessenes Netzkabel

C-887 an die Stromversorgung anschließen

1. Verbinden Sie den Stecker M12 (w) des Netzteils mit dem Anschluss **24 V In 8 A** des C-887.
2. Verbinden Sie das Netzkabel mit dem Netzteil.
3. Schließen Sie das Netzteil über das Netzkabel an die Steckdose an.

5.6 Hexapod installieren

5.6.1 Zulässige Posen ermitteln

Um Schäden am Hexapod zu vermeiden, dürfen nur zulässige Posen kommandiert werden. Für die Ermittlung zulässiger Posen gemäß der folgenden Definition ist das Simulationsprogramm PIVirtualMove vorgesehen:

Eine Pose ist zulässig, wenn sie mit dem verwendeten Systemaufbau (Einbaulage des Hexapods, zu bewegende Last, externe Kräfte und Momente) und dem verwendeten Koordinatensystem von allen der sechs Achsen X, Y, Z, U, V, W erreicht werden kann **und** die zulässige Belastung der Beine **nicht** überschritten wird.

Werkzeug und Zubehör

- PC mit Windows-Betriebssystem, auf dem das Simulationsprogramm PIVirtualMove installiert ist.

Zulässige Posen des Hexapods ermitteln

➢ Folgen Sie den Anweisungen im PIVirtualMove-Handbuch (SM163E).

INFORMATION

Das Simulationsprogramm PIVirtualMove berechnet die zulässigen Posen auf Basis der maximalen Nutzlast des Hexapods (= Grenzwert bei eingeschaltetem Servomodus). Die maximale Haltekraft bei ausgeschaltetem Servomodus basiert auf der Selbsthemmung der Aktoren in den Hexapod-Beinen und fällt geringer aus als die maximale Nutzlast.

INFORMATION

Die Belastung der Hexapod-Beine variiert in Abhängigkeit von folgenden Faktoren:

- Aktivierungszustand des Servomodus im Controller
- Einbaulage des Hexapods
- Zu bewegende Last: Masse und Position des Massenschwerpunkts auf der Bewegungsplattform
- An der Bewegungsplattform angreifende Kräfte und Momente
- Von der Bewegungsplattform im Betrieb anzufahrende Posen (Translations- und Rotationskoordinaten)

5.6.2 Hexapod erden

➢ Folgen Sie den Anweisungen im Handbuch des Hexapods.

5.6.3 Hexapod auf Unterlage befestigen

➢ Folgen Sie den Anweisungen im Handbuch des Hexapods.

5.6.4 Last auf Hexapod befestigen

➢ Folgen Sie den Anweisungen im Handbuch des Hexapods.

5.7 Hexapod über Kabelsatz an C-887 anschließen

INFORMATION

Wenn der Hexapod an der Buchse **24 V Out 7 A** des Controllers angeschlossen ist, wird die Versorgungsspannung des Controllers auch für den Hexapod verwendet.

Nur Modelle C-887.522, .523, .532, .533:

Die Buchse **E-Stop** steuert ein internes Relais mit Schließer, das den 24-V-Ausgang für den Hexapod (Buchse **24 V Out 7 A**) deaktiviert oder aktiviert.

INFORMATION

Wenn eine Sicherheitsfunktion gemäß geltenden Normen erforderlich ist:

➢ Definieren und implementieren Sie geeignete Maßnahmen. Beispiele für mögliche Maßnahmen:

- Wenn die Stromversorgung des Hexapods über den C-887 erfolgt, unterbrechen Sie die Versorgungsleitung durch geeignete Sicherheitstechnik.
- Verwenden Sie für den Hexapod ein externes Netzteil mit geeigneter Sicherheitstechnik.
- Nur Modelle C-887.522, .523, .532, .533: Schließen Sie an der Buchse **E-Stop** geeignete Sicherheitstechnik an. Weitere Informationen siehe "Buchse E-Stop verwenden" (S. 93).

INFORMATION

Ab der Seriennummer 121017873 unterstützt der C-887 auch Hexapod-Mikroroboter, die zur Datenübertragung das BiSS-Protokoll verwenden.

Hexapod über Kabelsatz an C-887 anschließen

- ➢ Folgen Sie den Anweisungen im Handbuch des Hexapods.
- ➢ Nur Modelle C-887.522, .523, .532, .533: Beschalten Sie die Buchse **E-Stop** so, dass Bewegungen der Achsen möglich sind. Details siehe "Buchse E-Stop verwenden" (S. 93).

5.8 Positionierer für Achsen A und B anschließen

HINWEIS

Schäden bei Anschluss eines falschen Motors!

Das Anschließen eines Positionierers mit Schrittmotor kann irreparable Schäden am C-887 verursachen.

- ➢ Schließen Sie an die Buchsen **A** und **B** des C-887 nur Positionierer mit DC-Motor und integrierten Motortreibern an.

INFORMATION

Die Buchsen **Motor A** und **Motor B** (Sub-D 15 (f)) des C-887 sind für den Anschluss von Positionierern mit DC-Motor und integrierten Motortreibern vorgesehen.

Wenn Sie PI vor Auslieferung des Hexapod-Systems über die verwendeten Positionierertypen informieren, konfiguriert PI den C-887 entsprechend Ihrer Bestellung, so dass die entsprechenden Positionierertypen zu den Achsen A und B des C-887 zugewiesen sind:

- Wenn Sie nur einen Positionierer bestellen, wird der entsprechende Positionierertyp der Achse A zugewiesen.
- Wenn Sie zwei Positionierer bestellen, werden die entsprechenden Positionierertypen den Achsen A und B in aufsteigender alphabetischer Reihenfolge zugewiesen, z. B. M-403.1DG zu A und M-403.2PD zu B; M-414.1PD zu A und M-511.DD1 zu B.

Weitere Informationen zur Zuweisung von Positionierertypen finden Sie in "Betriebsparameter der Achsen A und B" (S. 32).

Voraussetzungen

- ✓ Der C-887 ist ausgeschaltet, d. h. der Ein-/Ausschalter befindet sich in der Stellung ○.
- ✓ Sie haben das Benutzerhandbuch des Positionierers gelesen und verstanden.
- ✓ Sie haben den Positionierer gemäß der Beschreibung im entsprechenden Benutzerhandbuch montiert.

Werkzeug und Zubehör

- Positionierer von PI, der mit einem PWM-Verstärker ausgestattet ist, erhältlich als optionales Zubehör (S. 24)

Positionierer für Achsen A und B an C-887 anschließen

1. Schließen Sie die Positionierer an den Buchsen **Motor A** und **Motor B** an.
   - Wenn bekannt, beachten Sie die Einstellungen des C-887 für die Positionierertyp-Zuweisung.

   Der Positionierer an **Motor A** wird als Achse A kommandiert, und der Positionierer an **Motor B** wird als Achse B kommandiert.
2. Sichern Sie die Steckverbindungen mit den integrierten Schrauben gegen unbeabsichtigtes Abziehen.

5.9 Analoge Signalquellen anschließen

Alle C-887-Modelle sind mit vier analogen Eingängen auf der Buchse **I/O** (S. 381) ausgestattet.

Die Modelle C-887.521, .523, .531 und .533 sind zusätzlich mit zwei hochauflösenden analogen Eingängen auf den BNC-Buchsen **Analog In 5** und **Analog In 6** ausgestattet.

Details siehe "Spezifikationen der analogen Eingänge" (S. 376).

Anwendungsmöglichkeiten der analogen Eingangssignale:

- Verwendung in Fast-Alignment-Routinen (S. 2). Weitere Informationen finden Sie im Dokument "Schnelle, mehrkanalige Ausrichtung in der Photonik" (E712T0016).
- Verwendung in Scanprozeduren:
  - Vom C-887 ausgeführte Scanprozeduren: siehe die Befehle AAP (S. 158), FIO (S. 177), FLM (S. 180), FLS (S. 182), FSA (S. 187), FSC (S. 190), FSM (S. 195)
  - Von PIMikroMove® ausgeführte Scanprozeduren (im ***Tools***-Menü von PIMikroMove® verfügbar): siehe PIMikroMove®-Handbuch
- Verwendung in Makros:
  - Bedingte Ausführung des Makros: siehe WAC (S. 281)
  - Bedingtes Stoppen der Makroausführung: siehe MEX (S. 241)
  - Bedingter Sprung des Makroausführungszeigers: siehe JRC (S. 210)
  - Kopieren des Eingangszustands in eine Variable: siehe CPY (S. 163)

  Details und Beispiele zu Makros finden Sie in "Controllermakros" (S. 127).

INFORMATION

Für dynamische und hochauflösende Anwendungen wie zum Beispiel Fast-Alignment-Routinen oder Scanprozeduren wird die Verwendung der BNC-Buchsen **Analog In 5** und **Analog In 6** empfohlen.
Die analogen Eingänge auf der Buchse **I/O** eignen sich für die Verwendung in Makros.

Werkzeug und Zubehör

- Geeignete Signalquelle gemäß den Spezifikationen des Eingangs (S. 376), z. B. optischer Leistungsmesser F-712.PM1 (erhältlich als optionales Zubehör (S. 24)) für den Anschluss an **Analog In 5** oder **Analog In 6**

Analoge Signalquelle anschließen

1. Schließen Sie die Signalquelle am analogen Eingang an.
2. Sichern Sie die Steckverbindung gegen unbeabsichtigtes Abziehen.

5.10 Digitale Ein- und Ausgänge anschließen

Die digitalen Ein- und Ausgänge auf der Buchse **I/O** (S. 381) des C-887 (TTL-Signale) können wie folgt verwendet werden:

- Ausgänge: Status setzen mit dem Befehl DIO (S. 166)
- Eingänge: Status abfragen mit dem Befehl DIO? (S. 167), Verwendung in Makros:
  - Bedingte Ausführung des Makros: siehe WAC (S. 281)
  - Bedingtes Stoppen der Makroausführung: siehe MEX (S. 241)
  - Bedingter Sprung des Makroausführungszeigers: siehe JRC (S. 210)
  - Kopieren des Eingangszustands in eine Variable: siehe CPY (S. 163)

  Details und Beispiele zu Makros finden Sie in "Controllermakros" (S. 127).

Werkzeug und Zubehör

- Geeigneter Stecker HD D-Sub 26 (m)
- Wenn Sie die digitalen Eingänge verwenden wollen: Geeignete Signalquelle gemäß den Spezifikationen des Eingangs (S. 381)
- Wenn Sie die digitalen Ausgänge verwenden wollen: Geeignetes Gerät für die Signalauswertung gemäß den Spezifikationen des Ausgangs (S. 381)

Digitale Ein- und Ausgänge anschließen

- ➢ Digitale Eingänge: Schließen Sie die Signalquelle an einen der Pins 7, 16, 17 und 25 der Buchse **I/O** an.
- ➢ Digitale Ausgänge: Schließen Sie das Gerät für die Signalauswertung an einen der Pins 8, 9, 18 und 26 der Buchse **I/O** an.
- ➢ Sichern Sie die Steckverbindung gegen unbeabsichtigtes Abziehen.

5.11 PC anschließen

Die Kommunikation zwischen dem C-887 und einem PC kann zur Konfiguration des C-887 und zur Bewegungskommandierung mit den Befehlen des GCS genutzt werden. Der C-887 verfügt dazu über folgende Schnittstellen:

- TCP/IP
- RS-232-Schnittstelle

In diesem Abschnitt erfahren Sie, wie Sie den C-887 über die entsprechenden Kabel an den PC anschließen. Alle weiteren Schritte, die für die Herstellung der Kommunikation zwischen C-887 und PC erforderlich sind, finden Sie in "Kommunikation über TCP/IP-Schnittstelle herstellen" (S. 74) und in "Kommunikation über RS-232-Schnittstelle herstellen" (S. 82).

5.11.1 C-887 über die TCP/IP-Schnittstelle anschließen

Voraussetzungen

✓ Wenn der C-887 direkt an den PC angeschlossen werden soll:
Der PC verfügt über eine freie RJ45-Ethernet-Anschlussbuchse.

✓ Wenn C-887 und PC gemeinsam in einem Netzwerk betrieben werden sollen:
Für den C-887 ist ein freier Zugangspunkt zum Netzwerk vorhanden, gegebenenfalls ist dazu ein geeigneter Hub oder Switch an das Netzwerk angeschlossen.

Werkzeug und Zubehör

- Wenn der C-887 direkt an den PC angeschlossen werden soll:
Crossover-Netzwerkkabel (C-815.563 im Lieferumfang)
- Wenn der C-887 an einen Netzwerk-Zugangspunkt angeschlossen werden soll: Straight-Through-Netzwerkkabel (C-815.553 im Lieferumfang)

C-887 direkt an den PC anschließen

➢ Verbinden Sie die RJ45-Buchse des C-887 über das Crossover-Netzwerkkabel mit der RJ45-Ethernet-Anschlussbuchse des PC.

C-887 an das Netzwerk anschließen, in dem sich auch der PC befindet

➢ Verbinden Sie die RJ45-Buchse des C-887 über das Straight-Through-Netzwerkkabel mit dem Netzwerk-Zugangspunkt.

5.11.2 C-887 über die RS-232-Schnittstelle anschließen

Voraussetzungen

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

Werkzeug und Zubehör

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

C-887 an den PC anschließen

➢ Verbinden Sie den Einbaustecker **RS-232** des C-887 und die RS-232-Schnittstelle des PC (ein Einbaustecker D-Sub 9 (m)) mit dem Nullmodemkabel.

5.12 EtherCAT-Master anschließen

Die Modelle C-887.53x (.53x steht für .53, .531, .532 und .533) sind mit einer EtherCAT-Schnittstelle ausgestattet.

Weitere Informationen zur Verwendung der EtherCAT-Schnittstelle finden Sie im Dokument "EtherCAT Interface Description" (C887T0011).

HINWEIS

Fehlfunktion wegen falschem Anschluss!

Falsch angeschlossene Kabel können zu falscher Adressierung und Kommunikationsstörungen führen.

➢ Verwenden Sie die RJ45-Buchse **Port 1** (links), um den C-887.53x mit dem EtherCAT-Master zu verbinden (die RJ45-Buchse **Port 2** (rechts) ist für den Anschluss des nächsten EtherCAT-Slaves vorgesehen).

➢ Verwenden Sie EtherCAT und Standard-Ethernet **nicht** zusammen in einem physikalischen Netzwerk. Wenn möglich, verwenden Sie Kabel mit unterschiedlichen Farben für EtherCAT-Verbindungen und Standard-Ethernet-Verbindungen.

Werkzeug und Zubehör

- Geeignetes Kabel:
  - Patchkabel CAT 5 oder höher, Straight-Through oder Crossover
  - Kabellänge: 0,3 bis 100 m

EtherCAT-Master anschließen

➢ Verbinden Sie den EtherCAT-Master über ein geeignetes Kabel mit der RJ45-Buchse **Port 1** des C-887.53x.

6 Inbetriebnahme

In diesem Kapitel

Allgemeine Hinweise zur Inbetriebnahme 69
C-887 einschalten 74
Kommunikation über TCP/IP-Schnittstelle herstellen 74
Kommunikation über RS-232-Schnittstelle herstellen 82
Bewegungen starten 84

6.1 Allgemeine Hinweise zur Inbetriebnahme

VORSICHT

Quetschgefahr durch bewegte Teile!

Zwischen den bewegten Teilen des Hexapods und einem feststehenden Teil oder Hindernis besteht die Gefahr von leichten Verletzungen durch Quetschung.

- Halten Sie Ihre Finger von Bereichen fern, in denen sie von bewegten Teilen erfasst werden können.

HINWEIS

Falsche Konfiguration des Controllers!

Die vom Controller verwendeten Konfigurationsdaten (z. B. Geometriedaten und Regelparameter) müssen auf den Hexapod abgestimmt sein. Bei Verwendung falscher Konfigurationsdaten kann der Hexapod durch unkontrollierte Bewegungen oder Kollisionen beschädigt werden.

Eine Anpassung der Konfigurationsdaten findet beim Einschalten oder Neustart des Controllers anhand der vom ID-Chip geladenen Daten statt (S. 31).

- Nachdem Sie die Kommunikation über TCP/IP (S. 74) oder RS-232 (S. 82) hergestellt haben, senden Sie den Befehl `CST?`. Die Antwort zeigt an, auf welchen Hexapod der Controller abgestimmt ist.
- Betreiben Sie den Hexapod nur mit einem Controller, dessen Konfigurationsdaten auf den Hexapod abgestimmt sind.

HINWEIS

Zuweisung eines falschen Positionierertyps!

Wenn den Achsen A und B ein falscher Positionierertyp zugewiesen wird, können unpassende Parametereinstellungen geladen werden. Unpassende Parametereinstellungen können bei Inbetriebnahme und Betrieb zu Schäden am Positionierer führen.

- Stellen Sie sicher, dass der zugewiesene Positionierertyp mit dem angeschlossenen Positionierer übereinstimmt.
- Ändern Sie die Parameterwerte für die Achsen A und B nur nach sorgfältiger Überlegung.

HINWEIS

Schäden durch Kollisionen!

Kollisionen können den Hexapod, die zu bewegende Last und die Umgebung beschädigen.

- Stellen Sie sicher, dass im Arbeitsraum des Hexapods keine Kollisionen zwischen Hexapod, zu bewegender Last und Umgebung möglich sind.
- Platzieren Sie keine Gegenstände in Bereichen, in denen sie von bewegten Teilen erfasst werden können.
- Kommandieren Sie nur zulässige Posen. Für die Definition einer zulässigen Pose siehe "Zulässige Posen ermitteln" (S. 62).
- Halten Sie bei einer Fehlfunktion des Controllers die Bewegung sofort an.

HINWEIS

Schäden durch nicht entfernte Transportsicherung!

Wenn die Transportsicherung (S. 55) des Hexapods nicht entfernt wurde und eine Bewegung kommandiert wird, können Schäden am Hexapod entstehen.

- Entfernen Sie die Transportsicherung, bevor Sie das Hexapod-System in Betrieb nehmen.

HINWEIS

Schäden durch ungewollte Positionsänderungen!

Das Simulationsprogramm PIVirtualMove berechnet die zulässigen Posen auf Basis der maximalen Nutzlast des Hexapods (= Grenzwert bei eingeschaltetem Servomodus). Die maximale Haltekraft bei ausgeschaltetem Servomodus basiert auf der Selbsthemmung der Aktoren in den Hexapod-Beinen und fällt geringer aus als die maximale Nutzlast.

Wenn die tatsächliche Belastung des Hexapods die auf der Selbsthemmung der Aktoren basierende maximale Haltekraft überschreitet, können in folgenden Fällen ungewollte Positionsänderungen des Hexapods auftreten, obwohl PIVirtualMove die aktuelle Pose als zulässig eingestuft hat:

- Ausschalten des C-887
- Neustart des C-887
- Ausschalten des Servomodus für die Achsen der Bewegungsplattform des Hexapods, z. B. durch Verwendung der Buchse **E-Stop** (S. 93)

Dadurch sind Kollisionen zwischen Hexapod, zu bewegender Last und Umgebung möglich. Kollisionen können den Hexapod, die zu bewegende Last oder die Umgebung beschädigen.

➢ Stellen Sie sicher, dass die tatsächliche Belastung der Bewegungsplattform des Hexapods die auf der Selbsthemmung der Aktoren basierende maximale Haltekraft nicht überschreitet, bevor Sie den Servomodus ausschalten, den C-887 neu starten oder ausschalten.

HINWEIS

Schäden durch Kollisionen während der Referenzierungsfahrt!

Während einer Referenzierungsfahrt bewegt sich der Hexapod auf unvorhersehbare Weise. Es findet **keine** Kollisionsprüfung und -vermeidung statt, selbst wenn eine Konfiguration zur Kollisionsvermeidung auf dem C-887 abgelegt wurde (z. B. mit der kostenpflichtigen PIVeriMove Hexapod Software zur Kollisionsprüfung). Verfahrbereichsgrenzen, die mit den Befehlen NLM und PLM für die Bewegungsplattform des Hexapods gesetzt wurden, werden während der Referenzierungsfahrt ignoriert.

Dadurch sind Kollisionen zwischen Hexapod, zu bewegender Last und Umgebung möglich. Kollisionen können den Hexapod, die zu bewegende Last und die Umgebung beschädigen.

➢ Stellen Sie sicher, dass während der Referenzierungsfahrt des Hexapods keine Kollisionen zwischen Hexapod, zu bewegender Last und Umgebung möglich sind.

➢ Platzieren Sie keine Gegenstände in Bereichen, in denen sie während der Referenzierungsfahrt von bewegten Teilen erfasst werden können.

➢ Kommandieren Sie nach erfolgreicher Referenzierungsfahrt die entsprechende Zielposition, um von einer beliebigen Position zur Referenzposition (Standard: Nullposition) zurückzukehren. Starten Sie **nicht** eine erneute Referenzierungsfahrt.

HINWEIS

Schäden durch unkontrollierte Bewegung des Hexapods!

Geschwindigkeit und Beschleunigung der Bewegungsplattform des Hexapods werden in folgenden Fällen **nicht** vom C-887 vorgegeben:

- Zyklische Übertragung von Zielpositionen (S. 39)
- Der Hexapod (Achsen X, Y, Z, U, V, W) bewegt sich noch, während ein neuer Bewegungsbefehl gesendet wird. Die bisherige Zielposition wird dabei überschrieben, ohne dass Geschwindigkeit und Beschleunigung der Bewegungsplattform des Hexapods neu berechnet werden.

Die Plattform des Hexapods bewegt sich dann auf einer undefinierten Bahn. Auf dieser undefinierten Bahn sind Kollisionen mit der Umgebung des Hexapods möglich. Kollisionen können den Hexapod, die zu bewegende Last und die Umgebung beschädigen.

Wenn Sie Bewegungen durch zyklische Übertragung von Zielpositionen auslösen:

- ➢ Setzen Sie nur Zielpositionen, deren Abstand zueinander maximal so groß wie der Wert des Parameters ***Path Control Step Size*** (ID 0x19001504) ist.

Wenn Sie Punkt-zu-Punkt-Bewegungen mit Bewegungsbefehlen kommandieren (S. 34):

- ➢ Vermeiden Sie das Senden neuer Zielpositionen, wenn sich der Hexapod (Achsen X, Y, Z, U, V, W) noch bewegt.
- ➢ Wenn neue Zielpositionen gesendet werden müssen, während sich der Hexapod noch bewegt (Achsen X, Y, Z, U, V, W): Setzen Sie mit Bewegungsbefehlen nur Zielpositionen, die von der aktuellen Position maximal um den Wert des Parameters ***Path Control Step Size*** (ID 0x19001504) abweichen.

INFORMATION

Anwendungshinweise für die ID-Chip-Erkennung:

- ➢ Bevor Sie den angeschlossenen Hexapod austauschen, sichern Sie die aktuellen Parameterwerte des Controllers auf dem PC (S. 319).
- ➢ Schließen Sie den Hexapod nur an den ausgeschalteten Controller an.
- ➢ Wenn das Booten der Firmware abgeschlossen ist, senden Sie den Befehl `CST?` (S. 164), um zu prüfen, ob die installierte Konfiguration durch einen Neustart des Controllers aktiviert werden muss. Ein Neustart ist erforderlich, wenn die Antwort "NOSTAGE" lautet. Der Controller kann mit dem Befehl `RBT` (S. 253) neu gestartet werden.
- ➢ Senden Sie den Befehl `ERR?` (S. 176), um zu prüfen, ob die Konfiguration erfolgreich aktiviert wurde. Wenn die Antwort auf ERR? den Fehlercode 233 oder 211 enthält, fehlt im Controller die Konfigurationsdatei für den neuen Hexapod (möglich z. B. für kundenspezifische Hexapoden oder neue Standard-Hexapoden). Kontaktieren Sie unseren Kundendienst (S. 367), um eine geeignete Konfigurationsdatei zu erhalten. Für die Installation der neuen Konfigurationsdatei siehe "Firmware und Konfigurationsdateien aktualisieren" (S. 340).
- ➢ Senden Sie den Befehl `VER?` (S. 278), um die auf dem ID-Chip gespeicherten Angaben für Hexapod-Typ, Seriennummer und Herstelldatum zu prüfen. Beispiel für die Antwort: `IDChip: H-811.F-2 SN123456789 20/1/2016`

INFORMATION

Die Kommunikation zwischen dem C-887 und einem PC kann zur Konfiguration des C-887 und zur Bewegungskommandierung mit den Befehlen des GCS genutzt werden.

- Über die RS-232-Schnittstelle ist die Kommunikation ohne weitere Einstellungen möglich.
- Für die Kommunikation über TCP/IP kann einmalig die Anpassung der Schnittstellenparameter erforderlich sein (S. 75).

INFORMATION

Die Kommunikationsschnittstellen des C-887 (TCP/IP, RS-232) sind gleichzeitig aktiv. Befehle werden in der Reihenfolge abgearbeitet, in der die kompletten Befehlszeilen eintreffen. Die gleichzeitige Verwendung mehrerer Kommunikationsschnittstellen kann jedoch Probleme mit der PC-Software verursachen.

➢ Verwenden Sie immer nur eine Schnittstelle des C-887.

INFORMATION

Beim Einschalten oder Neustart schaltet der C-887 automatisch den Servomodus für alle Achsen ein. Wenn der Servomodus für die Achsen der Bewegungsplattform des Hexapods (X, Y, Z, U, V, W) ausgeschaltet ist, wird er beim Starten der Referenzierungsfahrt automatisch eingeschaltet.

INFORMATION

Für Achsen mit inkrementellen Sensoren können Bewegungen erst nach einer erfolgreichen Referenzierungsfahrt kommandiert werden (S. 185) (auch als "Initialisierung" bezeichnet).

Das Verhalten der Achsen des Hexapods nach der Referenzierungsfahrt wird durch die Parameter ***Behaviour After Reference Move*** (ID 0x07030401) und ***Target For Motion After Reference Move*** (ID 0x07030402) festgelegt. Entsprechend der Parameterwerte können die Achsen der Plattform z. B. nach der Referenzierungsfahrt automatisch zu einer vorgegebenen Position bewegt werden.

- Wert des Parameters 0x07030401 = 0: Achse bleibt nach der Referenzierungsfahrt in der Referenzposition.
- Wert des Parameters 0x07030401 = 1: Achse fährt nach der Referenzierungsfahrt zur absoluten Zielposition, die durch Parameter 0x07030402 vorgegeben ist.

Für Achsen mit absolut messenden Sensoren ist keine Referenzierungsfahrt erforderlich. Die Verwendung des Befehls FRF wird für diese Achsen trotzdem empfohlen. FRF startet für Achsen mit absolut messenden Sensoren **keine** Referenzierungsfahrt, sondern setzt die Zielpositionen auf die aktuellen Positionswerte. Zusätzlich werden die oben beschriebenen Parameterwerte wirksam, so dass die Achsen z. B. zu einer definierten "Ausgangsposition" bewegt werden können.

6.2 C-887 einschalten

Voraussetzungen

- ✓ Sie haben die allgemeinen Hinweise zur Inbetriebnahme gelesen und verstanden (S. 69).
- ✓ Der C-887 wurde ordnungsgemäß installiert (S. 57).

C-887 einschalten

- ➢ Bringen Sie den Kippschalter an der Vorderwand des C-887 in die Stellung | .

  Der C-887 startet das Betriebssystem und die Firmware. Der Startvorgang dauert circa 40 Sekunden, Beginn und Ende werden jeweils durch einen Signalton angezeigt. Nach dem Ende des Startvorgangs leuchten die LEDs **PWR** und **STA**.

  Während des Startvorgangs führt der C-887 unter anderem folgende Aktionen aus:

  – Einschalten des Servomodus für die Achsen der Bewegungsplattform des Hexapods

  – Aktivieren der im permanenten Speicher hinterlegten Einstellungen

  – Wenn vorhanden: Starten des Startup-Makros

6.3 Kommunikation über TCP/IP-Schnittstelle herstellen

Anpassung der Schnittstellenparameter

Vor der Herstellung der Kommunikation kann einmalig die Anpassung der werkseitigen Einstellungen der Schnittstellenparameter erforderlich sein. Mit dem Befehl `IFS` (S. 207) können im permanenten Speicher des C-887 folgende Schnittstellenparameter für die TCP/IP-Kommunikation angepasst werden:

Schnittstellen-parameter	Werkseitige Einstellung	Bemerkung
Standard-IP-Adresse (IPADR)	192.168.1.28:50000	Ermöglicht die Definition einer statischen (d. h. festen) Adresse. Diese statische Adresse wird **nicht** verwendet, wenn der C-887 für die Zuweisung einer IP-Adresse durch einen DHCP-Server oder AutoIP konfiguriert ist (werkseitige Einstellung des Startverhaltens zur Konfiguration der IP-Adresse). Wenn die statische Adresse verwendet werden soll: ▪ Das Startverhalten muss umgestellt werden, damit der C-887 die mit "IPADR" definierte IP-Adresse verwendet. ▪ Die IP-Adressen und Subnetzmasken von C-887 und PC bzw. allen restlichen Netzwerkteilnehmern müssen aufeinander abgestimmt werden. Details siehe "PC und C-887 für Verwendung statischer IP-Adressen vorbereiten" (S. 75).

Schnittstellen-parameter	Werkseitige Einstellung	Bemerkung
Startverhalten zur Konfiguration der IP-Adresse für die TCP/IP-Kommunikation (IPSTART)	DHCP oder AutoIP wird verwendet, um die IP-Adresse zu erhalten	Die IP-Adresse des C-887 wird mit der werkseitigen Einstellung des Startverhaltens über DHCP zugewiesen oder mit AutoIP automatisch konfiguriert. Die werkseitige Einstellung des Startverhaltens muss nur geändert werden, wenn die Netzwerkteilnehmer stattdessen statische Adressen verwenden sollen.
Subnetzmaske (IPMASK)	255.255.255.0	Die werkseitige Einstellung der Subnetzmaske muss eventuell geändert werden, wenn die Netzwerkteilnehmer statische Adressen verwenden sollen. Details siehe "PC und C-887 für Verwendung statischer IP-Adressen vorbereiten" (S. 75).

Nach dem Einschalten oder dem Neustart des C-887

Der Startvorgang des C-887 muss beendet sein, bevor die Kommunikation zwischen C-887 und PC hergestellt werden kann. Der Startvorgang dauert circa 40 Sekunden (zweiter Signalton zeigt das Ende an).

Wenn die IP-Adresse des C-887 über DHCP zugewiesen wird oder die IP-Adressen der Netzwerkteilnehmer mit AutoIP konfiguriert werden, dauert es nach dem Ende des Startvorgangs des C-887 (S. 74) bis zu 2 Minuten, bis die Kommunikation über TCP/IP möglich ist.

Anschluss des Netzwerkkabels bei eingeschaltetem Controller

Die Herstellung der Kommunikation über TCP/IP kann fehlschlagen, wenn das Netzwerkkabel bei eingeschaltetem C-887 an die RJ45-Buchse des C-887 angeschlossen wurde.

- ➢ Wenn die Herstellung der Kommunikation fehlschlägt, schalten Sie den C-887 aus und bei gestecktem Netzwerkkabel wieder ein.

Porteinstellung

Für die Kommunikation mit GCS-Befehlen über TCP/IP steht beim C-887 ein unveränderlicher Port (50000) zur Verfügung.

6.3.1 PC und C-887 für Verwendung statischer IP-Adressen vorbereiten

Wenn ein Netzwerk ohne DHCP-Server vorliegt oder wenn der C-887 direkt an die Ethernet-Anschlussbuchse des PC angeschlossen ist **und** statische IP-Adressen verwendet werden sollen, sind folgende Anpassungen der Schnittstellenparameter notwendig:

- Startverhalten zur Konfiguration der IP-Adresse des C-887 so einstellen, dass eine statische Adresse verwendet wird

- IP-Adressen und Subnetzmasken von C-887 und PC bzw. allen restlichen Netzwerkteilnehmern aufeinander abstimmen

Zur Anpassung von IP-Adressen und Subnetzmasken können Sie sich für eine der beiden folgenden Optionen entscheiden:

- Die PC-Einstellungen und gegebenenfalls die Einstellungen weiterer Netzwerkteilnehmer anpassen. Die Einstellungen des C-887 bleiben unverändert.
- Die Einstellungen des C-887 anpassen. Die PC-Einstellungen und gegebenenfalls die Einstellungen weiterer Netzwerkteilnehmer bleiben unverändert.

Voraussetzungen

✓ Sie haben die Kommunikation zwischen dem C-887 und dem PC über RS-232 hergestellt, um die Einstellungen des C-887 zu ermitteln und gegebenenfalls ändern zu können (S. 82).

IP-Adresse und Subnetzmaske des PC ermitteln

1. Öffnen Sie an Ihrem PC auf geeignete Weise das Fenster, in dem die Eigenschaften des Internetprotokolls TCP/IP angezeigt und eingestellt werden. Die erforderlichen Schritte hängen vom verwendeten Betriebssystem ab.

   Wenn Ihr Betriebssystem zwischen Internet Protocol Version 4 (IPv4) und Internet Protocol Version 6 (IPv6) unterscheidet (z. B. Windows 10), öffnen Sie das Fenster für Version 4.

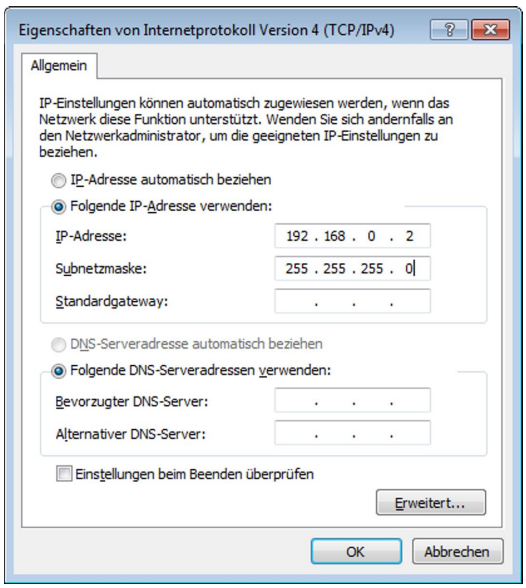


Abbildung 8: Fenster "Eigenschaften von Internet Protocol (TCP/IP)" mit beispielhaften Einstellungen (nicht unbedingt für Ihr System geeignet)

   Die Abbildung zeigt beispielhafte Einstellungen, die nicht unbedingt für Ihr System geeignet sind.

2. Notieren Sie die Einstellungen.

IP-Adresse und Subnetzmaske des C-887 ermitteln, Startverhalten des C-887 anpassen

1. Wenn Sie die Kommunikation zwischen C-887 und PC über die RS-232-Schnittstelle hergestellt haben, öffnen Sie im verwendeten Programm das Fenster zum Eingeben von Befehlen.
2. Geben Sie den Befehl `IFS?` ein.

   Dieser Befehl fragt die Werte der Schnittstellenparameter im permanenten Speicher ab.
3. Notieren Sie die Einstellungen für ***IPMASK*** und ***IPADR***.
4. Stellen Sie sicher, dass der Parameter ***IPSTART*** korrekt eingestellt ist:
   - Wenn ***IPSTART*** = 0 (die mit ***IPADR*** definierte statische IP-Adresse wird verwendet), ist die Einstellung korrekt.
   - Wenn ***IPSTART*** ≠ 0: Senden Sie den Befehl `IFS 100 IPSTART 0`.

INFORMATION

In PIMikroMove® können Sie im Fenster ***Configure Interface*** die IP-Adresse und die Subnetzmaske des C-887 ermitteln und das Startverhalten des C-887 anpassen, ohne Befehle senden zu müssen.

IP-Einstellungen des PC anpassen

➢ Wenn Sie die PC-Einstellungen unverändert lassen wollen, fahren Sie mit dem Abschnitt "C-887-Einstellungen anpassen" fort (S. 78).

1. Aktivieren Sie ***Folgende IP-Adresse verwenden*** in dem Fenster, in dem die Eigenschaften des Internetprotokolls TCP/IP (TCP/IPv4) angezeigt und eingestellt werden.
2. Passen Sie die IP-Adresse und die Subnetzmaske an die Einstellungen des C-887 an:
   a) Übernehmen Sie für die IP-Adresse am PC die ersten drei Abschnitte der IP-Adresse des C-887.
   b) Stellen Sie sicher, dass sich der letzte Abschnitt der IP-Adresse am PC vom letzten Abschnitt der IP-Adresse des C-887 unterscheidet und nicht "255" oder "0" ist.
   c) Übernehmen Sie für die Subnetzmaske am PC die Subnetzmaske des C-887.

   Beispiel:
   Neue IP-Adresse des PC: 192.168.1.29 (wenn der C-887 die IP-Adresse 192.168.1.28 hat)
   Neue Subnetzmaske des PC: 255.255.255.0 (wenn der C-887 die Subnetzmaske 255.255.255.0 hat)
3. Bestätigen Sie die Einstellungen mit der Schaltfläche ***OK***.
4. Wenn noch weitere Netzwerkteilnehmer angepasst werden müssen:
   Passen Sie die IP-Adressen und Subnetzmasken wie in den vorhergehenden Schritten an.

Weisen Sie jedem Netzwerkteilnehmer eine eigene, eindeutige IP-Adresse zu. IP-Adressen dürfen im selben Netzwerk nicht doppelt vorkommen.

5. Schließen Sie die Verbindung über die RS-232-Schnittstelle, z. B. in PIMikroMove® im Hauptfenster über den Menüeintrag ***Connections > Close > C-887***.
6. Schalten Sie den C-887 aus.
7. Fahren Sie mit dem Abschnitt "Kommunikation über TCP/IP in der PC-Software herstellen" fort (S. 79).

C-887-Einstellungen anpassen

1. Passen Sie mit dem Befehl `IFS` die Einstellungen des C-887 an die des PC an:
   a) Ändern Sie die Subnetzmaske mit dem Befehl `IFS 100 IPMASK xxx.xxx.xxx.xxx`, wobei *xxx.xxx.xxx.xxx* die Subnetzmaske des PC ist.
   b) Ändern Sie die IP-Adresse mit dem Befehl `IFS 100 IPADR xxx.xxx.xxx.yyy:50000`, wobei Folgendes gilt:
   – *xxx.xxx.xxx.* stimmt mit den ersten drei Abschnitten der IP-Adresse des PC überein
   – *yyy* unterscheidet sich vom letzten Abschnitt der IP-Adresse des PC und jedes anderen Geräts im gleichen Netzwerk
   – *yyy* ist nicht "255" und nicht "0" und liegt im Adressbereich, der durch den letzten Abschnitt der Subnetzmaske vorgegeben ist
   – die Portnummer "50000" darf nicht geändert werden

   Beispiel:
   Wenn die IP-Adresse des PC 192.168.0.1 ist und kein anderes Gerät die IP-Adresse 192.168.0.2 hat, senden Sie den Befehl `IFS 100 IPADR 192.168.0.2:50000`.
2. Schließen Sie die Verbindung über die RS-232-Schnittstelle, z. B. in PIMikroMove® im Hauptfenster über den Menüeintrag ***Connections > Close > C-887***.
3. Schalten Sie den C-887 aus.
4. Fahren Sie mit dem Abschnitt "Kommunikation über TCP/IP in der PC-Software herstellen" fort (S. 79).

6.3.2 Kommunikation über TCP/IP in der PC-Software herstellen

VORSICHT

Quetschgefahr durch unerwartete Bewegung

Wenn die Kommunikation zwischen C-887 und PC über TCP/IP hergestellt wird, bietet die PC-Software alle im selben Netzwerk vorhandenen Controller zur Auswahl an. Nach Auswahl eines C-887 für die Verbindung werden alle Befehle an diesen Controller geschickt. Bei Auswahl eines falschen Controllers besteht für das Bedien- und Wartungspersonal des angeschlossenen Hexapods die Gefahr von leichten Verletzungen durch Quetschung aufgrund von unerwartet kommandierten Bewegungen.

- ➢ Wenn in der PC-Software mehrere C-887 angezeigt werden, vergewissern Sie sich, dass Sie den richtigen C-887 auswählen.
- ➢ Vergeben Sie einen eindeutigen Namenszusatz für jeden C-887 im selben Netzwerk: Tragen Sie den Namenszusatz als Wert des Parameters ***Customer Device Name*** (ID 0x0D001000) ein.

INFORMATION

In der Auflistung der im selben Netzwerk gefundenen Controller wird der C-887 **nicht** als ***C-887*** angezeigt, sondern als ***HEXAPOD*** oder ***F-HEX***, gefolgt von der 9-stelligen Seriennummer des C-887.

Für den C-887 kann ein frei wählbarer Namenszusatz vergeben werden, um mehrere C-887 im selben Netzwerk oder am selben PC unterscheiden zu können. Gehen Sie wie folgt vor, um den Namenszusatz zu vergeben:

1. Stellen Sie die Kommunikation über eine der vorhandenen Schnittstellen her.
2. Wechseln Sie auf Befehlsebene 1, indem Sie senden:
   `CCL 1 advanced`
3. Tragen Sie einen eindeutigen Namenszusatz als Wert des Parameters ***Customer Device Name*** (ID 0x0D001000) ein, indem Sie senden:
   `SPA 1 0x0D001000` *Namenszusatz*
4. Speichern Sie den neuen Parameterwert im permanenten Speicher, indem Sie senden:
   `WPA 101 1 0x0D001000`.
5. Schließen Sie die Verbindung.

Alternativ:

- ➢ Identifizieren Sie den zu verbindenden C-887 in der Auflistung der gefundenen Controller anhand seiner Seriennummer (*SN*). Die Seriennummer des Controllers können Sie dem Typenschild auf der Rückwand des C-887 entnehmen.

Voraussetzungen

- ✓ Sie haben die allgemeinen Hinweise zur Inbetriebnahme gelesen und verstanden (S. 69).
- ✓ Der C-887 ist über die RJ45-Buchse an das Netzwerk oder direkt an den PC angeschlossen.
- ✓ Wenn der C-887 an ein Netzwerk angeschlossen ist:
  Der zur Kommunikation mit dem C-887 zu verwendende PC ist auf geeignete Weise an dasselbe Netzwerk wie der C-887 angeschlossen.
- ✓ Wenn das verwendete Netzwerk keinen DHCP-Server besitzt oder wenn der C-887 direkt an die Ethernet-Anschlussbuchse des PC angeschlossen ist **und** statische IP-Adressen verwendet werden sollen:
  Sie haben durch Anpassen der Schnittstellenparameter das passende Startverhalten zur Konfiguration der IP-Adresse des C-887 eingestellt und die IP-Adressen und Subnetzmasken von C-887 und PC bzw. allen restlichen Netzwerkteilnehmern aufeinander abgestimmt (S. 75).
- ✓ Wenn mehrere C-887 über ihre TCP/IP-Schnittstellen mit dem gleichen Netzwerk verbunden sind: Sie haben über den Parameter ***Customer Device Name*** (ID 0x0D001000) einen eindeutigen Namenszusatz für den C-887 vergeben, mit dem die Kommunikation hergestellt werden soll. Alternativ haben Sie die Seriennummer dieses C-887 parat. Die Seriennummer können Sie dem Typenschild auf der Rückwand des C-887 entnehmen.
- ✓ Der PC ist eingeschaltet.
- ✓ Die benötigte Software ist auf dem PC installiert (S. 58).
- ✓ Sie haben das Handbuch der verwendeten PC-Software gelesen und verstanden. Links auf die Softwarehandbücher finden Sie in der Datei A000T0081 auf dem Datenträger der PI Software Suite.
- ✓ Der C-887 ist ausgeschaltet.

Kommunikation über TCP/IP herstellen

Im Folgenden ist das Vorgehen für PIMikroMove® beschrieben. Das Vorgehen bei den anderen PC-Softwareprogrammen (z. B. PITerminal, Treiber zur Verwendung mit NI LabVIEW-Software) ist ähnlich.

1. Schalten Sie den C-887 ein.

   Wenn die IP-Adressen der Netzwerkteilnehmer mit AutoIP konfiguriert werden, dauert es nach dem Ende des Startvorgangs des C-887 (S. 74) bis zu 2 Minuten, bis die Kommunikation über TCP/IP möglich ist.

   – Warten Sie, bis die AutoIP-Konfiguration beendet ist, bevor Sie PIMikroMove® oder eine andere PC-Software von PI starten.

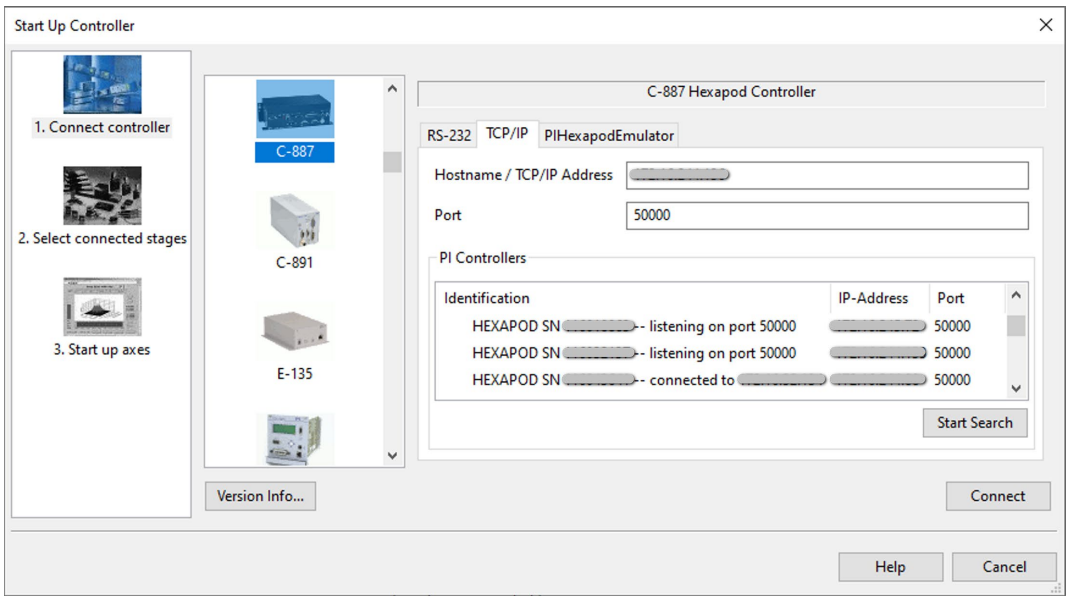
2. Starten Sie PIMikroMove®.

   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...***.

3. Wählen Sie im Feld für die Controllerauswahl ***C-887*** aus.
4. Wählen Sie auf der rechten Seite des Fensters die Registerkarte ***TCP/IP*** aus.

   Alle Controller im selben Netzwerk werden angezeigt.



5. Wählen Sie in der Liste der gefundenen Controller den Eintrag ***HEXAPOD SN ...*** oder ***F-HEX SN ...*** (*SN* steht für *Seriennummer*).
   - Wenn mehrere Einträge ***HEXAPOD SN ...*** oder ***F-HEX SN ...*** angezeigt werden, identifizieren Sie Ihren C-887 anhand des Namenszusatzes, den Sie zuvor vergeben haben, oder anhand seiner neunstelligen Seriennummer (*SN* ...).

   Wenn der C-887 nicht in der Liste der gefundenen Controller angezeigt wird:
   - Der C-887 befindet sich in einem anderen Subnetz als der PC. Bei entsprechender Netzwerkkonfiguration können Sie die Kommunikation trotzdem herstellen, wenn Sie die aktuelle IP-Adresse des C-887 kennen. Geben Sie die IP-Adresse im Feld ***Hostname / TCP/IP Address*** ein.
   - Prüfen Sie die Netzwerkeinstellungen (S. 357). Wenden Sie sich gegebenenfalls an Ihren Netzwerkadministrator.
6. Prüfen Sie die IP-Adresse im Feld ***Hostname / TCP/IP Address*** und die Portnummer im Feld ***Port***.
7. Wählen Sie die Schaltfläche ***Connect***, um die Kommunikation herzustellen.

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

6.4 Kommunikation über RS-232-Schnittstelle herstellen

6.4.1 Baudrate ändern

Die Schnittstellenparameter für die RS-232-Kommunikation sind werkseitig wie in der untenstehenden Tabelle eingestellt. Mit dem Befehl `IFS?` (S. 208) können die Werte im permanenten Speicher abgefragt werden.

Schnittstellenparameter	Werkseitige Einstellung	Bemerkung
Port für RS-232-Kommunikation (RSPORT)	1	Schreibgeschützt Gibt den für die RS-232-Kommunikation genutzten Port des C-887 an.
Handshake für RS-232-Kommunikation (RSHSHK)	RTS/CTS	Schreibgeschützt Gibt die Handshake-Einstellung des C-887 für die RS-232-Kommunikation an.
Baudrate (RSBAUD)	115200	Gibt die Baudrate des C-887 für die RS-232-Kommunikation an. Weitere mögliche Werte sind 9600, 19200, 38400, 57600. Für die erfolgreiche Herstellung der Kommunikation müssen die Baudraten von C-887 und PC übereinstimmen.

Vor der Herstellung der Kommunikation kann die Änderung der werkseitigen Baudrate-Einstellung des C-887 erforderlich sein.

Voraussetzungen

✓ Sie haben die Kommunikation zwischen dem C-887 und dem PC über eine der vorhandenen Schnittstellen hergestellt.

Baudrate für RS-232-Verbindung ändern

1. Öffnen Sie im verwendeten Programm das Fenster zum Eingeben von Befehlen.
2. Senden Sie den Befehl `IFS 100 RSBAUD xxxxx`, wobei *xxxxx* die neue Baudrate ist.

   Die Änderung der Baudrate erfolgt im permanenten Speicher und wird erst nach einem Neustart des C-887 wirksam.

6.4.2 Kommunikation über RS-232 in der PC-Software herstellen

Voraussetzungen

- ✓ Sie haben die allgemeinen Hinweise zur Inbetriebnahme gelesen und verstanden (S. 69).
- ✓ Der C-887 ist an die RS-232-Schnittstelle des PC angeschlossen.
- ✓ Der C-887 ist eingeschaltet, und der Startvorgang des C-887 ist beendet (S. 74).
- ✓ Der PC ist eingeschaltet.
- ✓ Die benötigte Software ist auf dem PC installiert.
- ✓ Sie haben das Handbuch der verwendeten PC-Software gelesen und verstanden. Links auf die Softwarehandbücher finden Sie in der Datei A000T0081 auf dem Datenträger der PI Software Suite.

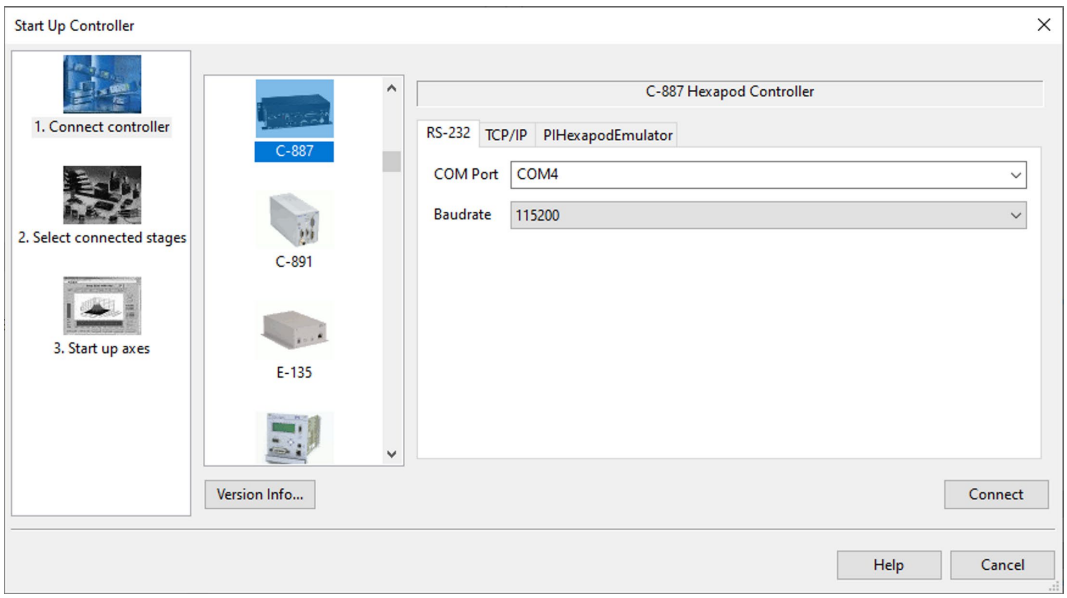
Kommunikation über RS-232 herstellen

Im Folgenden ist das Vorgehen für PIMikroMove® beschrieben. Das Vorgehen bei den anderen PC-Softwareprogrammen (PITerminal, Treiber zur Verwendung mit NI LabVIEW-Software) ist ähnlich.

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-887*** aus.
3. Wählen Sie auf der rechten Seite des Fensters die Registerkarte ***RS-232*** aus.

4. Wählen Sie im Feld ***COM Port*** den COM-Port des PC aus, an dem Sie den C-887 angeschlossen haben.
5. Stellen Sie im Feld ***Baudrate*** den Wert 115200 ein (werkseitige Einstellung bei Auslieferung des C-887).

   Damit passen Sie die Baudrate des PC an die Baudrate des C-887 an.

   Sofern Sie die Baudrate des C-887 geändert haben, müssen Sie stattdessen den neuen Wert in das Feld ***Baud Rate*** eintragen (S. 82).
6. Wählen Sie ***Connect***, um die Kommunikation herzustellen.

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

6.5 Bewegungen starten

Im Folgenden wird PIMikroMove® verwendet, um die Achsen zu bewegen.

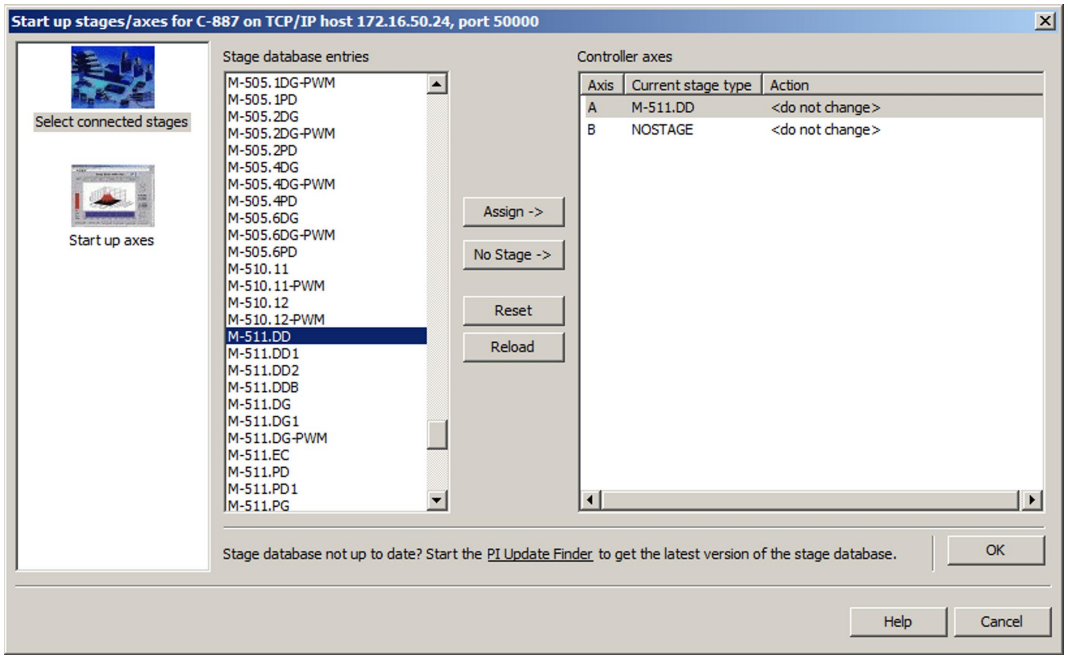
Voraussetzungen

- ✓ Sie haben die allgemeinen Hinweise zur Inbetriebnahme gelesen und verstanden (S. 69).
- ✓ PIMikroMove® ist auf dem PC installiert (S. 58).
- ✓ Sie haben das PIMikroMove®-Handbuch gelesen und verstanden. Links auf die Softwarehandbücher finden Sie in der Datei A000T0081 auf dem Datenträger der PI Software Suite.
- ✓ Sie haben die Benutzerhandbücher aller angeschlossenen Positionierer (Hexapod, Achsen A und B) gelesen und verstanden.
- ✓ Sie haben die Positionierer gemäß den Anweisungen in den entsprechenden Benutzerhandbüchern installiert und am C-887 angeschlossen (S. 63, S. 64).
- ✓ Sie haben die Kommunikation zwischen dem C-887 und dem PC mit PIMikroMove® über die TCP/IP-Schnittstelle (S. 79) oder die RS-232-Schnittstelle (S. 82) hergestellt.

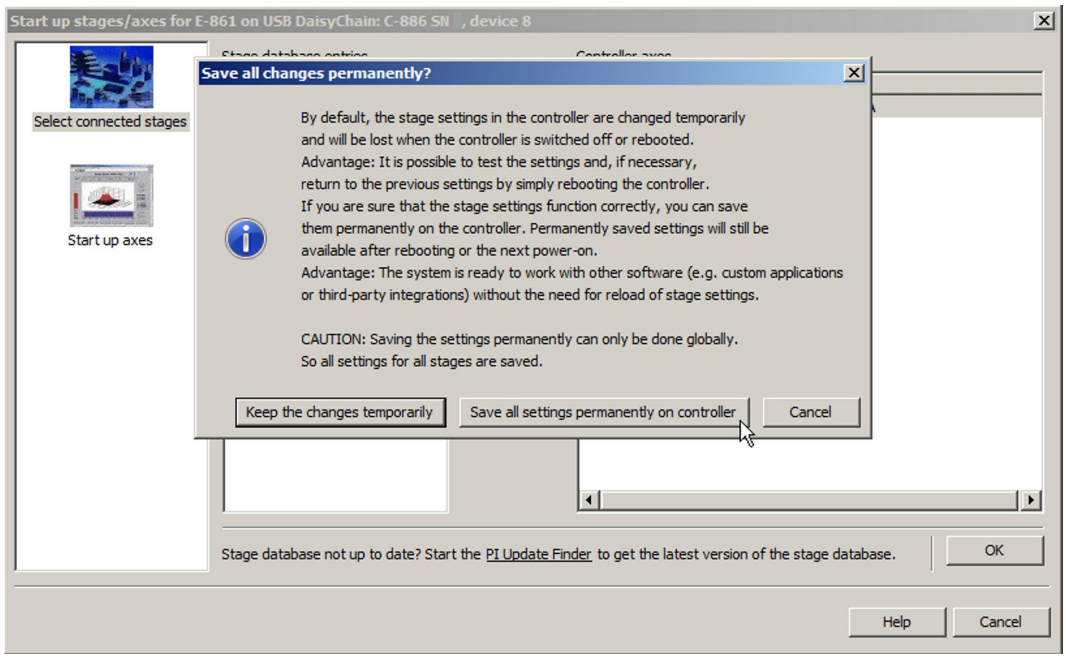
Bewegungen starten mit PIMikroMove®

1. Wenn notwendig, weisen Sie den Achsen A und B den passenden Positionierertyp zu:
   a) Wählen Sie im Fenster ***Start up controller*** in der linken Spalte ***2. Select connected stages***, um zum Schritt für die Zuweisung des Positionierertyps zu wechseln.
   b) Wählen Sie die Achse, der Sie einen Positionierertyp zuweisen wollen, in der Liste ***Controller axes***.
   c) Wählen Sie den Positionierertyp in der Liste ***Stage database entries***.
   d) Wählen Sie ***Assign***.

e) Wenn notwendig, wiederholen Sie die Schritte b bis d für die zweite Achse.



f) Bestätigen Sie die Auswahl mit ***OK***, um die Parametereinstellungen für den ausgewählten Positionierertyp aus der Positioniererdatenbank zu laden (Details siehe "Betriebsparameter der Achsen A und B" (S. 32)). Der Dialog ***Save all changes permanently?*** öffnet sich.

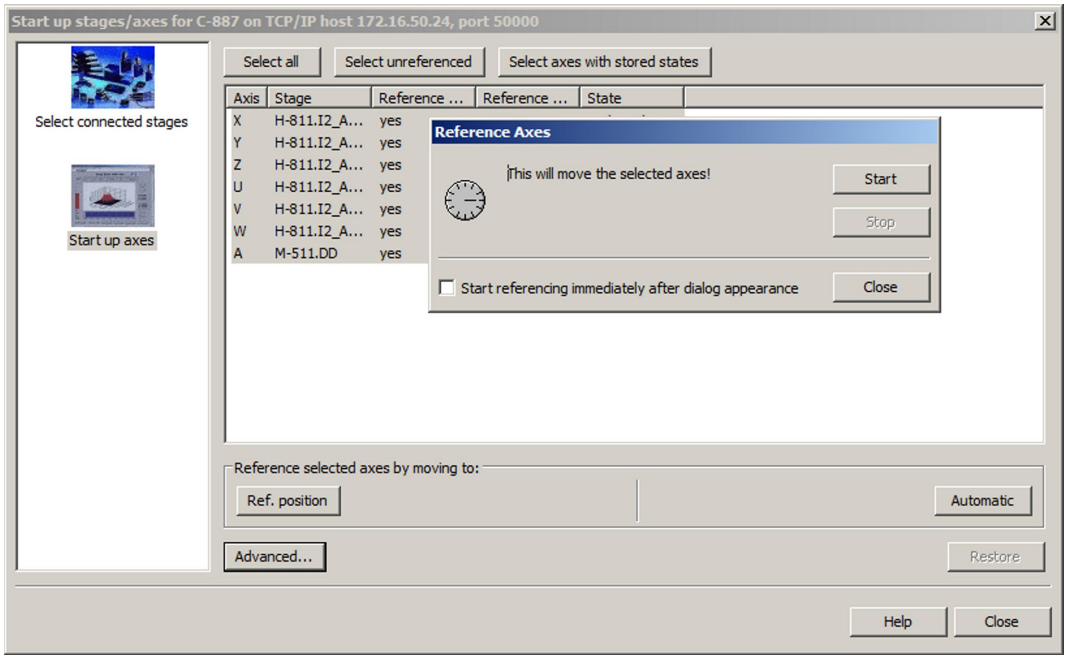


g) Geben Sie im Dialog ***Save all changes permanently?*** an, wie Sie die Parametereinstellungen laden wollen:

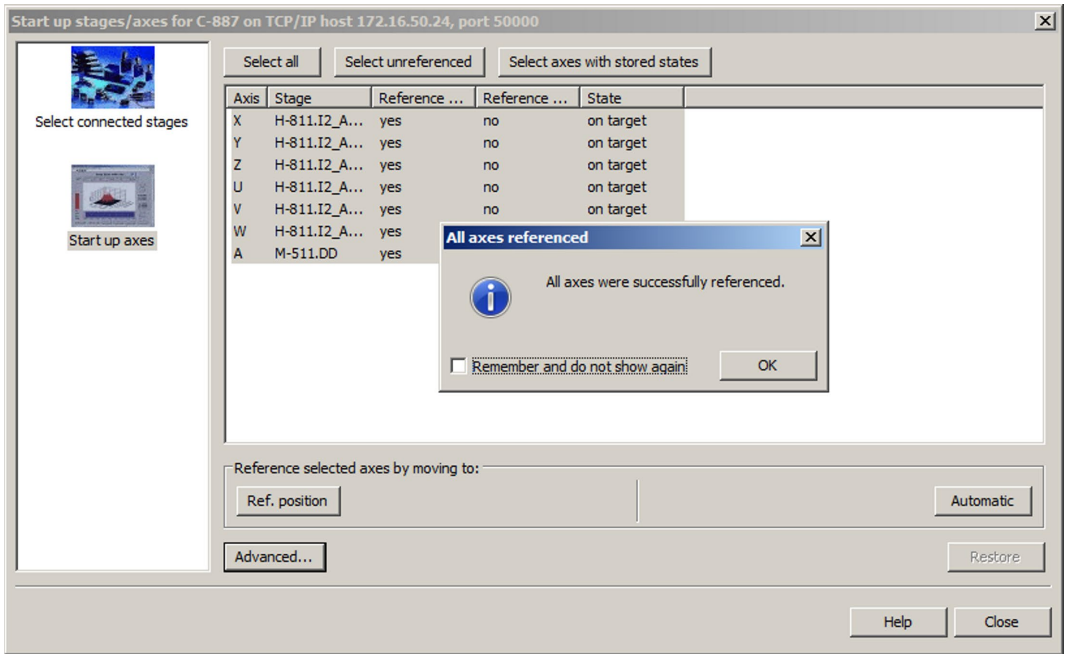
- Als Standardwerte laden (empfohlen): Wählen Sie ***Save all settings permanently on controller***, um die Parametereinstellungen in den permanenten Speicher zu laden. Die Einstellungen sind nach dem Einschalten oder Neustart des C-887 sofort vorhanden und müssen nicht erneut geladen werden.
- Temporär laden: Wählen Sie ***Keep the changes temporarily***, um die Parametereinstellungen in den flüchtigen Speicher zu laden. Die Einstellungen gehen beim Ausschalten oder Neustart des C-887 verloren.

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

2. Führen Sie im Schritt ***Start up axes*** die Referenzierungsfahrt für die Achsen aus (erforderlich für Achsen mit inkrementellen Sensoren). Wählen Sie dazu ***Ref.position*** oder auf ***Automatic***.



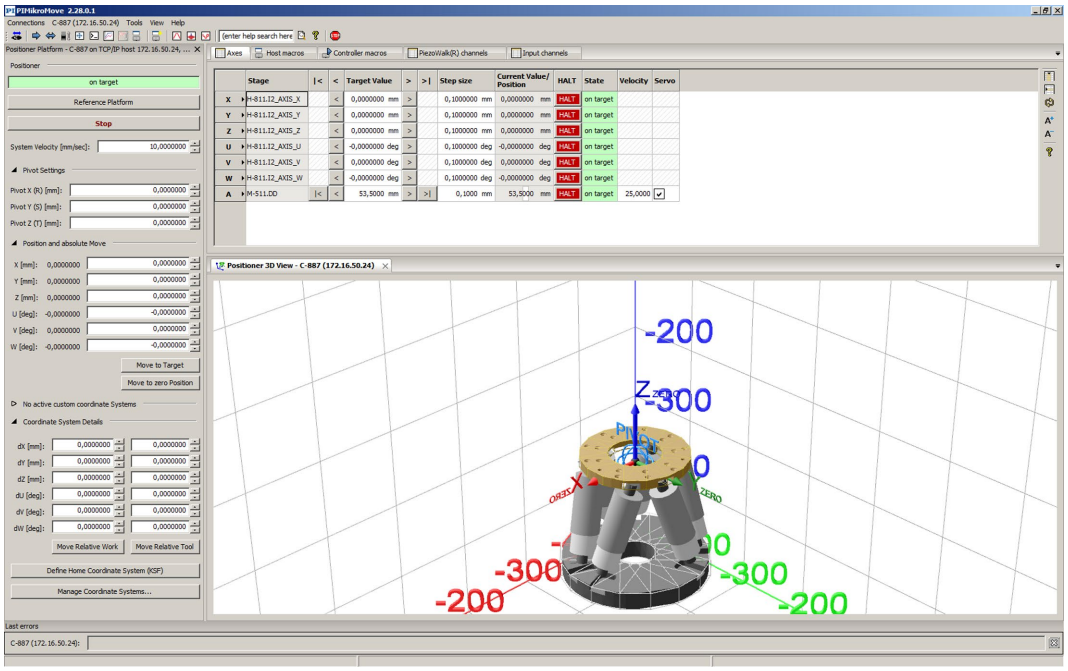
3. Nach erfolgreicher Referenzierungsfahrt wählen Sie ***OK > Close***.



4. Wenn im Hauptfenster von PIMikroMove® **nicht** das Fenster ***Positioner Platform*** und die Registerkarte ***Positioner 3D View*** angezeigt werden:

   – Blenden Sie das Fenster ***Positioner Platform*** mit dem Menüeintrag ***C-887 > Show Positioner platform settings*** ein.

   – Blenden Sie die Registerkarte ***Positioner 3D View*** mit dem Menüeintrag ***C-887 > Positioner 3D View > Show*** ein.



5. Starten Sie einige Testbewegungen für die Achsen der Bewegungsplattform des Hexapods (X bis W):

   a) Geben Sie im Fenster ***Positioner Platform*** im Bereich ***Position and absolute Move*** eine Zielposition für mindestens eine Achse der Bewegungsplattform des Hexapods ein.

   b) Wählen Sie ***Move to Target***, um die Bewegung zur angegebenen Zielposition zu starten.

   c) Wenn die Bewegung beendet ist, wiederholen Sie die Schritte a und b für eine neue Zielposition.

   Wenn eine Stützstelle des berechneten Dynamikprofils oder die Zielposition nicht erreicht werden kann, wird die Bewegung nicht ausgeführt, und ein Fenster mit einer Fehlermeldung öffnet sich.

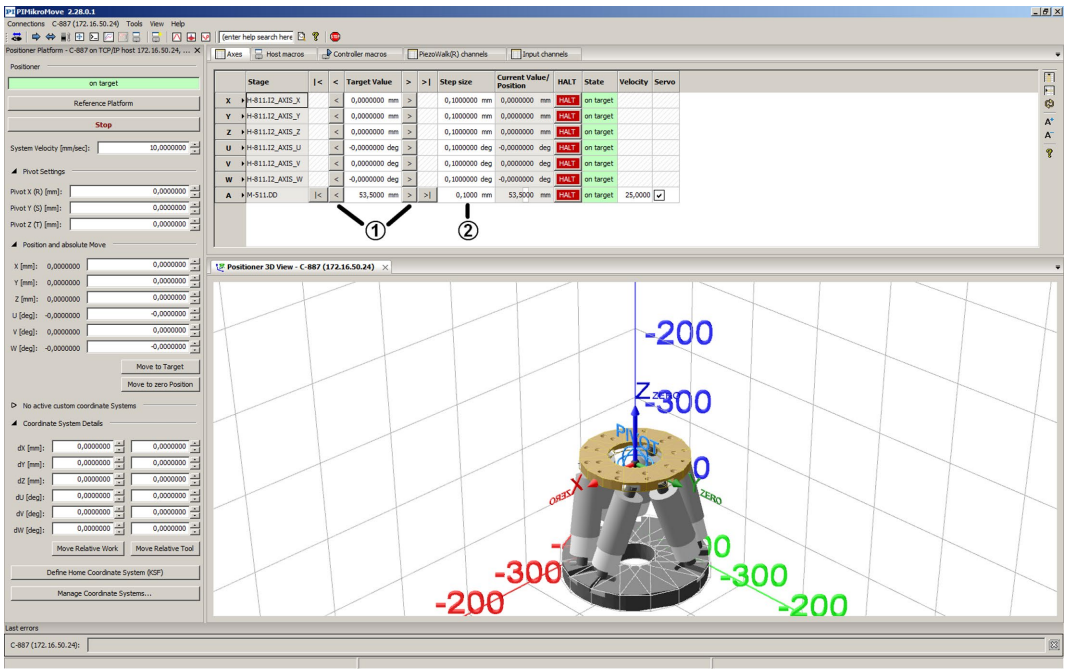
   Während einer Bewegung kann keine neue Zielposition eingegeben werden.

   Auf der Registerkarte ***Positioner 3D View*** wird die Bewegung grafisch abgebildet.

6. Starten Sie im Hauptfenster von PIMikroMove® auf der Registerkarte ***Axes*** einige Testbewegungen der Achsen A und B.

   a) Wenn notwendig, schalten Sie den Servomodus für die Achsen A und B ein, indem Sie das entsprechende Kontrollkästchen in der Spalte ***Servo*** aktivieren.

   b) Starten Sie die Testbewegungen: Sie können z. B. Schritte mit einer bestimmten Schrittweite (2) ausführen, indem Sie die entsprechenden Pfeilschaltflächen (1) für eine Achse wählen.



7 Betrieb

In diesem Kapitel

Allgemeine Hinweise zum Betrieb 89
Schutzfunktionen des C-887 90
Datenrekorder 99
Funktionsgenerator 102
Controllermakros 127

7.1 Allgemeine Hinweise zum Betrieb

VORSICHT

Quetschgefahr durch bewegte Teile!

Zwischen den bewegten Teilen des Hexapods und einem feststehenden Teil oder Hindernis besteht die Gefahr von leichten Verletzungen durch Quetschung.

- Halten Sie Ihre Finger von Bereichen fern, in denen sie von bewegten Teilen erfasst werden können.

HINWEIS

Schäden durch Kollisionen!

Kollisionen können den Hexapod, die zu bewegende Last und die Umgebung beschädigen.

- Stellen Sie sicher, dass im Arbeitsraum des Hexapods keine Kollisionen zwischen Hexapod, zu bewegender Last und Umgebung möglich sind.
- Platzieren Sie keine Gegenstände in Bereichen, in denen sie von bewegten Teilen erfasst werden können.
- Kommandieren Sie nur zulässige Posen. Für die Definition einer zulässigen Pose siehe "Zulässige Posen ermitteln" (S. 62).
- Halten Sie bei einer Fehlfunktion des Controllers die Bewegung sofort an.

HINWEIS

Schäden durch ungewollte Positionsänderungen!

Das Simulationsprogramm PIVirtualMove berechnet die zulässigen Posen auf Basis der maximalen Nutzlast des Hexapods (= Grenzwert bei eingeschaltetem Servomodus). Die maximale Haltekraft bei ausgeschaltetem Servomodus basiert auf der Selbsthemmung der Aktoren in den Hexapod-Beinen und fällt geringer aus als die maximale Nutzlast.

Wenn die tatsächliche Belastung des Hexapods die auf der Selbsthemmung der Aktoren basierende maximale Haltekraft überschreitet, können in folgenden Fällen ungewollte Positionsänderungen des Hexapods auftreten, obwohl PIVirtualMove die aktuelle Pose als zulässig eingestuft hat:

- Ausschalten des C-887
- Neustart des C-887
- Ausschalten des Servomodus für die Achsen der Bewegungsplattform des Hexapods, z. B. durch Verwendung der Buchse **E-Stop** (S. 93)

Dadurch sind Kollisionen zwischen Hexapod, zu bewegender Last und Umgebung möglich. Kollisionen können den Hexapod, die zu bewegende Last oder die Umgebung beschädigen.

➢ Stellen Sie sicher, dass die tatsächliche Belastung der Bewegungsplattform des Hexapods die auf der Selbsthemmung der Aktoren basierende maximale Haltekraft nicht überschreitet, bevor Sie den Servomodus ausschalten, den C-887 neu starten oder ausschalten.

7.2 Schutzfunktionen des C-887

INFORMATION

Wenn eine Sicherheitsfunktion gemäß geltenden Normen erforderlich ist:

➢ Definieren und implementieren Sie geeignete Maßnahmen. Beispiele für mögliche Maßnahmen:

- Wenn die Stromversorgung des Hexapods über den C-887 erfolgt, unterbrechen Sie die Versorgungsleitung durch geeignete Sicherheitstechnik.
- Verwenden Sie für den Hexapod ein externes Netzteil mit geeigneter Sicherheitstechnik.
- Nur Modelle C-887.522, .523, .532, .533: Schließen Sie an der Buchse **E-Stop** geeignete Sicherheitstechnik an. Weitere Informationen siehe "Buchse E-Stop verwenden" (S. 93).

7.2.1 Automatisches Ausschalten des Servomodus / Stoppen der Bewegung

Bewegungen der Achsen sind nur bei eingeschaltetem Servomodus möglich.

Der C-887 schaltet in folgenden Fällen automatisch den Servomodus für die Achsen des Hexapods (X, Y, Z, U, V, W) und die Achsen A und B aus und stoppt damit die Bewegung:

Fehlercode	Ursachen
66	Der Parameter ***Check PowerGood Signal*** (ID 0x19004000) hat den Wert 1. Der C-887 prüft deshalb das Power-Good-Signal (Pin 59 der Buchse **Hexapod**). Die Prüfung hat ergeben, dass das Power-Good-Signal außerhalb des erforderlichen Bereichs (S. 383) liegt. Mögliche Gründe: ▪ Stromversorgung der Antriebe des Hexapods ist unterbrochen oder fehlerhaft. ▪ Der angeschlossene Hexapod gehört zu einer älteren Modellreihe, die nicht mit einem Power-Good-Signal ausgestattet ist.
501	Nur Modelle C-887.522, .523, .532, .533: Durch die aktuelle Beschaltung der Buchse **E-Stop** ist das Power-OK-Signal **nicht** vorhanden, und der 24-V-Ausgang für den Hexapod (**24 V Out 7 A**) ist deaktiviert. Weitere Informationen siehe "Buchse E-Stop verwenden" (S. 93).
216	Für mindestens ein Hexapod-Bein oder eine der Achsen A und B ist ein Endschalter aktiv.

Für bestimmte Fehler kann die Behandlung der Achsen konfiguriert werden. Je nach Konfiguration stoppt der C-887 im Fehlerfall alle Achsen oder nur die betroffene Achse mit oder ohne Abschaltung des Servomodus.

Die folgenden Parameter konfigurieren die Behandlung der Achsen:

Parameter	Beschreibung und mögliche Werte
Axis Handling on Motion Error (ID 0x19004001)	Auswahl der im Fehlerfall zu behandelnden Achsen 0 = Behandlung nur für die betroffene Achse 1 = Behandlung für alle Achsen (Standard)
Type of Axis Handling on Motion Error (ID 0x19004002)	Art der Behandlung im Fehlerfall 0 = Stoppen der Bewegung durch STP (Fehlercode 10 wird gesetzt), Servomodus bleibt eingeschaltet 1 = Stoppen der Bewegung durch Ausschalten des Servomodus (Standard)

Die Konfiguration wird in folgenden Fällen wirksam:

Fehlercode	Ursachen
1024	Bewegungsfehler liegt für mindestens ein Hexapod-Bein vor: Differenz zwischen aktueller Position und kommandierter Position überschreitet maximal zulässigen Wert, z. B. wegen einer Antriebs- oder Sensorstörung oder wegen Überschreiten der Dynamikgrenzen für den aktuellen Lastfall.

Fehlercode	Ursachen
202	Die Bewegung wurde durch Funktionsgenerator (S. 102) oder Fast-Alignment-Routinen (S. 2) ausgelöst, und der interne FIFO-Puffer enthält nicht mehr genug Zielpositionen.
657	Für mindestens ein Hexapod-Bein ist der Antrieb blockiert. Weitere Informationen siehe "Sicherheitsabschaltung konfigurieren" (S. 97).
606	Für mindestens ein Hexapod-Bein wurde die maximal zulässige Geschwindigkeit für fehlerfreie Positionserfassung überschritten. Weitere Informationen siehe "Sicherheitsabschaltung konfigurieren" (S. 97).

INFORMATION

Abhängig von der aktuellen Geschwindigkeit und Last kann sich die Plattform des Hexapods nach dem Stoppen auf einer undefinierten Bahn bis zum kompletten Stillstand bewegen.

Die Sensoren des Hexapods werden über das Datenübertragungskabel (Buchse **Hexapod** (S. 383)) mit Strom versorgt. Deshalb ist die aktuelle Position der Achsen des Hexapods auch dann bekannt, wenn die Stromversorgung der Antriebe des Hexapods unterbrochen ist.

Betriebsbereitschaft wiederherstellen

1. Lesen Sie den Fehlercode und Diagnoseinformationen aus. Verwenden Sie die Befehle `ERR?` (S. 176) und `DIA?` (S. 165) oder die entsprechenden Bedienelemente der PC-Software.

   Die Abfrage setzt den Fehlercode auf null zurück.

2. Überprüfen Sie Ihr System und stellen Sie sicher, dass alle Achsen gefahrlos bewegt werden können. Siehe auch "Störungsbehebung" (S. 357).

3. Wenn die Stromversorgung der Antriebe des Hexapods unterbrochen ist: Stellen Sie die Stromversorgung wieder her.

4. Nur Modelle C-887.522, .523, .532, .533: Beschalten Sie die Buchse **E-Stop** so, dass Bewegungen der Achsen möglich sind. Details siehe "Buchse E-Stop verwenden" (S. 93).

5. Nur wenn der angeschlossene Hexapod nicht mit einem Power-Good-Signal (Pin 59 der Buchse **Hexapod**) ausgestattet ist: Deaktivieren Sie die Prüfung des Power-Good-Signals. Details siehe "Störungsbehebung" (S. 357).

6. Prüfen Sie den Aktivierungszustand des Servomodus: Verwenden Sie den Befehl `SVO?` (S. 269) oder die entsprechenden Bedienelemente der PC-Software.

7. Wenn der Servomodus ausgeschaltet ist: Schalten Sie den Servomodus für alle Achsen ein. Verwenden Sie den Befehl `SVO` (S. 268) oder die entsprechenden Bedienelemente der PC-Software.

   Eine erneute Referenzierungsfahrt ist nicht erforderlich.

7.2.2 Buchse E-Stop verwenden

Die Modelle C-887.522, .523, .532 und .533 sind mit der Buchse **E-Stop** ausgestattet. Die Buchse **E-Stop** steuert ein internes Relais mit Schließer, das den 24-V-Ausgang für den Hexapod (Buchse **24 V Out 7 A**) aktiviert oder deaktiviert. Ein internes Power-OK-Signal zeigt dem C-887 die aktuelle Beschaltung der Buchse **E-Stop** an.

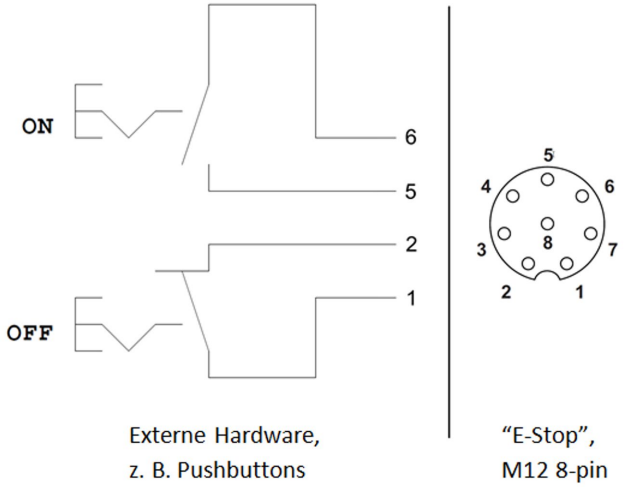


Abbildung 9: Beschaltung der Buchse E-Stop durch externe Hardware

ON	Make contact
OFF	Break contact
E-Stop	Pins der Buchse **E-Stop** auf der Vorderwand des C-887, Ansicht von vorn, für die komplette Pinbelegung siehe "E-Stop" (S. 381)

Damit Bewegungen der Achsen möglich sind, muss durch die externe Beschaltung der Buchse **E-Stop** Pin 1 mit Pin 2 verbunden sein und Pin 5 mit Pin 6 (24-V-Ausgang des C-887 ist aktiviert, das Power-OK-Signal wird bereitgestellt).

Wenn die Verbindung zwischen den Pins 1 und 2 und/oder zwischen den Pins 5 und 6 unterbrochen wird, werden alle Achsen gestoppt. Bei unterbrochener Verbindung ist das erneute Auslösen einer Bewegung nicht möglich. Details:

- Das interne Power-OK-Signal ist nicht vorhanden.
- Der 24-V-Ausgang des C-887 ist deaktiviert.
- Der C-887 schaltet automatisch den Servomodus für die Achsen des Hexapods (X, Y, Z, U, V, W) und die Achsen A und B aus und verhindert das erneute Einschalten.
- Der Fehlercode 501 wird gesetzt.
- Wenn der Hexapod am 24-V-Ausgang des C-887 angeschlossen ist, zeigt das Power-Good-Signal (Pin 59 der Buchse **Hexapod**) an, dass die Stromversorgung der Antriebe des Hexapods unterbrochen ist (siehe auch "Automatisches Ausschalten des Servomodus / Stoppen der Bewegung" (S. 91)).

INFORMATION

Die Buchse **E-Stop** bietet **keine** direkte Sicherheitsfunktion im Sinne geltender Normen (z. B. IEC 60204-1, IEC 61508 oder IEC 62061). Wenn eine Sicherheitsfunktion gemäß geltenden Normen erforderlich ist, müssen durch den Betreiber des Hexapod-Systems geeignete Maßnahmen definiert und implementiert werden.

INFORMATION

Abhängig von der aktuellen Geschwindigkeit und Last kann sich die Plattform des Hexapods nach dem Stoppen auf einer undefinierten Bahn bis zum kompletten Stillstand bewegen.

Die Sensoren des Hexapods werden über das Datenübertragungskabel (Buchse **Hexapod** (S. 383)) mit Strom versorgt. Deshalb ist die aktuelle Position der Achsen des Hexapods auch dann bekannt, wenn die Stromversorgung der Antriebe des Hexapods unterbrochen ist.

Werkzeug und Zubehör

Wenn Sie die Buchse **E-Stop** aktiv verwenden wollen und **keine Sicherheitsfunktion** gemäß geltenden Normen benötigen:

- Motion-Stop-Box C-887.MSB (erhältlich als optionales Zubehör (S. 24))

Wenn Sie die Buchse **E-Stop** aktiv verwenden wollen und eine Sicherheitsfunktion gemäß geltenden Normen implementieren wollen:

- Geeignete externe Hardware für die Beschaltung der Buchse **E-Stop**, z. B. Drucktaster oder Schalter
- Wenn die Stromversorgung des Hexapods nicht über den 24-V-Ausgang des C-887 erfolgen soll: separates geeignetes Netzteil

Wenn Sie die Buchse **E-Stop nicht** aktiv verwenden, sondern die Pinpaare 1/2 und 5/6 dauerhaft überbrücken wollen (Bewegungen sind immer möglich):

- Kurzschlussstecker C887B0038 (im Lieferumfang (S. 23))

Buchse E-Stop verwenden

1. Verbinden Sie die externe Hardware gemäß den Anforderungen Ihrer Anwendung mit der Buchse **E-Stop**. Die Abbildungen unten zeigen Beispiele für Anschlussvarianten.
   – Sichern Sie die Steckverbindung gegen unbeabsichtigtes Abziehen.
2. Stellen Sie bei der Inbetriebnahme (S. 69) des Hexapod-Systems sicher, dass die externe Beschaltung der Buchse **E-Stop** Bewegungen der Achsen erlaubt.
3. Wenn Sie die Achsen durch die an **E-Stop** angeschlossene Hardware gestoppt haben: Stellen Sie die Betriebsbereitschaft der Achsen wieder her:
   a) Lesen Sie den Fehlercode aus. Verwenden Sie den Befehl `ERR?` (S. 176) oder die entsprechenden Bedienelemente der PC-Software.
   Der Fehlercode 501 zeigt an, dass die Beschaltung der Buchse **E-Stop** Bewegungen verhindert.

b) Überprüfen Sie Ihr System und stellen Sie sicher, dass alle Achsen gefahrlos bewegt werden können. Siehe auch "Störungsbehebung" (S. 357).

c) Stellen Sie die Stromversorgung der Antriebe des Hexapods wieder her.

d) Beschalten Sie die Buchse **E-Stop** so, dass Bewegungen der Achsen möglich sind.

e) Schalten Sie den Servomodus für alle Achsen ein. Verwenden Sie den Befehl `SVO` (S. 268) oder die entsprechenden Bedienelemente der PC-Software.
Eine erneute Referenzierungsfahrt ist nicht erforderlich.

Anschlussvariante ohne Sicherheitsfunktion gemäß geltenden Normen

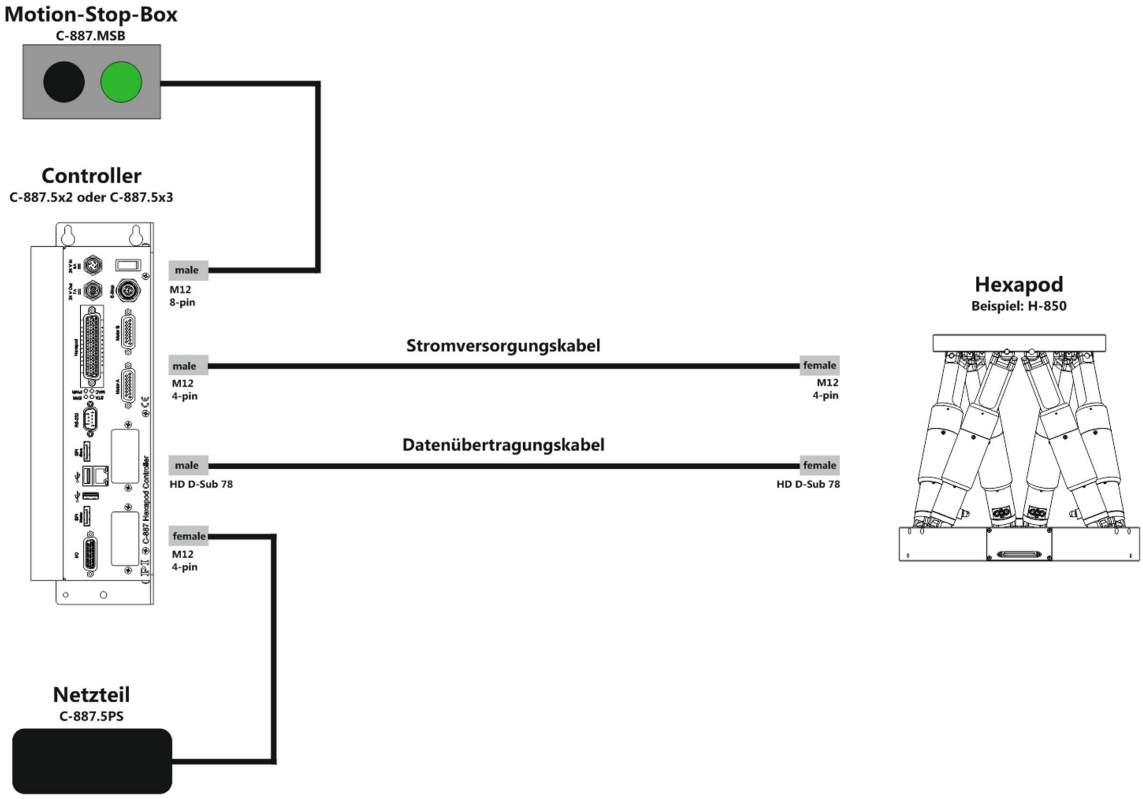


Abbildung 10: Buchse E-Stop mit Motion-Stop-Box C-887.MSB

Abbildung 11: Bedienelemente der Motion-Stop-Box C-887.MSB
STOP Pilzsperrtaster mit Verriegelung
RUN Grüner Drucktaster

Bedienung der Motion-Stop-Box:

Stoppen der Bewegung:
STOP durch Drücken verriegeln

Betriebsbereitschaft wiederherstellen:
STOP durch erneutes Drücken entriegeln, anschließend RUN drücken

Anschlussvariante mit Sicherheitsfunktion

Die Stromversorgung des Hexapods erfolgt über den C-887. Der Betreiber stellt externe Hardware für die Buchse **E-Stop** mit Sicherheitsfunktion gemäß geltenden Normen.

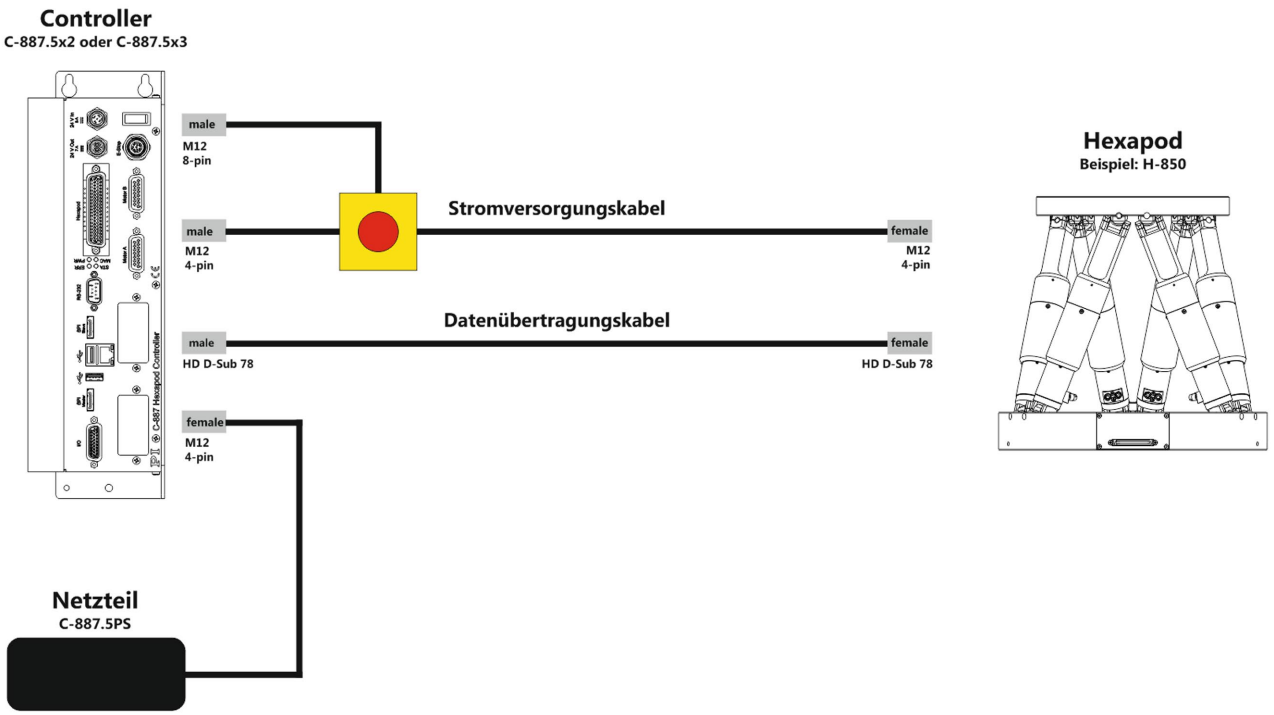


Abbildung 12: Buchse E-Stop mit externer Hardware verwenden

Anschlussvariante mit Sicherheitsfunktion

Der Betreiber stellt externe Hardware für die Buchse **E-Stop** und ein separates Netzteil für den Hexapod mit Sicherheitsfunktion gemäß geltenden Normen.

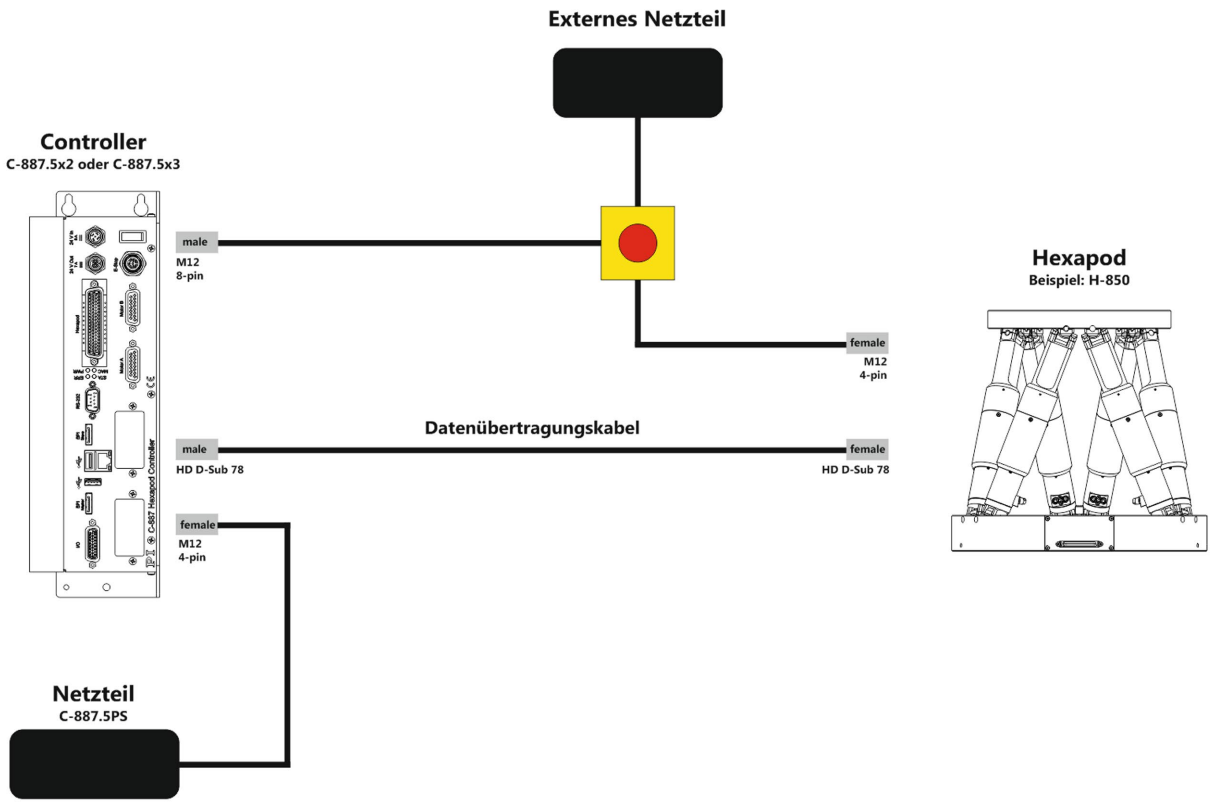


Abbildung 13: Buchse E-Stop mit externer Hardware verwenden, externes Netzteil für Hexapod

7.2.3 Sicherheitsabschaltung konfigurieren

Pro Hexapod-Bein kann im C-887 eine Sicherheitsabschaltung konfiguriert werden. Je nach Konfiguration der Sicherheitsabschaltung stoppt der C-887 im Fehlerfall alle Achsen oder nur die betroffene Achse mit oder ohne Ausschalten des Servomodus.

Fehler können zum Beispiel verursacht werden durch:

- Mechanische Überlastung
- Zu hohe Beschleunigung
- Defekter oder falscher Sensoranschluss
- Elektrischer oder mechanischer Sensorfehler
- Motorstrombegrenzung
- Phasenfehler des Motors

Die Sicherheitsabschaltung basiert auf folgenden Prüfungen:

- "Blockiererkennung": Ist der Antrieb blockiert?
- "Sensorfehler": Überschreitet die aktuelle Geschwindigkeit des Antriebs die maximal zulässige Geschwindigkeit für fehlerfreie Positionserfassung durch den Sensor?

Für das Wiederherstellen der Betriebsbereitschaft nach einer Sicherheitsabschaltung siehe "Automatisches Ausschalten des Servomodus / Stoppen der Bewegung" (S. 91).

Die folgenden Parameter konfigurieren die Sicherheitsabschaltung:

Parameter	Beschreibung und mögliche Werte
Axis Handling on Motion Error (ID 0x19004001)	Auswahl der im Fehlerfall zu behandelnden Achsen 0 = Behandlung nur für die betroffene Achse 1 = Behandlung für alle Achsen (Standard)
Type of Axis Handling on Motion Error (ID 0x19004002)	Art der Behandlung im Fehlerfall 0 = Stoppen der Bewegung durch STP (Fehlercode 10 wird gesetzt), Servomodus bleibt eingeschaltet 1 = Stoppen der Bewegung durch Ausschalten des Servomodus (Standard)
Max Enc Vel 0x1A002000	Sicherheitsabschaltung "Sensorfehler" Maximal zulässige Geschwindigkeit für fehlerfreie Positionserfassung durch den Sensor. Wenn die Geschwindigkeit des Antriebs den Wert des Parameters überschreitet, wird die Sicherheitsabschaltung ausgelöst und der Fehlercode 606 gesetzt. in mm/s
Activate Motor Stuck Check (ID 0x1A002100)	Sicherheitsabschaltung "Blockiererkennung" Status der Blockiererkennung für den Antrieb 0 = Blockiererkennung deaktiviert (Standard) 1 = Blockiererkennung aktiviert Für sinnvolle Verwendung der Blockiererkennung müssen I-Term und I-Limit für die Positionsregelung des Antriebs ausreichend hoch eingestellt sein. Ablauf der Blockiererkennung: ▪ Schritt 1: Prüfung, ob der Motorstellwert des Antriebs den Wert des Parameters 0x1A002200 überschritten hat. ▪ Schritt 2: Wenn Motorstellwert > Parameter 0x1A002200: Nach Ablauf der Verzögerungszeit (Vorgabe durch Parameter 0x1A002400) wird geprüft, ob die Geschwindigkeit des Antriebs größer ist als der Wert des Parameters 0x1A002300. ▪ Schritt 3: Wenn Geschwindigkeit < Parameter 0x1A002300, ist der Antrieb blockiert. Die Sicherheitsabschaltung wird ausgelöst, und der Fehlercode 657 wird gesetzt.

Parameter	Beschreibung und mögliche Werte
Minimal MotorOut to Move Axis (ID 0x1A002200)	Sicherheitsabschaltung "Blockiererkennung" Motorstellwert, ab dem der Antrieb den Status "in Bewegung" hat dimensionslos Dieser Parameter wird nur ausgewertet, wenn der Parameter 0x1A002100 den Wert 1 hat.
Velocity Threshold under which Axis is considered not moving (ID 0x1A002300)	Sicherheitsabschaltung "Blockiererkennung" Geschwindigkeit, ab der der Antrieb den Status "in Bewegung" hat. Sinnvolle Werte hängen ab von der Sensorauflösung und der minimalen Systemgeschwindigkeit (Parameter 0x19001501) des Hexapods. in mm/s Dieser Parameter wird nur ausgewertet, wenn der Parameter 0x1A002100 den Wert 1 hat.
Time Period for which Axis is not yet considered not moving (ID 0x1A002400)	Sicherheitsabschaltung "Blockiererkennung" Verzögerungszeit für die Prüfung der Geschwindigkeit des Antriebs, gilt für das gesamte System in s Dieser Parameter wird nur ausgewertet, wenn der Parameter 0x1A002100 den Wert 1 hat.

7.3 Datenrekorder

7.3.1 Eigenschaften des Datenrekorders

Der C-887 enthält einen Echtzeit-Datenrekorder. Der Datenrekorder kann verschiedene Größen für Achsen (z. B. aktuelle Position) sowie die Signale der analogen Eingänge aufzeichnen.

Die aufgezeichneten Daten werden temporär in 16 Datenrekordertabellen mit jeweils maximal 262144 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.

7.3.2 Datenrekorder einrichten

INFORMATION

Die Einstellungen zur Einrichtung des Datenrekorders lassen sich nur im flüchtigen Speicher des C-887 ändern. Nach dem Einschalten oder dem Neustart des C-887 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. 200).

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

Datenrekorder konfigurieren

Sie können den Datenrekordertabellen die Datenquellen und die Aufzeichnungsoptionen zuordnen.

➢ Senden Sie den Befehl `DRC?` (S. 170), 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-887 Folgendes auf:

- Datenrekordertabelle 1, 3, 5, 7, 9, 11: kommandierte Position der Achsen X, Y, Z, U, V und W
- Datenrekordertabelle 2, 4, 6, 8, 10, 12: aktuelle Position der Achsen X, Y, Z, U, V und W
- Datenrekordertabelle 13: die Zeit

➢ Konfigurieren Sie den Datenrekorder mit dem Befehl `DRC` (S. 168).

INFORMATION

Mit der Aufzeichnungsoption 80 (***Status register of axis***) können für die Beine 1 bis 6 des Hexapods und für die Achsen A und B die Bits des Signalstatusregisters (S. 386) aufgezeichnet werden.

➢ Verwenden Sie zur Aufzeichnung des Signalstatusregisters das Fenster ***Data Recorder*** in PIMikroMove®, das die gezielte Auswahl einzelner Bits für die grafische Darstellung der aufgezeichneten Daten erlaubt. Details siehe PIMikroMove®-Handbuch.

Sie können festlegen, wie die Aufzeichnung ausgelöst werden soll.

➢ Fragen Sie mit `DRT?` (S. 175) die aktuelle Triggeroption ab.

➢ Ändern Sie die Triggeroption mit dem Befehl `DRT` (S. 173). Die Triggeroption gilt für alle Datenrekordertabellen, deren Aufzeichnungsoption nicht auf 0 eingestellt ist.

Sie können festlegen, wieviele Punkte pro Datenrekordertabelle maximal aufgezeichnet werden sollen.

➢ Ändern Sie mit dem Befehl `SPA` (S. 258) den Parameter ***Data Recorder Points Per Table***, ID 0x16000201. Maximaler Wert: 262144 Punkte (Standard: 8192 Punkte).

Aufzeichnungsrate einstellen

- Senden Sie den Befehl `RTR?` (S. 255), um die Aufzeichnungsrate des Datenrekorders auszulesen.
  Die Aufzeichnungsrate gibt über einen Faktor an, mit welcher Frequenz Datenpunkte aufgezeichnet werden. Der mögliche Wertebereich der Faktoren für RTR ist 1 bis 10000. Dabei entspricht der Faktor 1 einer Frequenz von 10 kHz. Der Standardwert beträgt 10 und entspricht 1 kHz.
- Ändern Sie die Aufzeichnungsrate mit dem Befehl `RTR` (S. 254).

Je größer die mit `RTR` gesetzte Aufzeichnungsrate ist, desto größer ist die maximale Dauer der Datenaufzeichnung.

INFORMATION

Wenn der C-887 Fast-Alignment-Routinen (S. 2) ausführt, setzt er die Aufzeichnungsrate auf den Faktor 4.

7.3.3 Aufzeichnung starten

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

Unabhängig von der eingestellten Triggeroption wird die Datenaufzeichnung immer in folgenden Fällen ausgelöst:

- Starten einer Sprungantwortmessung mit `STE` (S. 266)
- Starten einer Impulsantwortmessung mit `IMP` (S. 209)
- Starten der Funktionsgeneratorausgabe mit `WGO` (S. 289)
- Während der Funktionsgeneratorausgabe: `WGR` (S. 291) startet die Aufzeichnung mit dem nächsten Ausgabezyklus des Funktionsgenerators

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

7.3.4 Aufgezeichnete Daten auslesen

INFORMATION

Das Auslesen der aufgezeichneten Daten kann abhängig von der Anzahl der Datenpunkte einige Zeit dauern.
Die Daten können auch bei laufender Datenaufzeichnung ausgelesen werden.

- Lesen Sie die zuletzt aufgezeichneten Daten mit dem Befehl `DRR?` (S. 172) 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. 171) ab.

7.4 Funktionsgenerator

7.4.1 Funktionsweise des Funktionsgenerators

Die Funktionsgeneratoren sind den Achsen des C-887 fest zugeordnet, siehe "Kommandierbare Elemente" (S. 27).

Ein Funktionsgenerator gibt auf Basis von definierten Kurvenformen absolute Zielpositionen für die Achsenbewegung aus und bestimmt damit das Dynamikprofil. Die Funktionsgeneratorausgabe eignet sich besonders für dynamische Anwendungen mit periodischen Achsenbewegungen.

Das folgende Blockdiagramm zeigt die Einbindung eines Funktionsgenerators im C-887.

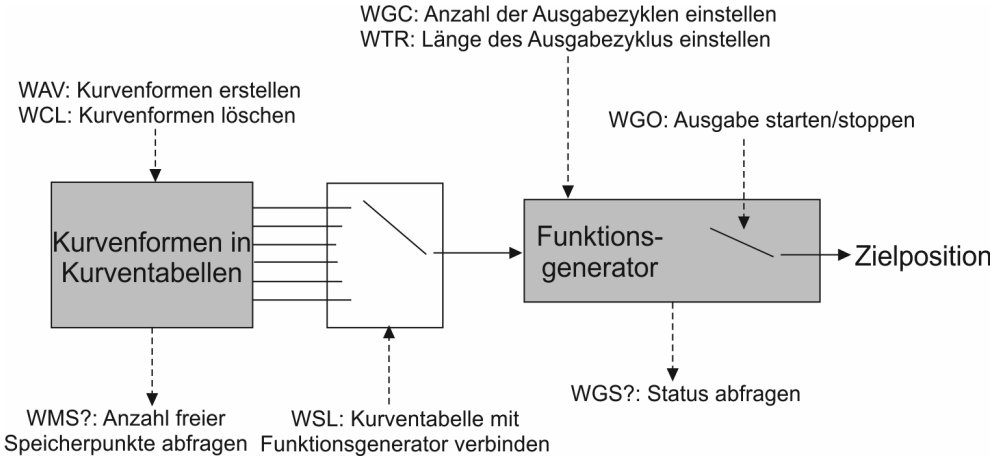


Abbildung 14: Blockdiagramm eines Funktionsgenerators

Kurvenformen können definiert und im flüchtigen Speicher des C-887 temporär in bis zu 100 Kurventabellen gespeichert werden (S. 106). Jede Kurventabelle enthält die Daten einer Kurvenform. Insgesamt sind 10.000.000 Datenpunkte für Kurventabellen verfügbar. Die Datenpunkte werden bei der Definition der Kurvenformen auf die einzelnen Kurventabellen verteilt.

Die Kurventabellen können den Funktionsgeneratoren und damit den Achsen beliebig zugeordnet werden (S. 113). Eine Kurventabelle kann gleichzeitig von mehreren Funktionsgeneratoren verwendet werden.

Die Anzahl der Ausgabezyklen (S. 114) und die Ausgaberate (S. 114) des Funktionsgenerators kann eingestellt werden.

INFORMATION

Mit der Makrofunktionalität des C-887 können Sie die Einstellungen des Funktionsgenerators dauerhaft im C-887 speichern. Zusätzlich können Sie ein Startup-Makro verwenden, um bei jedem Einschalten oder Neustart des C-887 den Funktionsgenerator zu konfigurieren und die Ausgabe zu starten, Details siehe "Anwendungstipps: Makros für Funktionsgenerator verwenden" (S. 124).

INFORMATION

Für die Arbeit mit dem Funktionsgenerator wird die Verwendung des PI Frequency Generator Tool oder des PI Wave Generator Tool empfohlen (beide verfügbar in PIMikroMove®).

Für die Arbeit mit PIMikroMove® sind keine Befehlskenntnisse erforderlich. Trotzdem wird empfohlen, dass Sie sich in diesem Kapitel mit dem Funktionsgenerator vertraut machen.

Alle Beispiele in diesem Kapitel können in PIMikroMove® oder PITerminal als Befehlsfolgen eingegeben werden.

Die Verwendung von PI Frequency Generator Tool und PI Wave Generator Tool ist im PIMikroMove®-Handbuch beschrieben (SM148E).

Sie können Makros verwenden, um Kurvenformen zu definieren und die Funktionsgeneratoren zu konfigurieren. Für die Arbeit mit Makros siehe "Controllermakros" (S. 127).

7.4.2 Befehle und Parameter für den Funktionsgenerator

Befehle

Folgende Befehle stehen für die Verwendung des Funktionsgenerators zur Verfügung:

Befehl	Syntax	Funktion
`GWD?`	GWD? [<StartPoint> <NumberOfPoints> [{<WaveTableID>}]]	Fragt den Inhalt der Kurventabellen (d. h. die Kurvenformen) ab.
`TWG?`	TWG?	Fragt die Anzahl der Funktionsgeneratoren (= Anzahl der Achsen) ab.
`WAV`	WAV <WaveTableID> <AppendWave> <WaveType> <WaveTypeParameters>	Definiert die Kurvenform.
`WAV?`	WAV? [{<WaveTableID> <WaveParameterID>}]	Fragt die aktuelle Länge der Kurventabellen ab (Punkteanzahl).
`WCL`	WCL {<WaveTableID>}	Löscht den Inhalt der Kurventabellen.
`WGC`	WGC {<WaveGenID> <Cycles>}	Stellt die Anzahl der Ausgabezyklen ein.

Befehl	Syntax	Funktion
`WGC?`	WGC? [{<WaveGenID>}]	Fragt die Anzahl der Ausgabezyklen ab.
`WGO`	WGO {<WaveGenID> <StartMode>}	Startet und stoppt die Ausgabe des Funktionsgenerators.
`WGO?`	WGO? [{<WaveGenID>}]	Fragt den Aktivierungszustand ab, der zuletzt für den Funktionsgenerator kommandiert wurde.
`WGR`	WGR	Startet die Datenaufzeichnung bei laufendem Funktionsgenerator neu.
`WGS?`	[<WaveGenID> [<InfoType>]]	Fragt den Status des Funktionsgenerators ab.
`WMS?`	[{<WaveTableID>}]	Fragt die Anzahl freier Speicherpunkte für die Kurventabelle ab.
`WSL`	WSL {<WaveGenID> <WaveTableID>}	Stellt die Verbindung zwischen Kurventabelle und Funktionsgenerator her.
`WSL?`	WSL? [{<WaveGenID>}]	Fragt die Verbindung zwischen Kurventabelle und Funktionsgenerator ab.
`WTR`	WTR {<WaveGenID> <WaveTableRate> <InterpolationType>}	Stellt die Ausgaberate des Funktionsgenerators ein (beeinflusst damit die Dauer eines Ausgabezyklus).
`WTR?`	WTR? [{<WaveGenID>}]	Fragt die Ausgaberate des Funktionsgenerators ab.
`#9`	#9	Fragt den aktuellen Aktivierungszustand des Funktionsgenerators ab.

Parameter

Die folgenden Parameter definieren die Eigenschaften des Funktionsgenerators:

Parameter	Beschreibung und mögliche Werte
Maximum Number Of Wave Points (ID 0x13000004)	Gesamtanzahl der für Kurvenformen verfügbaren Punkte Die Kurventabellen des C-887 haben insgesamt 10.000.000 Punkte. Die verfügbaren Punkte werden während der Definition von Kurvenformen mit dem Befehl `WAV` (S. 282) auf die Kurventabellen verteilt. Dieser Parameter ist schreibgeschützt.
Number of Wave Tables (ID 0x1300010A)	Anzahl der Kurventabellen zum Speichern von Kurvenformen Der C-887 hat 100 Kurventabellen. Dieser Parameter ist schreibgeschützt.

Parameter	Beschreibung und mögliche Werte
Start All Hexapod Wave Generators (ID 0x19002001)	Startverhalten der Funktionsgeneratoren für die Achsen der Bewegungsplattform des Hexapods (X, Y, Z, U, V, W): 0 = Jeder Funktionsgenerator, der gestartet werden soll, muss im WGO-Befehl addressiert werden (Standardeinstellung). 1 = Das Starten eines Funktionsgenerators startet alle Funktionsgeneratoren, die mit einer Kurventabelle verbunden sind. Diese Option ist nur aus Kompatibilitätsgründen vorhanden. Für die Achsen der Bewegungsplattform des Hexapods, deren Funktionsgenerator **nicht** gestartet wurde, wird immer die zuletzt gültige Zielposition kommandiert.

7.4.3 Kurvenform definieren

Das Definieren von Kurvenformen umfasst folgende Schritte:

- Optional: Informationen zu Kurventabellen abfragen (S. 105)
- Kurvenform in Kurventabelle erstellen (S. 106)
- Optional: Inhalt der Kurventabelle löschen (S. 107)

In diesem Handbuch finden Sie Beispiele für das Erstellen von Kurvenformen (S. 107).

INFORMATION

Mit der Makrofunktionalität des C-887 können Sie den Inhalt der Kurventabellen (= definierte Kurvenformen) dauerhaft im C-887 speichern, Details siehe "Anwendungstipps: Makros für Funktionsgenerator verwenden" (S. 124).

Optional: Informationen zu Kurventabellen abfragen

- Senden Sie den Befehl `SPA? 1 0x13000004`, um die Gesamtanzahl der Punkte abzufragen, die der C-887 für das Definieren von Kurvenformen in Kurventabellen bereitstellt.
- Senden Sie den Befehl `SPA? 1 0x1300010A`, um die Anzahl der im C-887 verfügbaren Kurventabellen abzufragen.
- Fragen Sie mit dem Befehl `WAV?` (S. 287) für die Kurventabellen die aktuelle Anzahl bereits definierter Kurvenformpunkte ab.
- Fragen Sie mit dem Befehl `GWD?` (S. 199) den aktuellen Inhalt der Kurventabellen ab (= bereits definierte Kurvenformen).

  Die Antwort enthält den Inhalt der Kurventabellen im GCS-Array-Format (siehe separates Handbuch für GCS Array, SM 146E).
- Fragen Sie mit dem Befehl `WMS?` (S. 292) für die Kurventabellen die Anzahl der verfügbaren freien Speicherpunkte ab.

Kurvenform in Kurventabelle erstellen

1. Stellen Sie sicher, dass die gewählte Kurventabelle **nicht** mit einem Funktionsgenerator verbunden ist, für den die Ausgabe gestartet ist. Details siehe "Funktionsgenerator konfigurieren" (S. 113) und "Funktionsgeneratorausgabe stoppen" (S. 116).
2. Erstellen Sie mit dem Befehl `WAV` (S. 282) (WAV + max. 12 Argumente) in der gewählten Kurventabelle die Kurvenform aus einzelnen Segmenten. Unterstützte Kurventypen:

   "PNT" (benutzerdefinierte Kurve)
   "SIN_P" (invertierte Kosinuskurve)
   "RAMP" (Rampenkurve)
   "LIN" (Kurve in Form einer einzelnen Abtastzeile)

   Die Kurvenform wird in die gewählte Kurventabelle im flüchtigen Speicher geschrieben. Details siehe "Beispiele für das Erstellen von Kurvenformen" (S. 107).

INFORMATION

Bei der Definition einer Kurvenform mit `WAV` (S. 282) werden die Zielpositionen und die resultierenden Geschwindigkeiten **nicht** geprüft. Die Prüfung erfolgt erst während der Funktionsgeneratorausgabe (S. 115).

INFORMATION

Bei der Definition einer Kurvenform prüft der C-887 **nicht** die Zielpositionen und die resultierenden Geschwindigkeiten. Die Prüfung erfolgt erst während der Funktionsgeneratorausgabe (S. 115). Wenn die Bewegung nicht möglich ist, stoppt der C-887 die Funktionsgeneratorausgabe und damit die Bewegung abrupt.

Um mit WAV (S. 282) geeignete Kurvenformen zu definieren und einen Abbruch der Funktionsgeneratorausgabe zu vermeiden, beachten Sie Folgendes:

- Berechnen Sie die Kurvenform extern möglichst exakt (z. B. mit NI LabVIEW, MATLAB oder Python), bevor Sie die Kurvenformdefinition mit WAV in den C-887 übertragen.
- Beachten Sie, dass die Kurvenform die Geschwindigkeit während der Bewegungen beeinflusst. Definieren Sie die Kurvenform so, dass die Spezifikationen der angeschlossenen Mechanik während der Funktionsgeneratorausgabe eingehalten werden. Die Geschwindigkeit wird unter anderem durch folgende Faktoren begrenzt:
  - Typ der Mechanik
  - Kombination der zu bewegenden Achsen
  - Aktuelle Einstellungen für Koordinatensystem und Drehpunkt
  - Amplitude der Bewegung
- Wenn die Kurvenform periodisch ausgegeben werden soll (d.h., mehr als ein Ausgabezyklus in Folge), muss das Ende der Kurvenform in Position und Geschwindigkeit identisch sein mit dem Anfang der Kurvenform.
- Wenn Sie eine Kurvenform aus mehreren Segmenten zusammensetzen, muss die Anfangsposition und Anfangsgeschwindigkeit eines anzuhängenden Segments jeweils an die Endposition und Endgeschwindigkeit des Vorgängersegments angepasst sein.
- Definieren Sie die Kurvenform möglichst so, dass sie mit der Standardausgaberate ausgegeben werden kann. Bei größeren Ausgaberaten interpoliert der C-887 fehlende Positionswerte, wodurch die ausgegebene Kurvenform möglicherweise nicht mehr exakt der Definition entspricht.

➢

Optional: Inhalt der Kurventabellen löschen

1. Stellen Sie sicher, dass die gewählte Kurventabelle **nicht** mit einem Funktionsgenerator verbunden ist, für den die Ausgabe gestartet ist. Details siehe "Funktionsgenerator konfigurieren" (S. 113) und "Funktionsgeneratorausgabe stoppen" (S. 116).
2. Löschen Sie mit dem Befehl `WCL` (S. 288) den Inhalt der Kurventabellen.

   Der komplette Inhalt der gewählten Kurventabelle wird gelöscht. Das segmentweise Löschen des Kurventabelleninhaltes ist **nicht** möglich.

INFORMATION

Beim Ausschalten oder Neustart des C-887 wird der Inhalt der Kurventabellen automatisch gelöscht.

Beispiele für das Erstellen von Kurvenformen

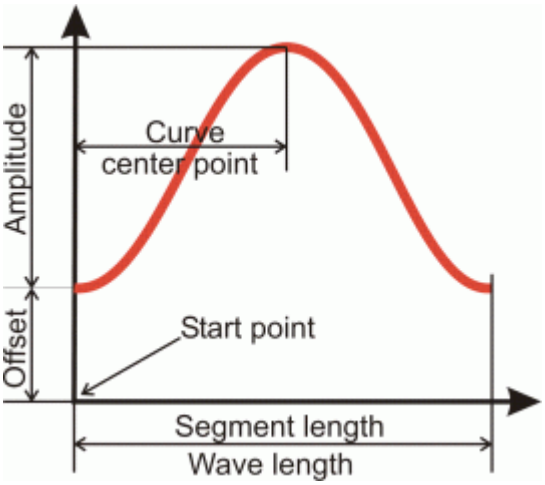
Die folgenden Beispiele helfen Ihnen beim Erstellen der Kurvenform.

Sinuskurve 1

- Symmetrische Sinuskurve mit Offset
- Segment überschreibt den Inhalt der Kurventabelle

Befehl: `WAV 2 X SIN_P 2000 0.2 0.1 2000 0 1000`

<WaveTableID> = 2
<AppendWave> = X
<WaveType> = SIN_P
<SegLength> = 2000
<Amp> = 0.2
<Offset> = 0.1
<WaveLength> = 2000
<StartPoint> = 0
<CurveCenterPoint> = 1000

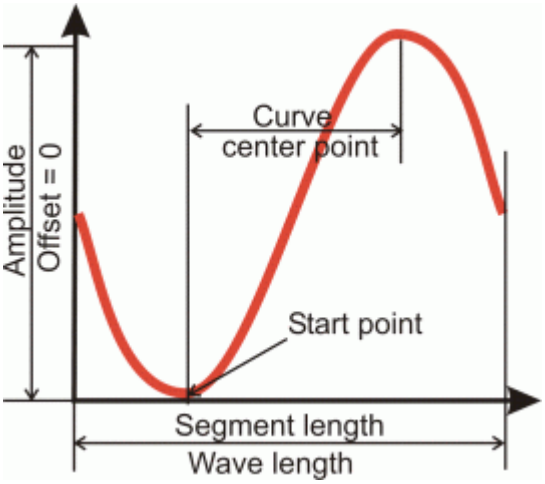


Sinuskurve 2

- Symmetrische Sinuskurve ohne Offset
- Segment überschreibt den Inhalt der Kurventabelle

Befehl: `WAV 2 X SIN_P 2000 0.3 0 2000 499 1000`

<WaveTableID> = 2
<AppendWave> = X
<WaveType> = SIN_P
<SegLength> = 2000
<Amp> = 0.3
<Offset> = 0
<WaveLength> = 2000
<StartPoint> = 499
<CurveCenterPoint> = 1000

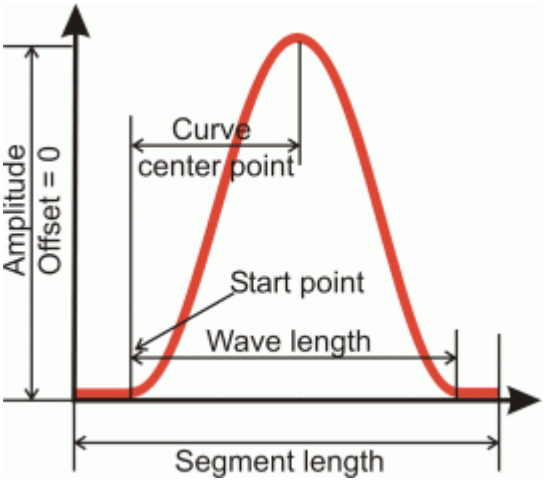


Sinuskurve 3

- Symmetrische Sinuskurve ohne Offset
- Segment wird an den Inhalt der Kurventabelle angehängt

Befehl: `WAV 2 & SIN_P 2000 0.25 0 1800 100 900`

<WaveTableID> = 2
<AppendWave> = &
<WaveType> = SIN_P
<SegLength> = 2000
<Amp> = 0.25
<Offset> = 0
<WaveLength> = 1800
<StartPoint> = 100
<CurveCenterPoint> = 900

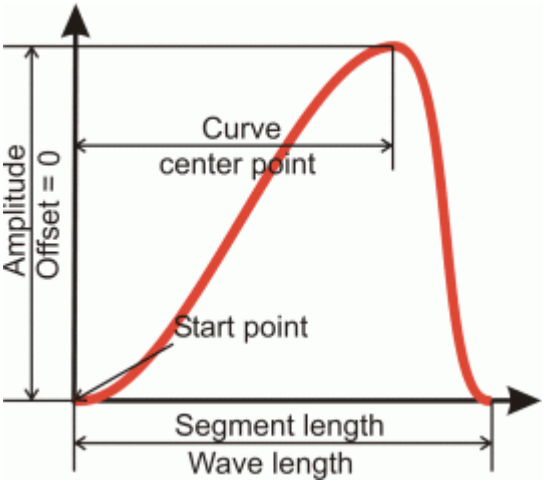


Sinuskurve 4

- Asymmetrische Kurve ohne Offset
- Segment überschreibt den Inhalt der Kurventabelle

Befehl: `WAV 3 X SIN_P 4000 0.2 0 4000 0 3100`

<WaveTableID> = 3
<AppendWave> = X
<WaveType> = SIN_P
<SegLength> = 4000
<Amp> = 0.2
<Offset> = 0
<WaveLength> = 4000
<StartPoint> = 0
<CurveCenterPoint> = 3100

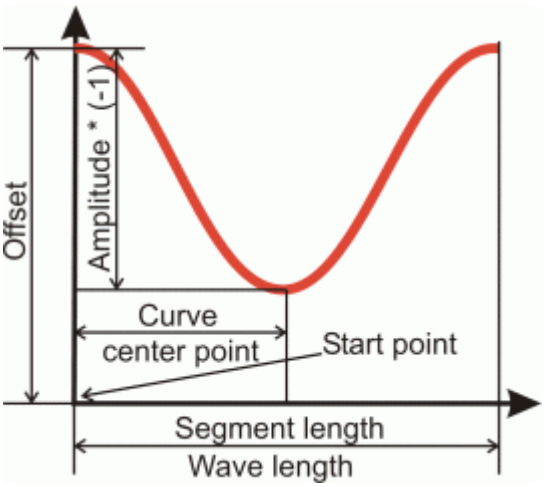


Sinuskurve 5

- Symmetrische Kurve mit negativer Amplitude
- Segment überschreibt den Inhalt der Kurventabelle

Befehl: `WAV 1 X SIN_P 1000 -0.3 0.45 1000 0 500`

<WaveTableID> = 1
<AppendWave> = X
<WaveType> = SIN_P
<SegLength> = 1000
<Amp> = –0.3
<Offset> = 0.45
<WaveLength> = 1000
<StartPoint> = 0
<CurveCenterPoint> = 500

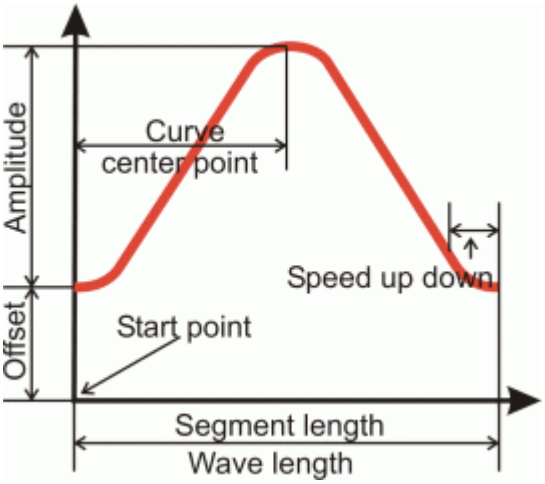


Rampenkurve 1

- Symmetrische Rampenkurve mit Offset
- Segment überschreibt den Inhalt der Kurventabelle

Befehl: `WAV 4 X RAMP 2000 0.2 0.1 2000 0 300 1000`

<WaveTableID> = 4
<AppendWave> = X
<WaveType> = RAMP
<SegLength> = 2000
<Amp> = 0.2
<Offset> = 0.1
<WaveLength> = 2000
<StartPoint> = 0
<SpeedUpDown> = 300
<CurveCenterPoint> = 1000

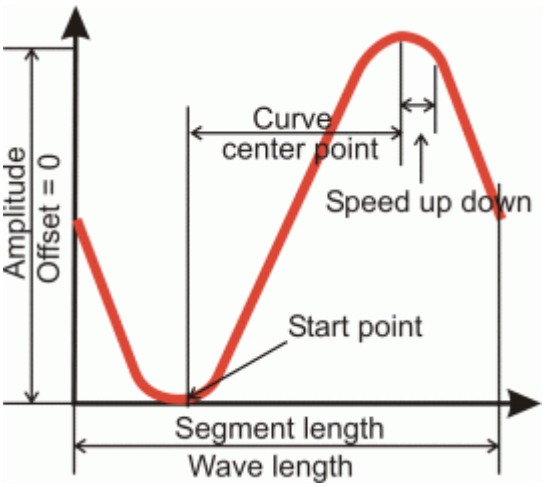


Rampenkurve 2

- Symmetrische Rampenkurve ohne Offset
- Segment überschreibt den Inhalt der Kurventabelle

Befehl: `WAV 4 X RAMP 2000 0.35 0 2000 499 300 1000`

<WaveTableID> = 4
<AppendWave> = X
<WaveType> = RAMP
<SegLength> = 2000
<Amp> = 0.35
<Offset> = 0
<WaveLength> = 2000
<StartPoint> = 499
<SpeedUpDown> = 300
<CurveCenterPoint> = 1000

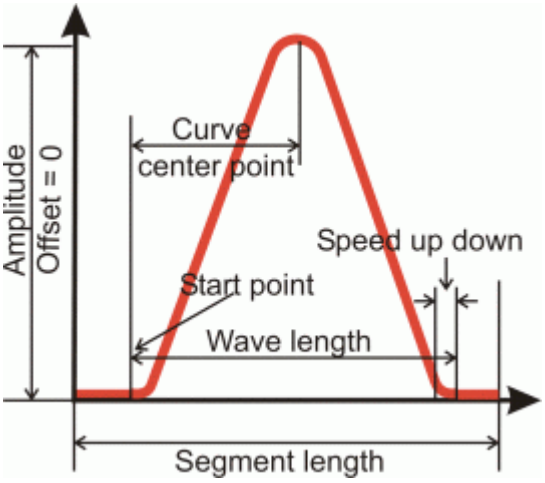


Rampenkurve 3

- Symmetrische Rampenkurve ohne Offset
- Segment überschreibt den Inhalt der Kurventabelle

Befehl: `WAV 5 X RAMP 2000 0.15 0 1800 120 150 900`

<WaveTableID> = 5
<AppendWave> = X
<WaveType> = RAMP
<SegLength> = 2000
<Amp> = 0.15
<Offset> = 0
<WaveLength> = 1800
<StartPoint> = 120
<SpeedUpDown> = 150
<CurveCenterPoint> = 900

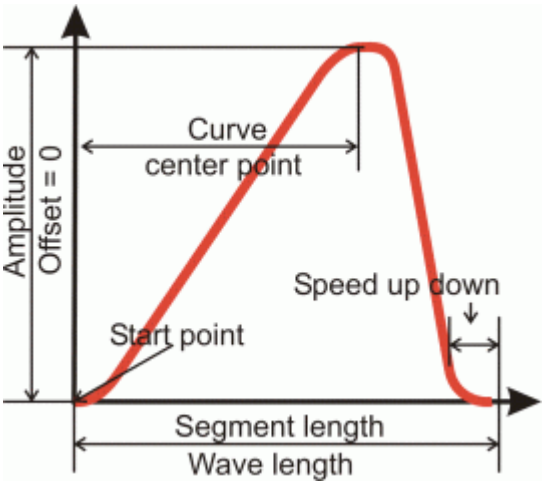


Rampenkurve 4

- Asymmetrische Rampenkurve ohne Offset
- Segment wird an den Inhalt der Kurventabelle angehängt

Befehl: `WAV 5 & RAMP 3000 0.35 0 3000 0 200 2250`

<WaveTableID> = 5
<AppendWave> = &
<WaveType> = RAMP
<SegLength> = 3000
<Amp> = 0.35
<Offset> = 0
<WaveLength> = 3000
<StartPoint> = 0
<SpeedUpDown> = 200
<CurveCenterPoint> = 2250

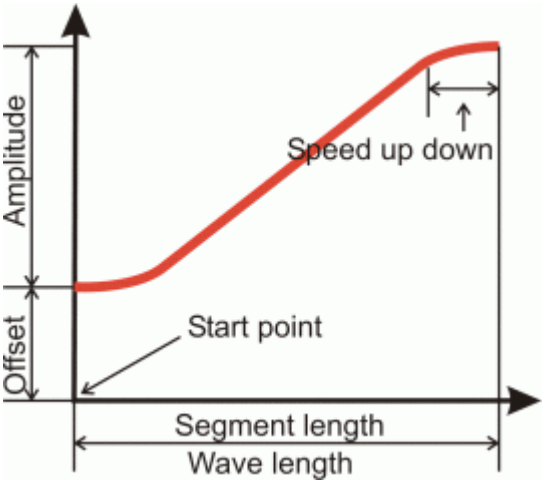


Einzelne Abtastzeile 1

- Abtastzeile mit Offset
- Segment überschreibt den Inhalt der Kurventabelle

Befehl: `WAV 1 X LIN 1500 0.3 0.15 1500 0 370`

<WaveTableID> = 1
<AppendWave> = X
<WaveType> = LIN
<SegLength> = 1500
<Amp> = 0.3
<Offset> = 0.15
<WaveLength> = 1500
<StartPoint> = 0
<SpeedUpDown> = 370

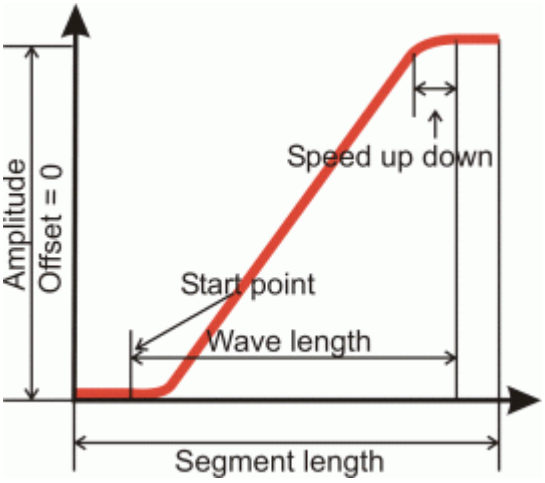


Einzelne Abtastzeile 2

- Abtastzeile ohne Offset
- Segment überschreibt den Inhalt der Kurventabelle

Befehl: `WAV 2 X LIN 1500 0.4 0 1100 210 180`

<WaveTableID> = 2
<AppendWave> = X
<WaveType> = LIN
<SegLength> = 1500
<Amp> = 0.4
<Offset> = 0
<WaveLength> = 1100
<StartPoint> = 210
<SpeedUpDown> = 180

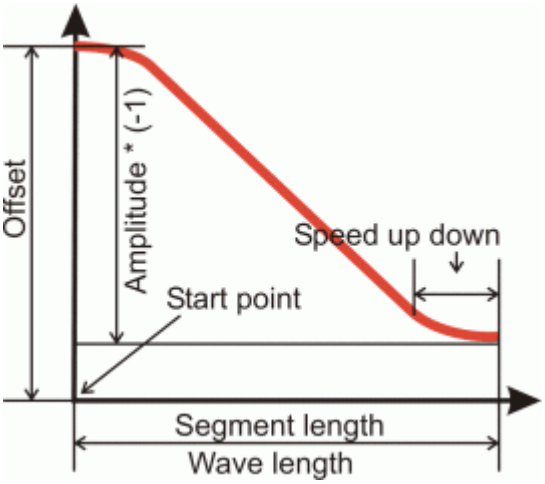


Einzelne Abtastzeile 3

- Abtastzeile mit negativer Amplitude
- Segment wird an den Inhalt der Kurventabelle angehängt

Befehl: `WAV 2 & LIN 3000 -0.4 0.5 3000 0 650`

<WaveTableID> = 2
<AppendWave> = &
<WaveType> = LIN
<SegLength> = 3000
<Amp> = –0.4
<Offset> = 0.5
<WaveLength> = 3000
<StartPoint> = 0
<SpeedUpDown> = 650



7.4.4 Funktionsgenerator konfigurieren

Das Konfigurieren des Funktionsgenerators umfasst folgende Schritte:

- Funktionsgenerator und Kurventabelle verbinden oder trennen (S. 113)
- Optional: Anzahl der Ausgabezyklen einstellen (S. 114)
- Optional: Ausgaberate einstellen (S. 114)

In diesem Handbuch finden Sie ein Beispiel für das Einstellen der Ausgaberate (S. 114).

Funktionsgenerator und Kurventabelle verbinden oder trennen

➢ Fragen Sie mit dem Befehl `WSL?` (S. 295) die aktuelle Verbindung von Funktionsgenerator und Kurventabelle ab.

➢ Verbinden oder trennen Sie Funktionsgenerator und Kurventabelle:

a) Stellen Sie sicher, dass für den gewählten Funktionsgenerator **nicht** die Ausgabe gestartet ist. Details siehe "Funktionsgeneratorausgabe stoppen" (S. 116).

b) Verbinden Sie mit dem Befehl `WSL` (S. 295) die gewählte Kurventabelle mit dem gewählten Funktionsgenerator, oder trennen Sie die Verbindung des ausgewählten Generators zu einer Kurventabelle.

Zwei oder mehr Generatoren können mit derselben Kurventabelle verbunden sein, ein Generator kann jedoch nicht mit mehreren Kurventabellen verbunden sein.

Optional: Anzahl der Ausgabezyklen einstellen

Die Standardeinstellung für die Anzahl der Ausgabezyklen ist 0. Mit der Standardeinstellung erfolgt die Ausgabe der Kurvenform ohne zeitliche Begrenzung, bis sie mit den Befehlen `WGO` (S. 289) oder `HLT` (S. 204) oder `#24` (S. 156) oder `STP` (S. 267) gestoppt wird.

- ➢ Senden Sie den Befehl `WGC?` (S. 289), um die aktuelle Einstellung für die Anzahl der Ausgabezyklen des Funktionsgenerators abzufragen.
- ➢ Stellen Sie mit dem Befehl `WGC` (S. 288) die Anzahl der Ausgabezyklen des Funktionsgenerators ein.

Optional: Ausgaberate einstellen

Durch das Anpassen der Ausgaberate können die einzelnen Ausgabezyklen der Kurvenform verlängert werden. Die Dauer eines Ausgabezyklus für die Kurvenform kann wie folgt berechnet werden:

Ausgabedauer = Servozykluszeit * Ausgaberate * Anzahl der Punkte

wobei

- die Servozykluszeit für den C-887 durch den Parameter 0x0E000200 angegeben wird (in Sekunden)
- die Ausgaberate der Anzahl der Servozyklen entspricht, über die sich die Ausgabe eines Kurvenpunkts zeitlich erstreckt; ganzzahliges Vielfaches von 10 (Minimum und Standard: 10, Maximum: 1000)
- die Anzahl der Punkte der Länge der Kurvenform (d. h. der Länge der Kurventabelle) entspricht

Die Ausgabe der Kurvenpunkte erfolgt bei Ausgaberaten > 10 standardmäßig mit Interpolation, um Positionssprünge der Achsen zu vermeiden.

- ➢ Senden Sie den Befehl `WTR?` (S. 298), um die aktuellen Einstellungen für Ausgaberate und Interpolation abzufragen.
- ➢ Stellen Sie mit dem Befehl `WTR` (S. 296) die Ausgaberate und die Interpolation ein.

Für die einzelnen Funktionsgeneratoren des C-887 können unterschiedliche Ausgaberaten eingestellt werden.

INFORMATION

- ➢ Bei Ausgaberaten >10 interpoliert der C-887 fehlende Positionswerte linear. Die Kurvenform ist dann an den Stützstellen nicht mehr differenzierbar. Daraus können größere Geschwindigkeits- und Beschleunigungssprünge resultieren, die Schwingungen der Mechanik anregen. Verwenden Sie deshalb möglichst die Standardausgaberate (10).

INFORMATION

Wenn beim Einstellen der Ausgaberate mit dem Befehl `WTR` die Funktionsgeneratorkennung 0 verwendet wird, wird die Ausgaberate für alle Funktionsgeneratoren auf denselben Wert gesetzt.

Beispiel für das Einstellen der Ausgaberate

Aktion	Befehl	Ergebnis
Sinuskurve für Kurventabelle 2 definieren.	`WAV 2 X SIN_P 2000 0.2 0.1 2000 0 1000`	Die Länge der Kurvenform und damit die Anzahl der Punkte in der Kurventabelle ist 2000.
Servozykluszeit des C-887 auslesen.	`SPA? 1 0x0E000200`	Die Servozykluszeit des C-887 ist 100 µs
Aktuelle Ausgaberate auslesen.	`WTR?`	Standardwert für die Ausgaberate = 10 (jeder Punkt in der Kurventabelle wird während 10 Servozyklen ausgegeben) Dauer eines Ausgabezyklus (siehe Berechnungsformel oben): 0,0001 s • 10 • 2000 = 2 s
Anzahl der Servozyklen pro Punkt der Kurventabelle für alle Funktionsgeneratoren verdreifachen.	`WTR 0 30 1`	Dauer eines Ausgabezyklus (siehe Berechnungsformel oben): 0,0001 s • 30 • 2000 = 6 s Zwischen der Ausgabe der Punkte interpoliert der C-887 linear.

7.4.5 Ausgabe starten und stoppen

Der Funktionsgenerator gibt absolute Zielpositionen aus.

- Funktionsgeneratorausgabe starten (S. 116)
- Funktionsgeneratorausgabe stoppen (S. 116)
- Optional: Aktivierungszustand und Status des Funktionsgenerators abfragen (S. 117)
- Optional: Datenaufzeichnung während Funktionsgeneratorausgabe starten (S. 117)

In diesem Handbuch finden Sie Beispiele für das Starten/Stoppen der Funktionsgeneratorausgabe (S. 118).

INFORMATION

Wenn die Funktionsgeneratorausgabe aktiv ist, sind Befehle zum Starten oder Konfigurieren von Bewegungen sowie das Ausführen entsprechender Makros **nicht** zulässig.

INFORMATION

Während der Funktionsgeneratorausgabe prüft der C-887 ständig, ob die Bewegung noch möglich ist. In folgenden Fällen stoppt der C-887 die Bewegung abrupt und setzt einen Fehlercode:

- Die auszugebenden Zielpositionen können nicht erreicht werden.
- Die erforderliche Geschwindigkeit kann nicht erreicht werden.
- Die Bewegung würde eine Kollision verursachen.

➢ Fragen Sie mit dem Befehl `WGS?` den aktuellen Status der Funktionsgeneratoren ab, insbesondere den Index der Kurvenformpunkte, an denen ein Fehler aufgetreten ist.

➢ Fragen Sie mit dem Befehl `ERR?` (S. 176) den Fehlercode des zuletzt aufgetretenen Fehlers ab.

Voraussetzungen

✓ Sie haben die gewünschte Kurvenform erstellt (S. 106).

✓ Sie haben den Funktionsgenerator mit der entsprechenden Kurventabelle verbunden (S. 113).

Funktionsgeneratorausgabe starten

➢ Starten Sie die Funktionsgeneratorausgabe mit dem Befehl `WGO` (S. 289).

Alle Funktionsgeneratoren, deren Ausgabe gleichzeitig aktiv sein soll, müssen im selben Befehl gestartet werden. Für die Achsen der Bewegungsplattform des Hexapods, deren Funktionsgenerator nicht gestartet wurde, wird immer die zuletzt gültige Zielposition kommandiert.

Die Ausgabe erfolgt synchron mit den Servozyklen des C-887.

Beim Start der Funktionsgeneratorausgabe wird automatisch ein Datenaufzeichnungszyklus gestartet.

Funktionsgeneratorausgabe stoppen

➢ Stoppen Sie die Funktionsgeneratorausgabe durch Senden eines der folgenden Befehle:

– `WGO` (S. 289) stoppt die angegebenen Funktionsgeneratoren

– `STP` (S. 267) stoppt alle Funktionsgeneratoren

– `#24` (S. 156) stoppt alle Funktionsgeneratoren

– `HLT` (S. 204) stoppt die Funktionsgeneratoren der angegebenen Achsen

Wenn die Funktionsgeneratorausgabe durch Senden von `STP`, `#24` oder `HLT` gestoppt wird, setzt der C-887 den Fehlercode 10 (Abfrage mit dem Befehl `ERR?` (S. 176)).

Wenn die Anzahl der Ausgabezyklen begrenzt wurde (S. 114), wird die Funktionsgeneratorausgabe nach Erreichen der vorgegebenen Zyklenzahl automatisch gestoppt.

INFORMATION

Das Beenden der PC-Software stoppt **nicht** die Funktionsgeneratorausgabe.

Optional: Aktivierungszustand des Funktionsgenerators abfragen

➢ Fragen Sie mit dem Befehl `#9` (S. 155) ab, ob die Funktionsgeneratorausgabe läuft.

➢ Fragen Sie mit dem Befehl `WGO?` (S. 291) die zuletzt kommandierten Start-/Stopp-Einstellungen des Funktionsgenerators ab.

Die Antwort auf `WGO?` wird auch durch das Stoppen der Funktionsgeneratorausgabe mit `#24` (S. 156), `STP` (S. 267) oder `HLT` (S. 204) und Ende der Ausgabe nach einer festgelegten Anzahl von Ausgabezyklen beeinflusst.

➢ Fragen Sie mit dem Befehl `WGS?` den Status des Funktionsgenerators ab:

– Läuft der Funktionsgenerator?

– Wie viele Zyklen hat der Funktionsgenerator seit dem letzten Start ausgegeben?

– Fehlercode des zuletzt während der Ausgabe aufgetretenen Fehlers

– Index des Kurvenformpunktes, an dem der Fehler aufgetreten ist

Optional: Datenaufzeichnung während Funktionsgeneratorausgabe starten

➢ Starten Sie die Datenaufzeichnung während der Funktionsgeneratorausgabe, indem Sie den Befehl `WGR` (S. 291) senden.

Beim Starten der Funktionsgeneratorausgabe (S. 116) startet automatisch ein erster Datenaufzeichnungszyklus.

Die aufgezeichneten Daten können mit dem Befehl `DRR?` (S. 172) ausgelesen werden. Weitere Informationen siehe "Datenrekorder" (S. 99).

Beispiele für das Starten/Stoppen der Funktionsgeneratorausgabe

Aktion	Befehl	Ergebnis
Sinuskurve für Kurventabelle 4 definieren.	`WAV 4 X SIN_P 2000 1 0 2000 0 1000`	Die Länge der Kurvenform und damit die Anzahl der Punkte in der Kurventabelle ist 2000.
Sinuskurve für Kurventabelle 5 definieren.	`WAV 5 X SIN_P 2000 0.2 0 2000 0 1000`	Die Länge der Kurvenform und damit die Anzahl der Punkte in der Kurventabelle ist 2000.
Funktionsgenerator 1 (Achse X) mit der Kurventabelle 4 verbinden. Funktionsgenerator 3 (Achse Z) mit der Kurventabelle 5 verbinden.	`WSL 1 4 3 5`	Voraussetzung für Ausgabe des Funktionsgenerators erfüllt: ohne Zuweisung einer Kurventabelle ist keine Ausgabe des Funktionsgenerators möglich.
Funktionsgeneratoren 1 und 3 starten.	`WGO 1 1 3 1`	Die in den Kurventabellen 4 und 5 definierten Kurvenformen werden ausgegeben. Für die Achsen Y, U, V und W der Bewegungsplattform des Hexapods, deren Funktionsgeneratoren nicht gestartet wurden, werden die zuletzt gültigen Zielpositionen kommandiert.
Funktionsgeneratoren 1 und 3 stoppen.	`WGO 1 0 3 0`	Die Ausgabe der Kurvenpunkte (und damit die Bewegung des Hexapods) wird gestoppt.

Alle Befehle bis auf die letzten beiden Zeilen wie im Beispiel oben; die letzten beiden Zeilen werden durch die folgenden Befehle ersetzt:

Aktion	Befehl	Ergebnis
Anzahl der Ausgabezyklen für Funktionsgeneratoren 1 und 3 auf 100 begrenzen.	`WGC 1 100 3 100`	Ein Befehl zum Stoppen der Funktionsgeneratoren ist nicht erforderlich.
Funktionsgeneratoren 1 und 3 starten.	`WGO 1 1 3 1`	Die Ausgabe der Funktionsgeneratoren endet automatisch nach 100 Ausgabezyklen.

7.4.6 Anwendungstipps: Kundenspezifische Daten laden

Sie können kundenspezifische Daten (Zeitreihen) aus Dateien vom PC in den C-887 laden und als benutzerdefinierte Kurven verwenden. Vor dem Laden müssen Sie die Daten in das GCS-Array-Format umwandeln. Für das Laden der umgewandelten Daten in den C-887 verwenden Sie das PI Wave Generator Tool oder das PI Frequency Generator Tool (beide verfügbar in PIMikroMove®). Die Verwendung des PI Wave Generator Tool ist nachfolgend beschrieben; für die Verwendung des PI Frequency Generator Tool siehe das PIMikroMove®-Handbuch (SM148E).

Daten in das GCS-Array-Format umwandeln

Das GCS-Array-Format basiert auf ASCII-Zeichen. Eine GCS-Array-Datei besteht aus einem Headerabschnitt und einem Datentabellenabschnitt. Als Dezimaltrennzeichen kann ein Komma oder ein Punkt verwendet werden. Eine ausführliche Beschreibung des GCS-Array-Formats siehe separates Handbuch (SM146E).

Wandeln Sie die Daten Ihrer Zeitreihe wie folgt in das GCS-Array-Format um:

1. Erzeugen Sie eine Textdatei mit der Dateiendung ".dat" auf dem PC, z. B. "New_GCS_Array.dat".
2. Fügen Sie den Header zur Datei hinzu. Struktur und Inhalt des Headers müssen wie folgt aufgebaut sein:

   Template für den GCS-Array-Header:

   `[GCS_ARRAY` **Target Positions XY**`]`

   `# VERSION = 1`

   `# TYPE = 1`

   `# SEPARATOR =` **32**

   `# DIM = 2`

   `# SAMPLE_TIME =` **0,001**

   `# NDATA =` **256**

   `#`

   `# NAME0 =` **TARGET POSITION X**

   `# NAME1 =` **TARGET POSITION Y**

   `#`

   `# DISP_UNIT0 =` **mm**

   `# DISP_UNIT1 =` **mm**

   `# END_HEADER`

Passen Sie die folgenden Angaben im Header an Ihre Daten an (**markiert** in der Liste oben):

`DIM`
Anzahl der Spalten in der Datentabelle (d. h. Anzahl der Zeitreihen), Minimum ist 1

`NDATA`
Anzahl der Zeilen in der Datentabelle (d. h. Anzahl der Datenpunkte pro Zeitreihe)

`NAME%`
Optional: Beschreibung der Spalte % (wobei % ein positiver ganzzahliger Wert < DIM ist; die Zählung beginnt mit 0)

`DISP_UNIT%`
Optional: Beschreibung der physikalischen Einheit für Spalte % (wobei % ein positiver ganzzahliger Wert < DIM ist; die Zählung beginnt mit 0)

`SAMPLE_TIME`
Abtastintervall der Daten (in s), d. h. der Zeitabstand zwischen zwei Datenpunkten der Zeitreihe

`SEPARATOR`
Separator für die Spalten in der Datentabelle. Muss als dezimales ASCII-Zeichen angegeben werden (z. B. 9 für TAB oder 32 für SPACE)

`[GCS_ARRAY %]`
Optional: % ist der Name der Datentabelle

3. Fügen Sie Ihre Daten zur Datei hinzu gemäß den Headerangaben für `DIM`, `NDATA` und `SEPARATOR`:

   Datentabellenabschnitt der GCS-Array-Datei, Leerzeichen werden hier als SP dargestellt:

   `Value1_TimeSeries0`SP`Value1_TimeSeries1`SP…

   `Value2_TimeSeries0`SP`Value2_TimeSeries1`SP…

   `Value3_TimeSeries0`SP`Value3_TimeSeries1`SP…

   …

   Das folgende Beispiel zeigt ein komplettes GCS-Array mit Daten für zwei Zeitreihen. Die erste Zeitreihe "TARGET POSITION X" enthält die Werte 1/2/3, und die zweite Zeitreihe "TARGET POSITION Y" enthält die Werte 0.1/0.2/0.3.

```
# REM data recorded with C-887 controller
#
# VERSION = 1
# TYPE = 1
# SEPARATOR = 32
# DIM = 2
# SAMPLE_TIME = 0.009009
# NDATA = 3
```

```
#
# NAME0 = TARGET POSITION X
# NAME1 = TARGET POSITION Y
#
# DISP_UNIT0 = mm
# DISP_UNIT1 = mm
# END_HEADER
1 0.1
2 0.2
3 0.3
```

Das Ergebnis der Umwandlung ist eine GCS-Array-Datei mit der Dateiendung ".dat", die die Zeitreihen-Daten enthält. Die Datei kann mit dem PI Wave Generator Tool (siehe unten) oder dem PI Frequency Generator Tool in den C-887 importiert werden.

GSC-Array-Daten laden

Das PI Wave Generator Tool ist in PIMikroMove® verfügbar. Weitere Informationen zum PI Wave Generator Tool siehe PIMikroMove®-Handbuch (SM148E).

1. Öffnen Sie das PI Wave Generator Tool aus dem Hauptfenster von PIMikroMove® über den Menüeintrag ***C-887 > Show wave generator...***.

2. Wählen Sie im Hauptfenster des PI Wave Generator Tools die Schaltfläche ***Load Data Set*** , um das Fenster ***Load Data Set for Wave Tables*** zu öffnen.

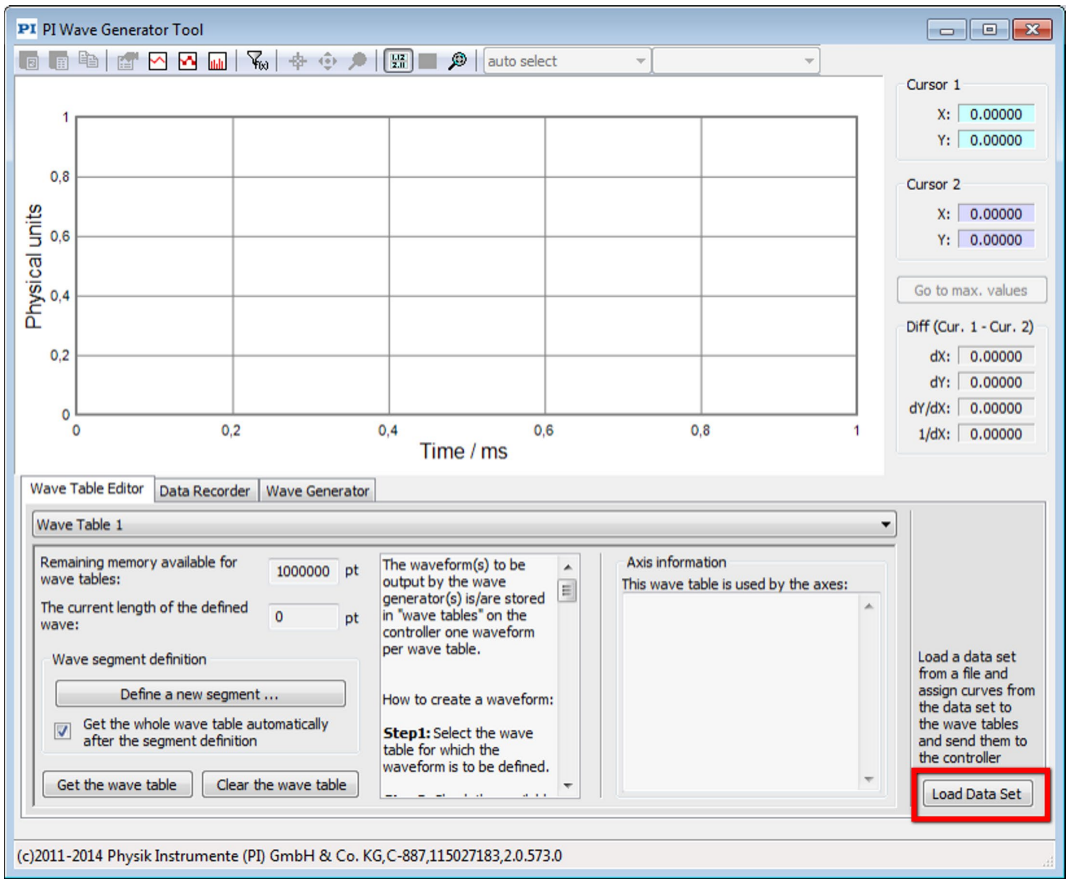


Abbildung 15: Hauptfenster des PI Wave Generator Tools

3. Importieren Sie die GCS-Array-Datei im Fenster ***Load Data Set for Wave Tables***.

Die Zeitreihen-Daten, die aus der Datei geladen wurden, werden im Grafikfeld angezeigt. Die Abbildung unten zeigt die Daten aus dem Beispiel in "Daten in das GCS-Array-Format umwandeln".

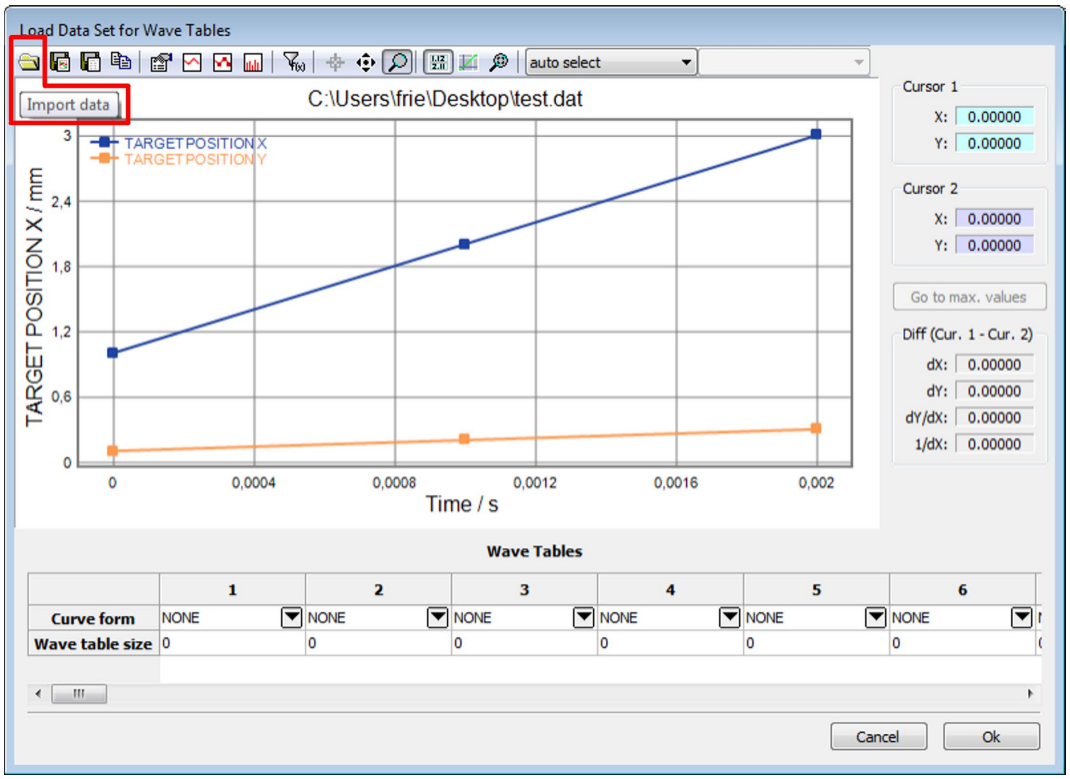


Abbildung 16: Fenster "Load Data Set for Wave Tables" mit grafischer Anzeige der geladenen Daten

4. Weisen Sie die einzelnen Zeitreihen den Kurventabellen des C-887 zu (siehe Abbildung unten).

Beachten Sie, dass die Daten erst dann an den C-887 gesendet werden, wenn Sie in der unteren rechten Ecke des Fensters ***Ok*** wählen.

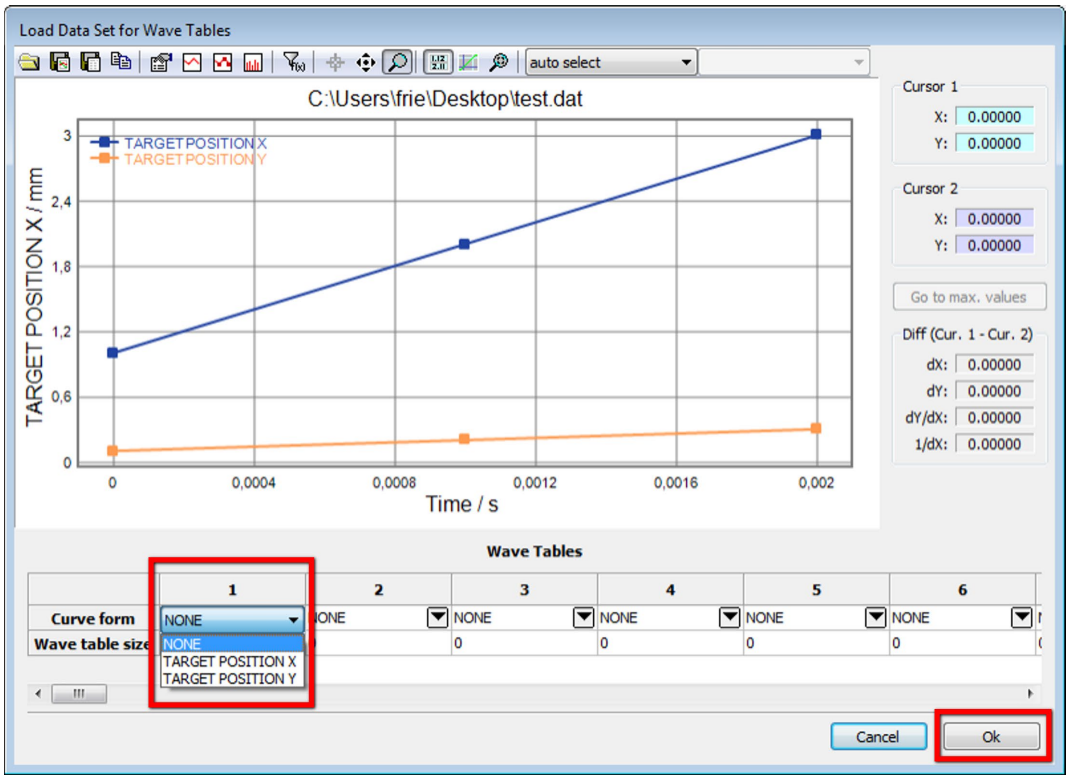


Abbildung 17: Zuweisung der geladenen Daten zu Kurventabellen, hier für Kurventabelle 1

7.4.7 Anwendungstipps: Makros für Funktionsgenerator verwenden

Die folgenden Einstellungen für den Funktionsgenerator lassen sich nur im flüchtigen Speicher des C-887 ändern und gehen beim Ausschalten oder Neustart des C-887 verloren:

- Inhalt der Kurventabellen (definiert mit WAV)
- Verbindung von Kurventabellen mit Funktionsgeneratoren (eingestellt mit WSL)
- Ausgaberate (eingestellt mit WTR)
- Anzahl der Ausgabezyklen (eingestellt mit WGC)

Mit der Makrofunktionalität des C-887 können Sie die Einstellungen des Funktionsgenerators dauerhaft im C-887 speichern. Zusätzlich können Sie ein Startup-Makro verwenden, um bei jedem Einschalten oder Neustart des C-887 den Funktionsgenerator zu konfigurieren und die Ausgabe zu starten.

Weitere Informationen zur Makrofunktionalität siehe "Controllermakros" (S. 127).

Im folgenden Beispiel werden zwei Makros verwendet, um für die Achsen U und V (Funktionsgeneratoren 4 und 5) des Hexapods die Funktionsgeneratorausgabe zu konfigurieren und zu starten:

- Das Makro "UVdata" enthält die Definition der Kurvenformen für die Kurventabellen 1 und 2 (WAV-Befehle).
- Das Makro "Start" führt die folgenden Aktionen aus:
  a) Durch Aufruf des Makros "UVdata" die Kurvenformen in die Kurventabellen 1 und 2 schreiben.
  b) Kurventabelle 1 mit Funktionsgenerator 4 verbinden und Kurventabelle 2 mit Funktionsgenerator 5 (Befehl WSL).
  c) Ausgaberate für Funktionsgeneratoren 4 und 5 auf 10 setzen, mit linearer Interpolation (Befehl WTR; die Einstellung für die Interpolation muss ebenfalls gesetzt werden, wird aber bei der Ausgabe mit Ausgaberate 10 ignoriert).
  d) Anzahl der Ausgabezyklen für Funktionsgeneratoren 4 und 5 auf 100 setzen (Befehl WGC).
  e) Optional, aber empfohlen, um Sprünge der Achsen zu vermeiden: Achsen U und V zur Startposition der Funktionsgeneratorausgabe kommandieren. Danach warten, bis die Zielpositionen erreicht sind.
  f) Ausgabe für Funktionsgeneratoren 4 und 5 - und damit die Bewegung der Achsen U und V - starten (Befehl WGO).

Inhalt des Makro "UVdata":

```
WAV 1 X PNT 1 1 0
WAV 1 & PNT 1 1 5.654625319357E-06
WAV 1 & PNT 1 1 3.091004951757E-05
[…]
WAV 1 & PNT 1 1 0.000148233661

WAV 2 X PNT 1 1 0
WAV 2 & PNT 1 1 -4.742444189387E-06
WAV 2 & PNT 1 1 -3.027352414704E-05
[…]
WAV 2 & PNT 1 1 -0.000257643502
```

Inhalt des Makros "Start", wobei `startposU` und `startposV` die Startposition der Funktionsgeneratorausgabe bezeichnen:

```
MAC START UVdata
WSL 4 1 5 2
```

```
WTR 4 10 1 5 10 1
WGC 4 100 5 100
MOV U startposU V startposV
WAC ONT? U = 1
WAC ONT? V = 1
WGO 4 1 5 1
```

Anmerkungen:

Im Beispielmakro "UVdata" wird zur besseren Übersicht pro WAV-Befehl nur ein Kurvenpunkt (= Zielposition) angegeben. Für schnellere Verarbeitung wird empfohlen, mehr als einen Punkt pro WAV-Befehl zu definieren (pro Befehlszeile sind maximal 256 Zeichen zulässig).

Kurvenformdefinitionen können leicht erstellt werden, indem der Inhalt des Logfensters von PIMikroMove® in ein Makro kopiert wird:

1. Öffnen Sie das Logfenster über den Menüeintrag ***C-887 > Log window...*** im Hauptfenster von PIMikroMove®.
2. Öffnen Sie aus dem Menü ***C-887*** im Hauptfenster von PIMikroMove® das PI Frequency Generator Tool oder das PI Wave Generator Tool.
3. Definieren Sie mit dem Tool eine Kurvenform und senden Sie die Kurvenform an den C-887.

   Die gesendeten Befehle werden im Logfenster aufgelistet.

   Beispielsweise können Sie mit dem PI Frequency Generator Tool eine Sinuskurve mit einer bestimmten Frequenz definieren. Die entsprechenden Befehle WAV … PNT können Sie dann aus dem Logfenster in ein Makro kopieren. Die Befehle WAV … PNT im Logfenster werden durch die Funktion PI_WAV_PNT() der von PIMikroMove® verwendeten PI GCS2 DLL erzeugt. Die Funktion PI_WAV_PNT() verteilt die Kurvenpunkte automatisch auf die einzelnen Befehle WAV … PNT.

Wenn das Beispielmakro "Start" als Startup-Makro verwendet wird, kann vor dem Start der Funktionsgeneratorausgabe eine Referenzierungsfahrt erforderlich sein. In diesem Fall müssen die folgenden Zeilen vor der Zeile `MOV U startposU V startposV` in das Makro eingefügt werden:

```
FRF X
WAC FRF? X = 1
```

Die Zeilen starten eine Referenzierungsfahrt und warten, bis die Referenzierungsfahrt erfolgreich beendet wurde.

7.5 Controllermakros

7.5.1 Übersicht: Makrofunktionalitäten und Beispielmakros

Der C-887 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-887 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 aufrufen.
- Variablen (S. 135) können für das Makro und im Makro selbst gesetzt und in verschiedenen Operationen verwendet werden.
- Eingangssignale können für Bedingungen und Variablen ausgewertet werden.

In diesem Handbuch finden Sie Beispielmakros für folgende Aufgaben:

- Achse hin und her bewegen (S. 131)
- Achse mit variablem Verfahrweg hin und her bewegen (S. 133)
- Referenzierungsfahrt für Hexapod durch Startup-Makro auslösen (S. 135)

7.5.2 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. 161)	ADD <Variable> <FLOAT1> <FLOAT2>	Kann nur in Makros verwendet werden. Addiert zwei Werte und speichert das Ergebnis als Variable (S. 135).
`CPY` (S. 163)	CPY <Variable> <CMD?>	Kann nur in Makros verwendet werden. Kopiert eine Antwort auf einen Befehl in eine Variable (S. 135).
`DEL` (S. 165)	DEL <uint>	Kann nur in Makros verwendet werden. Bewirkt eine Verzögerung um <uint> Millisekunden.
`JRC` (S. 210)	JRC <Jump> <CMD?> <OP> <Value>	Kann nur in Makros verwendet werden. Löst einen relativen Sprung des Makroausführungszeigers in Abhängigkeit von einer Bedingung aus.

Befehl	Syntax	Funktion
MAC (S. 237)	MAC BEG <macroname>	Startet die Aufzeichnung eines Makros mit dem Namen *macroname* auf dem Controller. *macroname* kann aus bis zu 8 Zeichen bestehen.
	MAC DEF <macroname>	Legt das angegebene Makro als Startup-Makro fest.
	MAC DEF?	Fragt das Startup-Makro ab.
	MAC DEL <macroname>	Löscht das angegebene Makro.
	MAC END	Stoppt die Makroaufzeichnung.
	MAC ERR?	Meldet den letzten Fehler, der während der Ausführung eines Makros auftrat.
	MAC FREE?	Fragt nach dem freien Speicherplatz für die Makroaufzeichnung.
	MAC NSTART <macroname> <uint> [{<String>}]	Startet das angegebene Makro n-mal hintereinander (n = Anzahl der Ausführungen). Mit <String> können die Werte lokaler Variablen für das Makro gesetzt werden.
	MAC START <macroname> [{<String>}]	Startet eine Ausführung des angegebenen Makros. Mit <String> können die Werte lokaler Variablen für das Makro gesetzt werden.
MAC? (S. 240)	MAC? [<macroname>]	Listet alle Makros oder den Inhalt des angegebenen Makros auf.
MEX (S. 241)	MEX <CMD?> <OP> <Value>	Kann nur in Makros verwendet werden. Stoppt die Makroausführung in Abhängigkeit von einer Bedingung.
RMC? (S. 253)	RMC?	Listet die aktuell laufenden Makros auf.
STP (S. 267)	STP	Stoppt die Makroausführung.
VAR (S. 275)	VAR <Variable> <String>	Setzt eine Variable (S. 135) auf einen bestimmten Wert oder löscht sie.
VAR? (S. 277)	VAR? [{<Variable>}]	Gibt Variablenwerte zurück.
WAC (S. 281)	WAC <CMD?> <OP> <Value>	Kann nur in Makros verwendet werden. Wartet, bis eine Bedingung erfüllt ist.
#8 (S. 155)	-	Prüft, ob ein Makro auf dem Controller ausgeführt wird.
#24 (S. 156)	-	Stoppt die Makroausführung.

INFORMATION

Pro Befehlszeile sind maximal 256 Zeichen zulässig.

Parameter

Folgender Parameter steht für die Arbeit mit Makros zur Verfügung:

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

7.5.3 Mit Makros arbeiten

Die Arbeit mit Makros umfasst Folgendes:

- Aufzeichnen von Makros (S. 129)
- Starten der Makroausführung (S. 132)
- Stoppen der Makroausführung (S. 134)
- Einrichten eines Startup-Makros (S. 134)
- Löschen von Makros (S. 135)

INFORMATION

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

Makro aufzeichnen

INFORMATION

Der C-887 kann eine unbegrenzte Anzahl von Makros gleichzeitig speichern. Maximal 10 Verschachtelungsebenen sind in Makros möglich.

INFORMATION

Grundsätzlich können alle GCS-Befehle (S. 139) Bestandteil eines Makros werden. Ausnahmen:

- `RBT` für den Neustart des C-887
- `MAC BEG` und `MAC END` für die Makroaufzeichnung
- `MAC DEL` zum Löschen eines Makros
- `#27` zum Anhalten des C-887

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 weiterleiten.

INFORMATION

Um die Anwendung von Makros flexibler zu gestalten, können Sie in Makros lokale und globale Variablen verwenden. Weitere Informationen siehe "Variablen" (S. 135).

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. 135) in Makros, und geben Sie beim Starten der Makroausführung die entsprechenden Variablenwerte an.
- Wenden Sie sich an unseren Kundendienst (S. 367), wenn der C-887 unerwartetes Verhalten zeigt.

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: Wählen Sie 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. Wählen Sie 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-887 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-887 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:

– Wählen Sie das Symbol ***Read list of macros from controller***.

– Wählen Sie das zu prüfende Makro in der Liste auf der linken Seite, und wählen Sie 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 X 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 X 12.5
WAC ONT? X = 1
MAC END
MAC BEG macro2
MVR X -12.5
WAC ONT? X = 1
MAC END
MAC BEG macro3
MAC START macro1
MAC START macro2
MAC END
```

Makroausführung starten

INFORMATION

Während auf dem C-887 ein Makro ausgeführt wird, können über alle Kommunikationsschnittstellen Befehle an den C-887 gesendet werden. Befehle werden in der Reihenfolge ihres Eintreffens abgearbeitet. Die Zeitdauer für das Abarbeiten einzelner Befehle hängt vom jeweiligen Befehl ab. Die Ausführung eines zeitaufwändigen Befehls, der über eine Kommunikationsschnittstelle gesendet wurde, kann daher zu einer Pause in der Makroausführung führen. Makros sind deshalb z. B. nicht für das zeitkritische Abfahren festgelegter Dynamikprofile geeignet.

- Verwenden Sie zum Abfahren eines festgelegten zeitkritischen Dynamikprofils kein Makro, sondern den Funktionsgenerator (S. 102) oder aufeinander folgende MOV-Befehle, die in einem Zwischenspeicher abgelegt werden (S. 39).
- Vermeiden Sie die Verwendung einer Bedieneinheit (C-887.MC2 oder C-887.MC) während der Makroausführung.
- Wenn möglich, starten oder verwenden Sie während der Makroausführung keine PC-Software wie PIMikroMove® oder Treiber zur Verwendung mit NI LabVIEW-Software.
- Wenn möglich, vermeiden Sie während der Makroausführung das Senden von Befehlen.

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 das 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. 317).
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 X 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-887 erneut an, da sie nur in den flüchtigen Speicher des C-887 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 X $1
   WAC ONT? X = 1
   MAC END
   ```

   MOVWAI bewegt die Achse X 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 X 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 das PITerminal oder das Fenster ***Command entry*** von PIMikroMove® verwendet, um Befehle einzugeben. Details zur Arbeit mit der Registerkarte ***Controller macros*** in PIMikroMove® finden Sie imPIMikroMove® -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-887 ausgeführt.

INFORMATION

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

Im Folgenden wird das 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: Referenzierungsfahrt für Hexapod durch Startup-Makro auslösen

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 startet eine Referenzierungsfahrt. STARTCL wird als Startup-Makro festgelegt, so dass der Hexapod die Referenzierungsfahrt sofort nach dem Einschalten ausführt.

- Senden Sie:

```
MAC BEG startcl
DEL 1000
FRF X
MAC END
MAC DEF startcl
```

Makro löschen

INFORMATION

Ein laufendes Makro kann nicht gelöscht werden.

Im Folgenden wird das 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.

7.5.4 Variablen

Für eine flexiblere Programmierung unterstützt der C-887 die Verwendung von Variablen in Makros. Während globale Variablen für alle Makros verwendet werden können, gelten lokale Variablen immer nur für ein bestimmtes Makro. Die Anzahl der globalen und lokalen Variablen ist nicht beschränkt.

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

Die Arbeit mit Variablen umfasst Folgendes:

- Erzeugen von Variablen:
  - Lokale Variablen: siehe "Besonderheiten lokaler Variablen"
  - Globale Variablen: innerhalb eines Makros mit den Befehlen `ADD` (S. 161), `CPY` (S. 163) und `VAR` (S. 275), außerhalb eines Makros mit `VAR`
- Ändern von Variablenwerten: innerhalb eines Makros mit den Befehlen `ADD`, `CPY` und `VAR`
- Abfragen von Variablenwerten: mit `VAR?` innerhalb oder außerhalb eines Makros
- Löschen von Variablen: mit `VAR` innerhalb eines Makros; globale Variablen auch außerhalb eines Makros

Für Variablennamen gelten folgende Konventionen:

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

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

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

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

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

Besonderheiten lokaler Variablen

- Die Namen der in einem Makro verwendeten lokalen Variablen müssen eine fortlaufende Reihe bilden. Beispiel für erlaubte Benennung: 1, 2, 3, 4. Nicht erlaubt ist z. B. die Benennung mit 1, 2, 5, 6.
- Die Werte lokaler Variablen werden als Argumente <String> der Befehle `MAC START` oder `MAC NSTART` beim Start des Makros angegeben.

  Befehlsformate:

  `MAC START <macroname> [{<String>}]`

  `MAC NSTART <macroname> <uint> [{<String>}]`

<String> steht für den Wert einer im Makro enthaltenen lokalen Variablen. 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. Pro Befehlszeile sind maximal 256 Zeichen zulässig. <String> kann direkt oder über den Wert einer anderen Variable angegeben werden.

<uint> bestimmt, wie oft das Makro ausgeführt werden soll. Weitere Informationen finden Sie in der Beschreibung des Befehls `MAC` (S. 237).

- 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.
- Die Werte lokaler Variablen können mit `VAR?` abgefragt werden, solange das Makro ausgeführt wird.

8 GCS-Befehle

In diesem Kapitel

Schreibweise 139
GCS-Syntax für Syntaxversion 2.0 139
Befehlsübersicht 142
Befehlsbeschreibungen für GCS 2.0 152
Fehlercodes 299

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.
LF	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-887 unterstützten Kennungen sind im Abschnitt "Kommandierbare Elemente" (S. 27) beschrieben.

Beispiel 1:

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

Senden: `MOV`SP`1`SP`10.0`LF

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

Beispiel 2:

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

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

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

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

Beispiel 3:

Alle Parameter im flüchtigen Speicher sollen zurückgesetzt werden.

Senden: `RPA`LF

Beispiel 4:

Die Position aller Achsen soll abgefragt werden.

Senden: `POS?`LF

Die Antwort-Syntax lautet wie folgt:

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

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

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

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

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

Abfragebefehl:

CMD?SP<Arg3>SP<Arg1>SP<Arg2>LF

Antwort auf diesen Befehl:

<Arg3>"="<Wert3>SP LF

<Arg1>"="<Wert1>SP LF

<Arg2>"="<Wert2> LF

Beispiel 5:

Senden: `TSP?`SP`2`SP`1`LF

Empfangen: `2=-1158.4405`SP LF

`1=+0000.0000`LF

INFORMATION

Pro Befehlszeile sind maximal 256 Zeichen zulässig.

8.3 Befehlsübersicht

Befehl	Format	Beschreibung
#3 (S. 152)	#3	Get Real Position
#4 (S. 152)	#4	Request Status Register
#5 (S. 153)	#5	Request Motion Status
#6 (S. 154)	#6	Query for Position Change
#7 (S. 154)	#7	Request Controller Ready Status
#8 (S. 155)	#8	Query if Macro is Running
#9 (S. 155)	#9	Get Wave Generator Status
#11 (S. 155)	#11	Get Memory Space for Trajectory Points
#24 (S. 156)	#24	Stop All Axes
#27 (S. 156)	#27	System Abort
*IDN? (S. 157)	*IDN?	Get Device Identification
AAP (S. 158)	AAP <AxisID> <Distance> <AxisID> <Distance> ["SA" <StepSize>] ["N" <NumberOfRepetitions>] ["A" <AnalogInputID>]	Automated Alignment Part
ADD (S. 161)	ADD <Variable> <FLOAT1> <FLOAT2>	Add and Save to Variable
CCL (S. 162)	CCL <Level> [<PSWD>]	Set Command Level
CCL? (S. 162)	CCL?	Get Command Level
CPY (S. 163)	CPY <Variable> <CMD?>	Copy into Variable
CST (S. 163)	CST {<AxisID> <StageName>}	Set Assignment of Stages to Axes
CST? (S. 164)	CST? [{<AxisID>}]	Get Assignment Of Stages To Axes
CSV? (S. 164)	CSV?	Get Current Syntax Version
DEL (S. 165)	DEL <uint>	Delay the Command Interpreter
DIA? (S. 165)	DIA? [{<MeasureID>}]	Get Diagnosis Information
DIO (S. 166)	DIO {<DIOID> <OutputOn>}	Set Digital Output Lines

Befehl	Format	Beschreibung
DIO? (S. 167)	DIO? [{<DIOID>}]	Get Digital Input Lines
DPA (S. 167)	DPA <Pswd> [{<ItemID> <PamID>}]	Reset Volatile Memory Settings To Default
DRC (S. 168)	DRC {<RecTableID> <Source> <RecOption>}	Set Data Recorder Configuration
DRC? (S. 170)	DRC? [{<RecTableID>}]	Get Data Recorder Configuration
DRL? (S. 171)	DRL? [{<RecTableID>}]	Get Number Of Recorded Points
DRR? (S. 172)	DRR? [<StartPoint> <NumberOfPoints> [{<RecTableID>}]]	Get Recorded Data Values
DRT (S. 173)	DRT {<RecTableID> <TriggerSource> <Value>}	Set Data Recorder Trigger Source
DRT? (S. 175)	DRT? [{<RecTableID>}]	Get Data Recorder Trigger Source
ECO? (S. 175)	ECO? <String>	Echo a String
ERR? (S. 176)	ERR?	Get Error Number
FDG	FDG <routine name> <scan axis> <step axis> [ML <stop level>] [A <alignment signal input channel>] [MIA <min radius>] [MAA <max radius>] [F <frequency>] [SP <speed factor>] [V <max velocity>] [MDC <max direction changes>] [SPO <speed offset>]	Fast Alignment: Defines a fast alignment gradient search routine. Details siehe E712T0016 "Schnelle, mehrkanalige Ausrichtung in der Photonik".
FDR	FDR <routine name> <scan axis> <scan axis range> <step axis> <step axis range> [L <threshold level>] [A <alignment signal input channel>] [F <frequency>] [V <velocity>] [MP1 <scan axis middle position>] [MP2 <step axis middle position>] [TT <target type>] [CM <estimation method>] [MIIL <minimum level of fast alignment input>] [MAIL <maximum level of fast alignment input>] [ST <stop position option>]	Fast Alignment: Defines a fast alignment area scan routine. Details siehe E712T0016 "Schnelle, mehrkanalige Ausrichtung in der Photonik".

Befehl	Format	Beschreibung
FGC	FGC {<routine name> <scan axis center position> <step axis center position>}	Fast Alignment: Changes the center position of a gradient search routine that is currently running. Details siehe E712T0016 "Schnelle, mehrkanalige Ausrichtung in der Photonik".
FGC?	FGC? [{<routine name>}]	Fast Alignment: Gets the current center position of a gradient search routine. Details siehe E712T0016 "Schnelle, mehrkanalige Ausrichtung in der Photonik".
FIO (S. 177)	FIO <AxisID> <Distance> <AxisID> <Distance> ["S" <LinearSpiralStepSize>] ["AR" <AngularScanSize>] ["L" <Threshold>] ["A" <AnalogInputID>]	Fast Input-Output Alignment Procedure
FLM (S. 180)	FLM <AxisID> <Distance> ["L" <Threshold>] ["A" <AnalogInputID>] ["D" <ScanDirection>]	Fast Line Scan to Maximum
FLS (S. 182)	FLS <AxisID> <Distance> ["L" <Threshold>] ["A" <AnalogInputID>] ["D" <ScanDirection>]	Fast Line Scan
FRF (S. 185)	FRF [{<AxisID>}]	Fast Reference Move To Reference Switch
FRF? (S. 186)	FRF? [{<AxisID>}]	Get Referencing Result
FRH?	FRH?	Fast Alignment: Lists descriptions and physical units for the routine results that can be queried with the FRR? command Details siehe E712T0016 "Schnelle, mehrkanalige Ausrichtung in der Photonik".
FRP	FRP {<routine name> <routine action>}	Fast Alignment: Stops a fast alignment routine. Details siehe E712T0016 "Schnelle, mehrkanalige Ausrichtung in der Photonik".

Befehl	Format	Beschreibung
FRP?	FRP? [{<routine name>}]	Fast Alignment: Gets the current state of a fast alignment routine Details siehe E712T0016 "Schnelle, mehrkanalige Ausrichtung in der Photonik".
FRR?	FRR? [<routine name> [<result ID>]]	Fast Alignment: Gets the results of a fast alignment routine Details siehe E712T0016 "Schnelle, mehrkanalige Ausrichtung in der Photonik".
FRS	FRS {<routine name>}	Fast Alignment: Starts a fast alignment routine Details siehe E712T0016 "Schnelle, mehrkanalige Ausrichtung in der Photonik".
FSA (S. 187)	FSA <Axis1ID> <Distance1> <Axis2ID> <Distance2> ["L" <Threshold>] ["S" <ScanLineDistance>] ["SA" <StepSize>] ["A" <AnalogInputID>]	Fast Scan with Automated Alignment
FSC (S. 190)	FSC <Axis1ID> <Distance1> <Axis2ID> <Distance2> ["L" <Threshold>] ["S" <ScanLineDistance>] ["A" <AnalogInputID>]	Fast Scan with Abort
FSM (S. 195)	FSM <Axis1ID> <Distance1> <Axis2ID> <Distance2> ["L" <Threshold>] ["S" <ScanLineDistance>] ["A" <AnalogInputID>]	Fast Scan to Maximum
FSS? (S. 198)	FSS?	Get Status of Fast Scan Routines
GWD? (S. 199)	GWD? [<StartPoint> <NumberOfPoints> [{<WaveTableID>}]]	Get Wave Table Data
HDI? (S. 200)	HDI?	Get Help For Interpretation Of DIA? Response
HDR? (S. 200)	HDR?	Get All Data Recorder Options
HIB? (S. 202)	HIB? [{<HIDeviceID> <HIDeviceButton>}]	Get State Of HID Button
HLP? (S. 203)	HLP?	Get List of Available Commands

Befehl	Format	Beschreibung
HLT (S. 204)	HLT [{<AxisID>}]	Halt Motion Smoothly
HPA? (S. 204)	HPA?	Get List Of Available Parameters
IFC? (S. 205)	IFC? [{<InterfacePam>}]	Get Current Interface Parameters
IFS (S. 207)	IFS <Pswd> {<InterfacePam> <PamValue>}	Set Interface Parameters as Default Values
IFS? (S. 208)	IFS? [{<InterfacePam>}]	Get Interface Parameters as Default Values
IMP (S. 209)	IMP <AxisID> <Amplitude>	Start Impulse and Response Measurement
JRC (S. 210)	JRC <Jump> <CMD?> <OP> <Value>	Jump Relatively Depending on Condition
KCP (S. 210)	KCP <CSNameSource> <CSNameCopy>	Copy Coordinate System
KEN (S. 211)	KEN <CSName>	Activate Coordinate System
KEN? (S. 213)	KEN? [{<CSName>}]	Get Active Coordinate Systems
KET? (S. 214)	KET? [{<CSType>}]	Get Active Coordinate System Types
KLC? (S. 215)	KLC? [<CSName1>[<CSName2>[<Item1>[<Item2>]]]]	Get Properties Of Work-And-Tool Combinations
KLD (S. 218)	KLD <CSName> [{<AxisID> <Offset>}]	Define Leveling Coordinate System By Specifying Values Vor dem Definieren eines Korrektur-Koordinatensystems ist der Wechsel auf Befehlsebene 1 erforderlich (siehe CCL).
KLF (S. 220)	KLF <CSName>	Define Leveling Coordinate System At Current Position Vor dem Definieren eines Korrektur-Koordinatensystems ist der Wechsel auf Befehlsebene 1 erforderlich (siehe CCL).
KLN (S. 221)	KLN <ChildCS> <ParentCS>	Link Coordinate Systems

Befehl	Format	Beschreibung
KLN? (S. 223)	KLN? [{<CSName>}]	Get Coordinate System Chains
KLS? (S. 223)	KLS? [<CSName>[<Item1>[<Item2>]]]	Get Coordinate System Properties
KLT? (S. 226)	KLT? [<StartCS> [<EndCS>]]	Get Offsets Resulting From A Chain
KRM (S. 227)	KRM <CSName>	Remove Coordinate System
KSB (S. 227)	KSB <CSName> [{<AxisID> <Angle>}]	Define Orientational Coordinate System Vor dem Definieren eines Achsorientierungs-Koordinatensystems ist der Wechsel auf Befehlsebene 1 erforderlich (siehe CCL).
KSD (S. 229)	KSD <CSName> [{<AxisID> <Offset>}]	Define Operating Coordinate System By Specifying Values
KSF (S. 231)	KSF <CSName>	Define Operating Coordinate System At Current Position
KST (S. 232)	KST <CSName> [{<AxisID> <Offset>}]	Define “Tool” Operating Coordinate System
KSW (S. 235)	KSW <CSName> [{<AxisID> <Offset>}]	Define “Work” Operating Coordinate System
LIM? (S. 237)	LIM? [{<AxisID>}]	Indicate Limit Switches
MAC (S. 237)	MAC <keyword> {<parameter>}	Call Macro Function
MAC? (S. 240)	MAC? [<macroname>]	List Macros
MAN? (S. 240)	MAN? {<CMD>}	Get Help String For Command
MEX (S. 241)	MEX <CMD?> <OP> <value>	Stop Macro Execution due to Condition
MOV (S. 242)	MOV {<AxisID> <Position>}	Set Target Position
MOV? (S. 243)	MOV? [{<AxisID>}]	Get Target Position
MRT (S. 244)	MRT {<AxisID> <Distance>}	Set Target Relative In Tool Coordinate System

Befehl	Format	Beschreibung
MRW (S. 245)	MRW {<AxisID> <Distance>}	Set Target Relative In Work Coordinate System
MVR (S. 246)	MVR {<AxisID> <Distance>}	Set Target Relative To Current Position
NAV (S. 247)	NAV {<AnalogInputID> <NumberOfReadings>}	Set Number of Readout Values to be Averaged
NAV? (S. 248)	NAV? [{<AnalogInputID>}]	Get Number of Readout Values to be Averaged
NLM (S. 248)	NLM {<AxisID> <LowLimit>}	Set Low Position Soft Limit
NLM? (S. 249)	NLM? [{<AxisID>}]	Get Low Position Soft Limit
ONT? (S. 250)	ONT? [{<AxisID>}]	Get On-Target State
PLM (S. 251)	PLM {<AxisID> <HighLimit>}	Set High Position Soft Limit
PLM? (S. 252)	PLM? [{<AxisID>}]	Get High Position Soft Limit
POS? (S. 252)	POS? [{<AxisID>}]	Get Real Position
PUN? (S. 253)	PUN? [{<AxisID>}]	Get Axis Unit
RBT (S. 253)	RBT	Reboot System
RMC? (S. 253)	RMC?	List Running Macros
RON? (S. 254)	RON? [{<AxisID>}]	Get Reference Mode
RTR (S. 254)	RTR <RecordTableRate>	Set Record Table Rate
RTR? (S. 255)	RTR?	Get Record Table Rate
SAI? (S. 256)	SAI? [ALL]	Get List Of Current Axis Identifiers
SCT (S. 256)	SCT "T" <CycleTime>	Set Cycle Time
SCT? (S. 257)	SCT? [<T>]	Get Cycle Time

Befehl	Format	Beschreibung
SGA (S. 258)	SGA {<AnalogInputID> <Gain>}	Set Gain
SGA? (S. 258)	SGA? [{<AnalogInputID>}]	Get Gain
SIC	SIC <FA input channel ID> <calculation type> [{<calculation parameter>}]	Fast Alignment: Defines calculation settings for an analog input channel Details siehe E712T0016 "Schnelle, mehrkanalige Ausrichtung in der Photonik".
SIC?	SIC? [{<FA input channel ID>}]	Fast Alignment: Gets the calculation settings for an analog input channel Details siehe E712T0016 "Schnelle, mehrkanalige Ausrichtung in der Photonik".
SPA (S. 258)	SPA {<ItemID> <PamID> <PamValue>}	Set Volatile Memory Parameters
SPA? (S. 259)	SPA? [{<ItemID> <PamID>}]	Get Volatile Memory Parameters
SPI (S. 260)	SPI {<PPCoordinate> <Position>}	Set Pivot Point
SPI? (S. 262)	SPI? [{<PPCoordinate>}]	Get Pivot Point
SRG? (S. 262)	SRG? {<ItemID> <RegisterID>}	Query Status Register Value
SSL (S. 263)	SSL {<AxisID> <SoftLimitsOn>}	Set Soft Limit
SSL? (S. 264)	SSL? [{<AxisID>}]	Get Soft Limit Status
SSN? (S. 264)	SSN?	Get Device Serial Number
SST (S. 264)	SST {<AxisID> <StepSize>}	Set Step Size
SST? (S. 265)	SST? [{<AxisID>}]	Get Step Size
STA? (S. 266)	STA?	Query Status Register Value
STE (S. 266)	STE <AxisID> <Amplitude>	Start Step And Response Measurement

Befehl	Format	Beschreibung
STP (S. 267)	STP	Stop All Axes
SVO (S. 268)	SVO {<AxisID> <ServoState>}	Set Servo Mode
SVO? (S. 269)	SVO? [{<AxisID>}]	Get Servo Mode
TAC? (S. 269)	TAC?	Tell Analog Channels
TAD? (S. 270)	TAD? [{<InputSignalID>}]	Get ADC Value Of Input Signal
TAV? (S. 270)	TAV? [{<AnalogInputID>}]	Get Analog Input Voltage
TCI?	TCI? [{<FA input channel ID>}]	Fast Alignment: Gets calculated value of an analog input channel Details siehe E712T0016 "Schnelle, mehrkanalige Ausrichtung in der Photonik".
TMN? (S. 271)	TMN? [{<AxisID>}]	Get Minimum Commandable Position
TMX? (S. 272)	TMX? [{<AxisID>}]	Get Maximum Commandable Position
TNR? (S. 273)	TNR?	Get Number Of Record Tables
TRA? (S. 273)	TRA? {<AxisID> <Component>}	Get Maximum Commandable Position For Direction Vector
TRS? (S. 275)	TRS? [{<AxisID>}]	Indicate Reference Switch
TWG? (S. 275)	TWG?	Get Number of Wave Generators
VAR (S. 275)	VAR <Variable> <String>	Set Variable Value
VAR? (S. 277)	VAR? [{<Variable>}]	Get Variable Value
VEL (S. 277)	VEL {<AxisID> <Velocity>}	Set Closed-Loop Velocity
VEL? (S. 278)	VEL? [{<AxisID>}]	Get Closed-Loop Velocity

Befehl	Format	Beschreibung
VER? (S. 278)	VER?	Get Version
VLS (S. 279)	VLS <SystemVelocity>	Set System Velocity
VLS? (S. 280)	VLS?	Get System Velocity
VMO? (S. 280)	VMO? {<AxisID> <Position>}	Virtual Move
VST? (S. 281)	VST?	Get Connectable Stages
WAC (S. 281)	WAC <CMD?> <OP> <value>	Wait for Condition
WAV (S. 282)	WAV <WaveTableID> <AppendWave> <WaveType> <WaveTypeParameters>	Set Waveform Definition
WAV? (S. 287)	WAV? [{<WaveTableID> <WaveParameterID>}]	Get Waveform Definition
WCL (S. 288)	WCL {<WaveTableID>}	Clear Wave Table Data
WGC (S. 288)	WGC {<WaveGenID> <Cycles>}	Set Number Of Wave Generator Cycles
WGC? (S. 289)	WGC? [{<WaveGenID>}]	Get Number Of Wave Generator Cycles
WGO (S. 289)	WGO {<WaveGenID> <StartMode>}	Set Wave Generator Start/Stop Mode
WGO? (S. 291)	WGO? [{<WaveGenID>}]	Get Wave Generator Start/Stop Mode
WGR (S. 291)	WGR	Starts Recording In Sync With Wave Generator
WGS? (S. 292)	WGS? [<WaveGenID> [<ItemID>]]	Get Status Information of Wave Generator
WMS? (S. 292)	WMS? [{<WaveTableID>}]	Get Maximum Number of Values for the Waveform
WPA (S. 293)	WPA <Pswd> [{<ItemID> <PamID>}]	Save Parameters To Nonvolatile Memory
WSL (S. 295)	WSL {<WaveGenID> <WaveTableID>}	Set Connection Of Wave Table To Wave Generator
WSL? (S. 295)	WSL? [{<WaveGenID>}]	Get Connection Of Wave Table To Wave Generator

Befehl	Format	Beschreibung
WTR (S. 296)	WTR {<WaveGenID> <WaveTableRate> <InterpolationType>}	Set Wave Generator Table Rate
WTR? (S. 298)	WTR? [{<WaveGenID>}]	Get Wave Generator Table Rate

8.4 Befehlsbeschreibungen für GCS 2.0

#3 (Get Real Position)

Beschreibung: Fragt die aktuelle Achsenposition ab.

Format: #3 (einzelnes ASCII Zeichen Nummer 3)

Argumente: Keine

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

wobei

<float> die aktuelle Achsenposition in physikalischen Einheiten ist.

Hinweise: Dieser Befehl ist funktionsgleich mit POS? (S. 252), aber es muss nur ein Zeichen über die Schnittstelle gesendet werden.

Die aktuelle Position der Achsen X, Y, Z, U, V und W wird aus den gemessenen Positionen der einzelnen Beine berechnet.

Zwischen Einschalten des Controllers und der Referenzierung des Hexapods mit FRF (S. 185) ist die aktuelle Position des Hexapods und der Achsen A und B unbekannt. Dennoch liefert die Antwort auf #3 für alle Achsen den Positionswert 0.

Die physikalische Einheit, in der die Achsenposition angegeben wird, kann mit PUN? (S. 253) abgefragt werden.

#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: Die Antwort ist die Summe der Codes in Hexadezimalformat. Für die Beschreibung der Bits und ein Beispiel siehe "Systemstatusregister" (S. 386).

Dieser Befehl ist funktionsgleich mit STA? (S. 266), aber es muss nur ein Zeichen über die Schnittstelle gesendet werden.

Zusätzliche Statusinformationen, die **nicht** mit #4 und STA? abgefragt werden können: Für jedes der Beine 1 bis 6 des Hexapods und für die Achsen A und B hat der C-887 jeweils ein Statusregister (S. 386). Sie können die Bits dieser Register mit dem Befehl SRG? (S. 262) abfragen und mit dem Datenrekorder des C-887 (S. 99) aufzeichnen, Aufzeichnungsoption 80 (Status register of axis).

#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.

Hinweis: Der Bewegungsstatus *aller* Achsen des Hexapods (X bis W) lautet "in Bewegung", solange mindestens *ein* Hexapod-Bein in Bewegung ist (Beinstatus gemäß den Statusregisterbits, Details siehe "Bewegungsstatus, Einschwingfenster, Einschwingzeit" (S. 46)).

Die Achsen 1 bis 8 entsprechen den Achsen X, Y, Z, U, V, W, A und B in dieser Reihenfolge.

Ausnahme: Wenn einer Achse der Positionierertyp "NOSTAGE" zugewiesen ist (möglich für die Achsen A und B; Abfrage mit dem Befehl CST? (S. 164)), ist diese Achse nicht in der bit-codierten Antwort enthalten. Sie wird in diesem Fall bei der Zählung der Achsen übersprungen.

#6 (Query for Position Change)

Beschreibung: Fragt, ob sich die Achsenpositionen seit der zuletzt gesendeten Positionsabfrage durch die Kommandierung einer neuen Zielposition geändert haben (Kommandierungsmöglichkeiten siehe "Unterstützte Bewegungsarten" (S. 35)).

Format: #6 (einzelnes ASCII Zeichen Nummer 6)

Argumente: Keine

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

1 = Position der ersten Achse hat sich geändert
2 = Position der zweiten Achse hat sich geändert
4 = Position der dritten Achse hat sich geändert
...

Beispiele: 0 gibt an, dass sich keine Achsenposition geändert hat.

3 gibt an, dass sich die Positionen der ersten und der zweiten Achse geändert haben.

49 gibt an, dass sich die Positionen der Achsen X, U und A geändert haben.

Hinweise: Wenn einer Achse der Positionierertyp "NOSTAGE" zugewiesen ist (möglich für die Achsen A und B; Abfrage mit dem Befehl CST? (S. 164)), ist diese Achse nicht in der bit-codierten Antwort enthalten. Sie wird in diesem Fall bei der Zählung der Achsen übersprungen.

Als Positionsänderung wird gewertet, wenn seit dem zuletzt gesendeten POS? (S. 252) oder #3 (S. 152) eine neue Zielposition per Befehl - auch innerhalb eines Makros -, durch den Funktionsgenerator oder über eine angeschlossene Bedieneinheit (S. 24) (C-887.MC2 oder C-887.MC) vorgegeben wurde.

#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. 153) 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

#9 (Get Wave Generator Status)

Beschreibung: Fragt den Status der Funktionsgeneratoren ab.

Format: #9

Argumente: Keine

Antwort: Die Antwort <uint> ist bit-codiert und wird als hexadezimale Summe der folgenden Codes ausgegeben:
1 = Funktionsgenerator 1 ist aktiv,
2 = Funktionsgenerator 2 ist aktiv,
4 = Funktionsgenerator 3 ist aktiv usw.
"Aktiv" = Funktionsgeneratorausgabe läuft

Beispiele: 0 gibt an, dass kein Funktionsgenerator aktiv ist
5 gibt an, dass die Funktionsgeneratoren 1 und 3 aktiv sind

#11 (Get Memory Space for Trajectory Points)

Beschreibung: Fragt den freien Speicherplatz für die Punkte des Dynamikprofils ab.

Format: #11 (einzelnes ASCII Zeichen Nummer 11)

Argumente: Keine

Antwort: <uint> ist der freie Speicherplatz, Angabe als Anzahl der Dynamikprofilpunkte.

Hinweise: #11 fragt den freien Speicherplatz eines Zwischenspeichers ab, der Dynamikprofilpunkte für den Hexapod enthält. Ein Dynamikprofilpunkt entspricht einem Satz von Zielpositionen für die Achsen des Hexapods (X, Y, Z, U, V, W). Der Inhalt des Zwischenspeichers wird nur dann zur Festlegung des Dynamikprofils verwendet, wenn die Parameter 0x19001900 und 0x19001901 jeweils den Wert 1 haben.

Weitere Informationen siehe "Zyklische Übertragung von Zielpositionen" (S. 39).

#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. 267), aber es wird nur ein Zeichen über die Schnittstelle gesendet.

Format: #24

Argumente: Keine

Antwort: Keine

Hinweise: #24 stoppt alle Achsenbewegungen, die durch Bewegungsbefehle, Fast-Alignment-Routinen, Scanprozeduren oder Funktionsgeneratorausgabe verursacht werden, und die Referenzierungsfahrt.

#24 stoppt Makros.

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

#27 (System Abort)

Beschreibung: Hält den Controller an.

Format: #27 (einzelnes ASCII Zeichen Nummer 27)

Argumente: Keine

Antwort: Keine

Hinweise: #27 löst Folgendes aus:

- Motoren der Beine werden abgeschaltet
- Servomodus wird ausgeschaltet
- Befehle werden nicht mehr abgearbeitet
- Parameter im flüchtigen Speicher werden auf Standardwerte gesetzt
- Netzteil des Controllers bleibt eingeschaltet

Für einen Neustart muss der Controller aus- und wieder eingeschaltet werden.

***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

Hinweis: Beispiel-Antwort für den C-887:

```
(c)2011-2017 Physik Instrumente (PI) GmbH & Co. KG,C-887,117009333,2.3.1.1
```

Bedeutung der Antwort:

- `C-887` Gerätebezeichnung
- `117009333` Seriennummer des C-887
- `2.3.1.1` Version der Firmwarekomponente FW

AAP (Automated Alignment Part)

Beschreibung: Startet eine Scanprozedur zum verfeinerten Ermitteln des Intensitätsmaximums eines analogen Eingangssignals.

Die mit AAP gestartete Scanprozedur entspricht dem „Feinanteil" der Scanprozedur, die mit dem Befehl FSA (S. 187) gestartet wird.

Scannt eine vorgegebene Fläche („Scanbereich") mit Hilfe einer Gradientenbildung über die Intensität, bis ein lokales Intensitätsmaximum des analogen Eingangssignals gefunden wird.

Die Ebene, in der der Scanbereich liegt, kann durch eines der folgenden Achsenpaare definiert sein:

X Y
Y Z
X Z
U V
U W
V W

Die Scanprozedur startet an der aktuellen Position zum Zeitpunkt des Befehlsempfangs („Ausgangsposition"). Der Scanbereich ist um die Ausgangsposition herum zentriert.

Die Schrittweite der Bewegung wird automatisch angepasst, um das lokale Maximum möglichst genau zu erfassen.

AAP (und damit die Bewegung der Plattform) ist erfolgreich beendet, wenn die mit <NumberOfRepetitions> festgelegte Anzahl von Überprüfungen bestätigt hat, dass an der aktuellen Position das lokale Maximum vorliegt. Durch mehrmalige Überprüfungen kann der Einfluß eines überlagerten Rauschsignals vermindert werden.

Wenn <NumberOfRepetitions> auf null gesetzt ist, wird kontinuierlich nach einem Gradienten der Intensität gesucht. Mit dieser Einstellung kann ein wanderndes lokales Maximum kontinuierlich verfolgt werden. Die Scanprozedur endet nur, wenn sie mit #24 (S. 156), STP (S. 267) oder HLT (S. 204) abgebrochen wird, und gilt dann als erfolglos beendet.

AAP wird in folgenden Fällen erfolglos beendet:

- Kein Gradient über die Intensität gefunden: die Bewegungsplattform kehrt zurück zur Ausgangsposition
- Scanbereich würde überschritten: die Bewegungsplattform kehrt zurück zur Ausgangsposition
- #24 (S. 156), STP (S. 267) oder HLT (S. 204) wurde gesendet: die Bewegungsplattform bleibt an der aktuellen Position stehen

Mit FSS? (S. 198) kann geprüft werden, ob eine Scanprozedur erfolgreich beendet wurde.

Um zu prüfen, ob eine Scanprozedur noch andauert, kann mit #5 (S. 153) der Bewegungsstatus der Achsen abgefragt werden.

Format:

AAP <AxisID> <Distance> <AxisID> <Distance>
["SA" <StepSize>] ["N" <NumberOfRepetitions>]
["A" <AnalogInputID>]

Argumente:

<AxisID>: ist eine Achse des Controllers, Format: String
Zulässig sind die Achsen X, Y, Z, U, V, W.

<Distance>: abzuscannende Strecke entlang der Achse, Format: double

"SA": erforderliches Schlüsselwort zur Eingabe von <StepSize>
<StepSize>: Startwert für die Schrittweite, Format: double

"N": erforderliches Schlüsselwort zur Eingabe von <NumberOfRepetitions>
<NumberOfRepetitions>: Anzahl der erfolgreichen Überprüfungen des lokalen Maximums an der aktuellen Position, die für erfolgreiches Beenden von AAP erforderlich ist. Format: Integer

"A": erforderliches Schlüsselwort zur Eingabe von <AnalogInputID>
<AnalogInputID>: ist die Kennung des analogen Eingangssignals, dessen Intensitätsmaximum gesucht wird, Format: Integer

Antwort:

Keine

Hinweise:

Die physikalische Einheit, in der <Distance> und <Stepsize> anzugeben sind, kann mit PUN? abgefragt werden.

Die Werte für <Distance> müssen für beide Achsen identisch sein.

Folgende Standardwerte werden verwendet, wenn die entsprechenden Argumente weggelassen werden:
<StepSize>: 0,001
<NumberOfRepetitions>: 3
<AnalogInputID>: 1
Diese Standardwerte wurden ausschließlich für die Hexapoden H-810, H-811 und H-206 geprüft. Für andere Hexapoden müssen die idealen Parameter experimentell ermittelt werden.

Wenn mehrere lokale Maxima im Scanbereich vorhanden sind, kann vor dem Aufruf von AAP FSM (S. 195) verwendet werden, um das globale Intensitätsmaximum zu finden. Dazu wird empfohlen, zuerst eine FSM-Scanprozedur mit großer Schrittweite über den gesamten Scanbereich durchzuführen, anschließend mit kleinerer Schrittweite einen verringerten Bereich mittels FSM zu scannen und schließlich die Ausrichtung der Bewegungsplattform auf das gefundene globale Intensitätsmaximum mit AAP zu optimieren.

Je größer das Rauschen des abzuscannenden Signals ist, desto häufiger sollte geprüft werden, ob ein lokales Maximum erreicht wurde, d. h., für <NumberOfRepetitions> sollte ein möglichst großer Wert gewählt werden.

Je größer das Rauschen des abzuscannenden Signals ist, desto höher sollte mit NAV (S. 247) die Anzahl der Auslesewerte des Analogsignals gesetzt werden, über die der Mittelwert des Analogsignals gebildet wird.
Dabei ist zu beachten, dass diese Einstellung den Zeitbedarf für die Scanprozedur erhöht.

Wenn der Parameter ***Trajectory Source*** (ID 0x19001900) auf 1 gesetzt ist, muss das Dynamikprofil durch aufeinander folgende MOV-Befehle vorgegeben werden, und AAP ist nicht zulässig.
Weitere Informationen siehe "Bewegungen des Hexapods" (S. 34).

Beispiel:

Senden: AAP Y 0.1 Z 0.1 SA 0.001 N 3 A 2

Startet eine Scanprozedur in der YZ-Ebene:

- Seitenlänge der quadratischen Fläche: 0,1 mm
- Startwert der Schrittweite: 1 µm
- Anzahl der erfolgreichen Überprüfungen des lokalen Maximums, nach denen AAP erfolgreich beendet ist: 3
- Kennung des analogen Eingangskanals, dessen Intensitätsmaximum gesucht wird: 2

ADD (Add And Save To Variable)

Beschreibung:	Addiert zwei Werte und speichert das Ergebnis als Variable (S. 135). 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. 367).
Antwort:	Keine
Fehlersuche:	Ungültiges Kennwort
Hinweise:	HLP? (S. 203) führt alle in der aktuellen Befehlsebene verfügbaren Befehle auf. HPA? (S. 204) listet die Parameter einschließlich der Information darüber, welche Befehlsebene Schreibzugriff auf sie erlaubt. Weitere Informationen zur Parameterverwendung siehe "Anpassen von Einstellungen" (S. 317). Nach dem Anschalten 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 Standardeinstellung, alle Befehle, die dem „normalen“ Benutzer zur Verfügung gestellt werden, sowie der Lesezugriff auf alle Parameter sind zugänglich

<Level> = 1 lässt zusätzliche Befehle und Schreibzugriff für Parameter der Ebene 1 zu (Befehle und Parameter von Ebene 0 sind inbegriffen)

CPY (Copy Into Variable)

Beschreibung: Kopiert eine Antwort auf einen Befehl in eine Variable (S. 135).

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

Hinweis: CPY kann nur in Makros verwendet werden.

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

Senden: `CPY TARGET VAR? SOURCE`

CST (Set Assignment of Stages to Axes)

Beschreibung: Weist einer Achse einen Positionierertyp zu.

Format: CST {<AxisID> <StageName>}

Argumente: <AxisID> ist eine Achse des Controllers
<StageName> ist der Name des Positionierertyps

Hinweise: Die Zuweisung eines Positionierertyps mit CST ist nur für die Achsen A und B zulässig.

CST schaltet für die Achsen A und B auch den Servomodus ein.

CST lädt die Betriebsparameter des zugewiesenen Positionierertyps aus der Positioniererdatenbank PIStages2.dat auf dem Controller in den Arbeitsspeicher. Die zulässigen Positionierertypen können mit dem Befehl VST? (S. 281) aufgelistet werden. Weitere Informationen finden Sie in "Betriebsparameter der Achsen A und B" (S. 32).

Die Zuweisung der Positionierertypen kann mit dem Befehl WPA A12 (S. 293) im permanenten Speicher des C-887 gespeichert werden.

Die aktuelle Zuweisung der Positionierertypen kann mit dem Befehl CST? (S. 164) abgefragt werden.

Wenn der Positionierertyp mit CST auf NOSTAGE gesetzt wird, 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.

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.

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

Hinweis: Als Antwort ist nur 2.0 (für GCS 2.0) möglich.

DEL (Delay the Command Interpreter)

Beschreibung: Verzögert um <uint> Millisekunden.

Format: DEL <uint>

Argumente: <uint> ist der Verzögerungswert in Millisekunden.

Antwort: Keine

Hinweis: DEL sollte nur in Makros verwendet werden.

DIA? (Get Diagnosis Information)

Beschreibung: Fragt den aktuellen Wert der angegebenen Messgröße ab.

Werden alle Argumente weggelassen, werden die aktuellen Werte aller Messgrößen abgefragt.

Format: DIA? [{<MeasureID>}]

Argumente: <MeasureID> ist die Kennung einer Messgröße; nähere Angaben siehe unten.

Antwort: {<MeasureID>"="<MeasuredValue> LF}

wobei

<MeasuredValue> den aktuellen Wert der Messgröße angibt; nähere Angaben siehe unten.

Hinweise: Verwenden Sie die Antwort auf den Befehl HDI? (S. 200), um Beschreibungen und physikalische Einheiten der unterstützten Messgrößen zu erhalten.

C-887 unterstützt die folgenden Messgrößen:

<MeasureID>	<Description> (mit HDI? abfragen)
1	Hexapod Powered: Power-Good-Signal (Pin 59 der Buchse **Hexapod** (S. 381)): 0 = Power-Good-Signal außerhalb des erforderlichen Bereichs oder nicht vorhanden. 1 = Power-Good-Signal im erforderlichen Bereich Anmerkung: Der Wert des Parameters ***Check PowerGood Signal*** (ID 0x19004000) ist für die Antwort auf DIA? **nicht** relevant.
2	Controller E-Stop Activated: Aktueller Status 0 = OFF, d. h. 24-V- Ausgang deaktiviert, kein Power-OK-Signal 1 = ON, d. h. 24-V-Ausgang aktiviert, Power-OK-Signal ist vorhanden
3	Temperature of Controller: Aktuelle Temperatur der CPU, in °C.
4	Index of Faulty Point in Waveform: Index des Kurvenformpunktes, an dem der letzte Fehler während der Funktionsgeneratorausgabe aufgetreten ist

DIO (Set Digital Output Lines)

Beschreibung: Schaltet die angegebene(n) digitale(n) Ausgangsleitung(en) in den angegebenen Status.

Verwenden Sie TIO?, um die Anzahl installierter digitaler I/O-Leitungen abzufragen.

Format: DIO {<DIOID> <OutputOn>}

Argumente: <DIOID> ist eine digitale Ausgangsleitung des Controllers; weitere Angaben siehe unten.

<OutputOn> ist der Status der digitalen Ausgangsleitung; nähere Angaben siehe unten.

Antwort: Keine

Hinweise: Mit dem Befehl DIO können die digitalen Ausgangsleitungen 1 bis 4, die sich auf der Buchse **I/O** (S. 381) befinden, aktiviert/deaktiviert werden.

Die für die Leitungen zu verwendenden Kennungen <DIOID> sind 1 bis 4.

Wenn <OutputOn>=1 wird die Leitung auf HIGH/ON gesetzt, wenn <OutputOn>=0 wird sie auf LOW/OFF gesetzt.

DIO? (Get Digital Input Lines)

Beschreibung: Fragt den Status der angegebenen digitalen Eingangsleitungen ab.

Verwenden Sie TIO?, um die Anzahl verfügbarer digitaler I/O-Leitungen abzufragen.

Format: DIO? [{<DIOID>}]

Argumente: <DIOID> ist die Kennung der digitalen Eingangsleitung; nähere Angaben siehe unten.

Antwort: {<DIOID>"="<InputOn> LF}

wobei

<InputOn> den Status der digitalen Eingangsleitung angibt; nähere Angaben siehe unten.

Hinweise: Mit dem Befehl DIO? können die digitalen Eingangsleitungen 1 bis 4, die sich auf der Buchse **I/O** (S. 381) befinden, direkt gelesen werden.

Die für die Leitungen zu verwendenden Kennungen <DIOID> sind 1 bis 4. Wenn die Kennung weggelassen wird, werden alle Leitungen abgefragt.

Wenn <InputOn>=0, ist das digitale Eingangssignal LOW/OFF, wenn <InputOn>=1, ist das digitale Eingangssignal HIGH/ON.

DPA (Reset Settings to Default)

Beschreibung: Setzt Parameterwerte und parameterunabhängige Einstellungen auf Werkseinstellungen zurück.

Format: DPA <Pswd> [{<ItemID> <PamID>}]

Argumente: <Pswd> ist das Kennwort zum Zurücksetzen des Speichers. Nähere Angaben siehe unten.

<ItemID> ist das Element, für das ein Parameter zurückzusetzen ist. Nähere Angaben siehe unten.

<PamID> ist die Parameterkennung, kann im Hexadezimal- oder Dezimalformat geschrieben werden. Nähere Angaben siehe unten.

Antwort: Keine

Fehlersuche: Unzulässige Elementkennung, falsche Parameter-ID, ungültiges Kennwort

Hinweise: DPA setzt im flüchtigen Speicher des C-887 die mit WPA gespeicherten Parameterwerte und die Einstellungen für Koordinatensysteme auf Werkseinstellungen zurück.

Beim C-887 entfällt für DPA die Angabe von <ItemID> und <PamID>.

Die Werkseinstellungen, die mit DPA geladen werden, sind unabhängig von den Einstellungen im permanenten Speicher, die mit WPA (S. 293) überschrieben werden können. Die (mit WPA gespeicherten) Einstellungen aus dem permanenten Speicher werden beim Einschalten oder Neustart des C-887 automatisch in den flüchtigen Speicher geladen.

Gültige Kennwörter:

100 Setzt die Werte aller Parameter und die Einstellungen für Koordinatensysteme (Details siehe Kennwort SKS) auf Werkseinstellungen zurück

SKS Einstellungen, die mit DPA SKS auf Werkseinstellungen zurückgesetzt werden:

- Achsorientierungs-Koordinatensystem (Typ KSB()): PI_BASE wird aktiviert
- Korrektur-Koordinatensystem (Typ KLD() oder KLF()): PI_Levelling wird aktiviert
- Betriebs-Koordinatensystem ZERO wird aktiviert und basiert auf PI_BASE und PI_Levelling
- Pivotpunkt (siehe SPI (S. 260)), Verfahrbereichsgrenzen der Achsen (siehe NLM (S. 248), PLM (S. 251) und SSL (S. 263)), Schrittweite für Bewegungen, die durch eine manuelle Bedieneinheit ausgelöst werden (siehe SST (S. 264))

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:	Die Anzahl verfügbarer Datenrekordertabellen kann mit TNR? (S. 273) abgefragt werden. Der C-887 hat 16 Datenrekordertabellen. Die maximale Anzahl der für die Datenaufzeichnung verfügbaren Punkte pro Datenrekordertabelle ist 262144 (Standard: 8192). Mit HDR? (S. 200) erhalten Sie eine Liste aller verfügbaren Aufzeichnungsoptionen und Informationen über zusätzliche Parameter und Befehle für die Datenaufzeichnung. Nähere Informationen finden Sie unter "Datenrekorder" (S. 99).

Aufzeichnungsoptionen für die entsprechenden Datenquellen:	<Source>	<RecOption>
	Achse / Hexapod-Bein	0 = Nothing is recorded 1 = Commanded position of axis 2 = Real position of axis 3 = Position error of axis 8 = Measurement time 70 = Commanded velocity of axis 71 = Commanded acceleration of axis 73 = Motor output of axis 74 = Current proportion to the current error of axis 75 = Current integrated position error of axis 76 = Current derivative position error of axis 80 = Status register of axis Weitere Informationen finden Sie in "Statusregister für Hexapod-Beine und Achsen A und B" (S. 386). 84=Real position of second sensor in axis 86 = FIFO size for continuous position mode 87 = Commanded position for continuous position mode

Continuous position mode steht für "zyklische Übertragung von Zielpositionen" (S. 39); aufgezeichnet werden nicht die empfangenen Werte, sondern bereits vom C-887 bearbeitete Werte.

Eingangs-signal-kanal	17 = Input value of channel, calculated, in volt. 18 = Input value of channel, directly from channel, without dimension. 150 = Input value of channel, calculated in Volt, input calculation made with SIC

HDR? zeigt an, für welche Datenquellen die einzelnen Aufzeichnungsoptionen verwendet werden können. Für Details zu vorhandenen Achsen und Kanälen siehe "Kommandierbare Elemente" (S. 27).

Die Standardeinstellungen des Datenrekorders sind wie folgt:

```
drc?
1=X 1
2=X 2
3=Y 1
4=Y 2
5=Z 1
6=Z 2
7=U 1
8=U 2
9=V 1
10=V 2
11=W 1
12=W 2
13=1 8
14=0 0
15=0 0
16=0 0
```

Beispiel: Senden: DRC 4 X 2

Die aktuelle Position der Achse X soll in Datenrekordertabelle 4 aufgezeichnet werden.

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. 200) 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.

Hinweise: Die Anzahl der Punkte wird für die Datenrekordertabelle auf Null zurückgesetzt, wenn ihre Konfiguration mit DRC (S. 168) geändert wird.

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 für <NumberOfPoints> der Wert -1 gesetzt wird, werden alle gültigen Punkte der ausgewählten Tabellen ausgelesen

Wenn <RecTableID> weggelassen wird, werden alle Tabellen ausgelesen, deren mit DRC (S. 168) gesetzte Aufzeichnungsoption von null verschieden ist

Mit HDR? (S. 200) erhalten Sie eine Liste aller verfügbaren Aufzeichnungs- und Triggeroptionen sowie Informationen über zusätzliche Parameter und Befehle für die Datenaufzeichnung.

Nähere Informationen siehe "Datenrekorder" (S. 99).

Beispiel:

```
drc 1 X 1 2 X 2
drt 1 1 1
mov x 2
drr? 1 5 1 2 13
# REM data recorded with C-887 controller
#
# TYPE = 1
# SEPARATOR = 32
# DIM = 3
# NDATA = 5
# SAMPLE_TIME = 0.001000
#
# NAME0 = TARGET POSITION X
# NAME1 = REAL POSITION X
```

```
# NAME2 = MEASUREMENT TIME Strut 1
# DISPUNIT0 = mm
# DISPUNIT1 = mm
# DISPUNIT2 = sec
# END_HEADER
0.004220971838 -0.000364157866 0
0.064530499279 0.036015000194 0.005040000193
0.145174950361 0.140960231423 0.010040000081
0.232607111335 0.216114446521 0.015200000256
0.309781700373 0.30354321003 0.020020000637
```

Mit DRC (S. 168) wird festgelegt, dass die aktuelle Position der Achse X in der Datenrekordertabelle 1 und die kommandierte Position der Achse X in der Datenrekordertabelle 2 aufgezeichnet werden soll.

Mit DRT (S. 173) wird festgelegt, dass eine Aufnahme durch den nächsten Bewegungsbefehl, z. B. durch MOV (S. 242), ausgelöst werden soll.

MOV löst die Bewegung der Achse X zur Position 2 aus.

Mit DRR? werden die ersten fünf Punkte der Datenrekordertabellen 1, 2 und 13 abgefragt.

Der Datenrekorder ist standardmäßig so konfiguriert, dass die Zeit in der Datenrekordertabelle 13 aufgenommen wird.

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:

Unabhängig von der mit <RecTableID> gewählten Datenrekordertabelle wird die mit <Triggersource> gewählte Triggeroption immer für alle Datenrekordertabellen eingestellt.

Mit HDR? (S. 200) erhalten Sie eine Liste aller verfügbaren Aufzeichnungs- und Triggeroptionen sowie zusätzliche Informationen über die Datenaufzeichnung.

IMP (S. 209), STE (S. 266), WGO (S. 289) und WGR (S. 291) lösen unabhängig von den DRT-Einstellungen jeweils eine Aufzeichnung durch den Datenrekorder aus (WGR nur, wenn der Funktionsgenerator aktiv ist).

Die Datenrekorderkonfiguration, d. h. die Zuweisung der Datenquellen und der Aufzeichnungsoptionen zu den Rekordertabellen, nehmen Sie mit DRC (S. 168) vor. Tatsächlich gestartet wird eine Aufzeichnung nur für die Datenrekordertabellen, deren mit DRC eingestellte Aufzeichnungsoption von null verschieden ist.

Die Aufzeichnung endet, wenn die maximale Punktzahl erreicht ist (vorgegeben durch den Parameter ***Data Recorder Points Per Table***, ID 0x16000201).

Nähere Informationen finden Sie unter "Datenrekorder" (S. 99).

Verfügbare Triggeroptionen:

0 = No trigger. Exception: STE, IMP, WGO and WGR always trigger data recording;
Datenaufzeichnung wird nicht ausgelöst, und eine noch laufende Datenaufzeichnung wird fortgeführt, bis die maximale Punktanzahl erreicht ist. Ausnahme: STE, IMP, WGO (wenn zum Starten der Funktionsgeneratorausgabe verwendet) und WGR lösen immer eine Aufnahme aus. <Value> muss ein Dummy sein.

1 = Trigger with next command that changes the position;
z. B. MOV (S. 242), MVR (S. 246); <Value> muss ein Dummy sein. Standardmäßig eingestellt.

2 = Trigger with next command;
setzt Trigger nach Ausführung auf die Triggeroption 0; <Value> muss ein Dummy sein.

4 = Trigger immediately;
setzt Trigger nach Ausführung auf die Triggeroption 0; <Value> muss ein Dummy sein

6 = Trigger with next command that changes the position; z. B. MOV, MVR;setzt Trigger nach Ausführung auf die Triggeroption 0; <Value> muss ein Dummy sein.

Ist die Triggeroption 2, 4 oder 6 eingestellt, wird nach dem Auslösen der Aufzeichnung automatisch auf die Triggeroption 0 umgeschaltet. Dadurch läuft die Aufzeichnung so lange weiter, bis die maximale Punktanzahl der Datenrekordertabelle(n) erreicht ist. So können mehrere Bewegungen nacheinander aufgezeichnet werden. Die Triggeroptionen 2, 4 oder 6 sind z. B. dann sinnvoll, wenn das Dynamikprofil durch aufeinander folgende MOV-Befehle festgelegt werden soll (siehe MOV).

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. 173).

ECO? (Echo a String)

Beschreibung: Gibt einen String aus.

ECO? kann verwendet werden, um die Kommunikation zu testen.

Format: ECO? <String>

Argumente: <String> ist eine beliebige Zeichenkombination aus Buchstaben und Zahlen

Antwort: <String> LF

Hinweis: <String> kann aus maximal 100 Zeichen bestehen.

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. 299) 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.

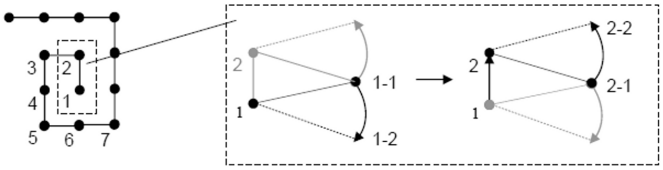
Wenn die Ursache eines Fehlers weiterhin besteht, wird der entsprechende Fehlercode sofort nach der Abfrage mit ERR? wieder neu gesetzt.

Beispiel: Im C-887 fehlt die Konfigurationsdatei für den angeschlossenen Hexapod. Der Fehlercode 233 wird trotz Abfrage so lange gesetzt, bis eine passende Konfigurationsdatei in den C-887 geladen wurde.

FIO (Fast Input-Output Alignment Procedure)

Beschreibung: Startet eine Scanprozedur zur Ausrichtung optischer Elemente (z. B. Lichtleitfasern), deren Eingang und Ausgang auf der gleichen Seite liegen. Innerhalb des Elements wird Licht vom Eingang zum Ausgang geleitet. Damit ein Sensor, der mit einem analogen Eingangskanal des Controllers verbunden ist, das Signal am Ausgang des optischen Elements mit maximaler Intensität empfängt, müssen Eingang und Ausgang des optischen Elements gleichzeitig ausgerichtet werden.

Fährt in einer vorgegebenen Fläche eine lineare Spirale mit einer vorgegebenen Schrittweite ab, siehe Abbildung



Die Ebene, in der die vorgegebene Fläche liegt, kann durch eines der folgenden Achsenpaare definiert sein:

X Z

Y Z

X Y

Die Scanprozedur startet an der aktuellen Position zum Zeitpunkt des Befehlsempfangs („Ausgangsposition").

Am Startpunkt und nach jedem Schritt wird in einem vorgegebenen Winkel um den Pivotpunkt herum gescannt, bis ein analoges Eingangssignal einen vorgegebenen Intensitätsschwellenwert erreicht. Dies entspricht einem Winkelscan um die Achse herum, die senkrecht zur abgefahrenen Fläche steht.

Der Pivotpunkt muss sich an der Position befinden, die dem Zentrum des Eingangs oder dem Zentrum des Ausgangs des optischen Elements entspricht. Mit SPI (S. 260) können die Koordinaten des Pivotpunkts gesetzt werden.

FIO (und damit die Bewegung der Plattform) ist erfolgreich beendet, wenn das analoge Eingangssignal den vorgegebenen Intensitätsschwellenwert erreicht.

FIO wird in folgenden Fällen erfolglos beendet:

- Intensitätsschwellenwert wurde in der vorgegebenen Fläche nicht erreicht: die Bewegungsplattform kehrt zurück zur Ausgangsposition
- #24 (S. 156), STP (S. 267) oder HLT (S. 204) wurde gesendet: die Bewegungsplattform bleibt an der aktuellen Position stehen

Mit FSS? (S. 198) kann geprüft werden, ob eine Scanprozedur erfolgreich beendet wurde.

Um zu prüfen, ob eine Scanprozedur noch andauert, kann mit #5 (S. 153) der Bewegungsstatus der Achsen abgefragt werden.

Format:

FIO <AxisID> <Distance> <AxisID> <Distance>
["S" <LinearSpiralStepSize>] ["AR" <AngularScanSize>]
["L" <Threshold>] ["A" <AnalogInputID>]

Argumente:

<AxisID>: ist eine Achse des Controllers, Format: String
Zulässig sind die Achsen X, Y und Z
<Distance>: abzufahrende Strecke entlang der Achse, Format: double

"S": erforderliches Schlüsselwort zur Eingabe von <LinearSpiralStepSize>
<LinearSpiralStepSize>: Schrittweite, in der die Plattform eine spiralförmige Strecke abfährt, Format: double

"AR": erforderliches Schlüsselwort zur Eingabe von <AngularScanSize>
<AngularScanSize>: Winkel, um den Pivotpunkt herum, in dem gescannt wird, in Grad, Format: double

"L": erforderliches Schlüsselwort zur Eingabe von <Threshold>
<Threshold>: Intensitätsschwellenwert des analogen Eingangssignals, in V, Format: double

"A": erforderliches Schlüsselwort zur Eingabe von <AnalogInputID>
<AnalogInputID>: ist die Kennung des analogen Eingangssignals, dessen Intensitätsmaximum gesucht wird, Format: Integer

Antwort:

Keine

Hinweise:

Die physikalische Einheit, in der <Distance> und <LinearSpiralStepSize> anzugeben sind, kann mit PUN? (S. 253) abgefragt werden.

Die durch die Werte für <Distance> vorgegebene Fläche muss quadratisch sein.

Folgende Standardwerte werden verwendet, wenn die entsprechenden Argumente weggelassen werden:

<LinearSpiralStepSize>: 0,01 mm

<AngularScanSize: 0,2 Grad

<Threshold>: 1,0 V

<AnalogInputID>: 1

Diese Standardwerte wurden ausschließlich für die Hexapod-Modelle H-810, H-811 und H-206 geprüft. Für andere Hexapod-Modelle müssen die idealen Parameter experimentell ermittelt werden.

Je kleiner die Systemgeschwindigkeit mit VLS (S. 279) während einer Scanprozedur gesetzt ist, desto größer ist die Genauigkeit, mit der der Schwellenwert der Intensität gefunden wird.

Geschwindigkeiten im Bereich von unter 1 mm/s werden empfohlen.

Hinweis:

Bei zu großen Werten für Strecken oder Winkel bewegt sich die Plattform während einer Scanprozedur auf einer undefinierten Bahn und kann verkippen. Dadurch sind Kollisionen und ein unbefriedigendes Ergebnis der Scanprozedur möglich. Maßnahmen zur Vermeidung des Verkippens:

- Wählen Sie für <AngularScanSize> Werte bis 0,2 Grad
- Richten Sie die Bewegungsplattform vor der Scanprozedur schon passend aus.
- Verwenden Sie passende Halterungen für die auszurichtenden Ein- und/oder Ausgänge des optischen Elements auf der Bewegungsplattform, so dass die Bewegung während der Scanprozedur nur über kleine Strecken oder Winkel erfolgt.

Wenn der Parameter ***Trajectory Source*** (ID 0x19001900) auf 1 gesetzt ist, muss das Dynamikprofil durch aufeinander folgende MOV-Befehle vorgegeben werden, und FIO ist nicht zulässig.

Weitere Informationen siehe "Bewegungen des Hexapods" (S. 34).

Beispiel: FIO Y 0.1 Z 0.1 S 0.01 AR 0.1 L 1 A 2

Startet eine Scanprozedur entlang einer linearen Spirale in der YZ-Ebene:

- Seitenlänge der quadratischen Fläche: 0,1 mm
- Schrittweite: 10 µm
- Winkel um den Pivotpunkt: 0,1 Grad (für die Achse U)
- Intensitätsschwellenwert: 1 V
- Kennung des analogen Eingangskanals, dessen Intensitätsmaximum gesucht wird: 2

FLM (Fast Line Scan to Maximum)

Beschreibung: Startet eine Scanprozedur zum Ermitteln des globalen Intensitätsmaximums eines analogen Eingangssignals.

Scannt eine vorgegebene Strecke entlang einer Achse vollständig auf die Intensität des analogen Eingangssignals ab. Damit wird verhindert, dass bei mehreren Intensitätsmaxima statt des globalen Maximums lediglich ein lokales Maximum gefunden wird.

Die Richtung der Scanprozedur sowie die Start- und Endposition der Strecke können durch das Argument <ScanDirection> vorgegeben werden.

FLM (und damit die Bewegung der Plattform) ist erfolgreich beendet, wenn die beiden folgenden Bedingungen erfüllt sind:

- Das analoge Eingangssignal hat den vorgegebenen Intensitätsschwellenwert auf der vorgegebenen Strecke mindestens einmal erreicht.
- Die Bewegungsplattform ist von der Endposition der Strecke zur Position mit maximaler Intensität zurückgekehrt.

FLM wird in folgenden Fällen erfolglos beendet:

- Intensitätsschwellenwert wurde auf der vorgegebenen Strecke nicht erreicht: die Bewegungsplattform kehrt von der Endposition der Strecke zurück zur Ausgangsposition (aktuelle Position zum Zeitpunkt des Befehlsempfangs).
- #24 (S. 156), STP (S. 267) oder HLT (S. 204) wurde gesendet: die Bewegungsplattform bleibt an der aktuellen Position stehen

Mit FSS? (S. 198) kann geprüft werden, ob eine Scanprozedur erfolgreich beendet wurde.

Um zu prüfen, ob eine Scanprozedur noch andauert, kann mit #5 (S. 153) der Bewegungsstatus der Achsen abgefragt werden.
Mit FLS (S. 182) kann eine ähnliche Scanprozedur gestartet werden. Im Gegensatz zu FLM ist FLS bereits erfolgreich beendet, wenn der Intensitätsschwellenwert erstmalig erreicht ist.

Format: FLM <AxisID> <Distance> ["L" <Threshold>]
["A" <AnalogInputID>] ["D" <ScanDirection>]

Argumente: <AxisID>: ist eine Achse des Controllers, Format: String
Zulässig sind die Achsen X, Y, Z, U, V, W.
<Distance>: abzuscannende Strecke entlang der Achse, Format: double

"L": erforderliches Schlüsselwort zur Eingabe von <Threshold>
<Threshold>: Intensitätsschwellenwert des analogen Eingangssignals, in V, Format: double

"A": erforderliches Schlüsselwort zur Eingabe von <AnalogInputID>
<AnalogInputID>: ist die Kennung des analogen Eingangssignals, dessen Intensitätsmaximum gesucht wird, Format: Integer

"D": erforderliches Schlüsselwort zur Eingabe von <ScanDirection>
<ScanDirection>: gibt die Richtung der Scanprozedur sowie die Start- und Endposition der Strecke vor:
0: Scanprozedur wird zentriert um die aktuelle Position herum durchgeführt, in positive Richtung.
Startposition = aktuelle Position - <Distance>/2
Endposition = aktuelle Position + <Distance>/2

1: Scanprozedur in positive Richtung:
Startposition = aktuelle Position
Endposition = aktuelle Position + <Distance>
-1: Scanprozedur in negative Richtung:
Startposition = aktuelle Position
Endposition = aktuelle Position - <Distance>

Antwort: Keine

Hinweise: Die physikalische Einheit, in der <Distance> anzugeben ist, kann mit PUN? (S. 253) abgefragt werden.

Folgende Standardwerte werden verwendet, wenn die entsprechenden Argumente weggelassen werden:
<Threshold>: 1,0 V
<AnalogInputID>: 1
<ScanDirection>: 0 (Scanprozedur zentriert um die aktuelle Position herum, in positive Richtung)

Je kleiner die Systemgeschwindigkeit mit VLS (S. 279) während einer Scanprozedur gesetzt ist, desto größer ist die Genauigkeit, mit der das Intensitätsmaximum gefunden wird.
Geschwindigkeiten im Bereich von unter 1 mm/s werden empfohlen.

Bei zu großen Werten für Strecken oder Winkel bewegt sich die Plattform während einer Scanprozedur auf einer undefinierten Bahn und kann verkippen. Dadurch sind Kollisionen und ein unbefriedigendes Ergebnis der Scanprozedur möglich. Maßnahmen zur Vermeidung des Verkippens:

- Wählen Sie für <Distance> passende Werte. Für die Hexapod-Modelle H-810, H-811 und H-206 sollten 0,2 mm bzw. 0,2 Grad nicht überschritten werden; für andere Hexapod-Modelle sowie bei geänderten Einstellungen für Koordinatensysteme und Pivotpunkt müssen die idealen Werte experimentell ermittelt werden.
- Stellen Sie die Geschwindigkeit für die Bewegungsplattform des Hexapods so gering wie möglich ein (mit dem Befehl VLS).
- Richten Sie die Bewegungsplattform vor der Scanprozedur schon passend aus.
- Verwenden Sie passende Halterungen für die auszurichtenden Ein- und/oder Ausgänge des optischen Elements auf der Bewegungsplattform, so dass die Bewegung während der Scanprozedur nur über kleine Strecken oder Winkel erfolgt.

Wenn der Parameter ***Trajectory Source*** (ID 0x19001900) auf 1 gesetzt ist, muss das Dynamikprofil durch aufeinander folgende MOV-Befehle vorgegeben werden, und FLM ist nicht zulässig.

Weitere Informationen siehe "Bewegungen des Hexapods" (S. 34).

Beispiel:

FLM Z 0.2 L 1.2 A 2 D 0

Startet eine Scanprozedur entlang der Z-Achse:

- Strecke: 0,2 mm
- Intensitätsschwellenwert: 1,2 V
- Kennung des analogen Eingangskanals, dessen Intensitätsmaximum gesucht wird: 2
- Scanprozedur verläuft zentriert um die aktuelle Position, in positive Richtung

FLS (Fast Line Scan)

Beschreibung:

Startet eine Scanprozedur, bei der über eine vorgegebene Strecke entlang einer Achse gescannt wird, bis das analoge Eingangssignal einen vorgegebenen Intensitätsschwellenwert erreicht.

Die Richtung der Scanprozedur sowie die Startposition der Strecke können durch das Argument <ScanDirection> vorgegeben werden.

Wenn der Intensitätsschwellenwert schon in der Ausgangsposition (aktuelle Position zum Zeitpunkt des Befehlsempfangs) erreicht ist, wird eine davon abweichende Startposition nicht angefahren.

FLS (und damit die Bewegung der Plattform) ist erfolgreich beendet,

wenn das analoge Eingangssignal den vorgegebenen Intensitätsschwellenwert erreicht. Wenn aufgrund des Bremswegs nötig, fährt die Plattform nach dem Anhalten zurück zur Position, an der der Intensitätsschwellenwert erreicht war.

FLS wird in folgenden Fällen erfolglos beendet:

- Intensitätsschwellenwert wurde auf der vorgegebenen Strecke nicht erreicht: die Bewegungsplattform kehrt von der Endposition der Strecke zurück zur Ausgangsposition.
- #24 (S. 156), STP (S. 267) oder HLT (S. 204) wurde gesendet: die Bewegungsplattform bleibt an der aktuellen Position stehen

Mit FSS? (S. 198) kann geprüft werden, ob eine Scanprozedur erfolgreich beendet wurde.

Um zu prüfen, ob eine Scanprozedur noch andauert, kann mit #5 (S. 153) der Bewegungsstatus der Achsen abgefragt werden.

Mit FLM (S. 180) kann eine ähnliche Scanprozedur gestartet werden. Im Gegensatz zu FLS scannt FLM entlang der gesamten vorgegebenen Strecke, so dass bei Vorliegen mehrerer lokaler Maxima das globale Maximum gefunden wird.

Format: FLS <AxisID> <Distance> ["L" <Threshold>] ["A" <AnalogInputID>] ["D" <ScanDirection>]

Argumente: <AxisID>: ist eine Achse des Controllers, Format: String
Zulässig sind die Achsen X, Y, Z, U, V. W.
<Distance>: abzuscannende Strecke entlang der Achse, Format: double

"L": erforderliches Schlüsselwort zur Eingabe von <Threshold>
<Threshold>: Intensitätsschwellenwert des analogen Eingangssignals, in V, Format: double

"A": erforderliches Schlüsselwort zur Eingabe von <AnalogInputID>
<AnalogInputID>: ist die Kennung des analogen Eingangssignals, dessen Intensitätsmaximum gesucht wird, Format: Integer

"D": erforderliches Schlüsselwort zur Eingabe von <ScanDirection>
<ScanDirection>: gibt die Richtung der Scanprozedur sowie die Start- und Endposition der Strecke vor:
0: Scanprozedur wird zentriert um die aktuelle Position herum durchgeführt, in positive Richtung.
Startposition = aktuelle Position - <Distance>/2
Endposition = aktuelle Position + <Distance>/2

1: Scanprozedur in positive Richtung:
Startposition = aktuelle Position
Endposition = aktuelle Position + <Distance>

-1: Scanprozedur in negative Richtung:
Startposition = aktuelle Position
Endposition = aktuelle Position - <Distance>

Antwort: Keine

Hinweise: Die physikalische Einheit, in der <Distance> anzugeben ist, kann mit PUN? abgefragt werden.

Folgende Standardwerte werden verwendet, wenn die entsprechenden Argumente weggelassen werden:
<Threshold>: 1,0 V
<AnalogInputID>: 1.
<ScanDirection>: 0 (Scanprozedur zentriert um die aktuelle Position herum, in positive Richtung)

Je kleiner die Systemgeschwindigkeit mit VLS (S. 279) während einer Scanprozedur gesetzt ist, desto größer ist die Genauigkeit, mit der das Intensitätsmaximum gefunden wird.
Geschwindigkeiten im Bereich von unter 1 mm/s werden empfohlen.

Bei zu großen Werten für Strecken oder Winkel bewegt sich die Plattform während einer Scanprozedur auf einer undefinierten Bahn und kann verkippen. Dadurch sind Kollisionen und ein unbefriedigendes Ergebnis der Scanprozedur möglich. Maßnahmen zur Vermeidung des Verkippens:

- Wählen Sie für <Distance> passende Werte. Für die Hexapod-Modelle H-810, H-811 und H-206 sollten 0,2 mm bzw. 0,2 Grad nicht überschritten werden; für andere Hexapod-Modelle sowie bei geänderten Einstellungen für Koordinatensysteme und Pivotpunkt müssen die idealen Werte experimentell ermittelt werden.
- Stellen Sie die Geschwindigkeit für die Bewegungsplattform des Hexapods so gering wie möglich ein (mit dem Befehl VLS).
- Richten Sie die Bewegungsplattform vor der Scanprozedur schon passend aus.
- Verwenden Sie passende Halterungen für die auszurichtenden Ein- und/oder Ausgänge des optischen Elements auf der Bewegungsplattform, so dass die Bewegung während der Scanprozedur nur über kleine Strecken oder Winkel erfolgt.

Wenn der Parameter ***Trajectory Source*** (ID 0x19001900) auf 1 gesetzt ist, muss das Dynamikprofil durch aufeinander folgende MOV-Befehle vorgegeben werden, und FLS ist nicht zulässig.

Weitere Informationen siehe "Bewegungen des Hexapods" (S. 34).

Beispiel: FLS Z 0.2 L 1 A 2 D 0

Startet eine Scanprozedur entlang der Z-Achse:

- Strecke: 0,2 mm
- Intensitätsschwellenwert: 1 V
- Kennung des analogen Eingangskanals, dessen Intensitätsmaximum gesucht wird: 2
- Scanprozedur verläuft zentriert um die aktuelle Position, in positive Richtung

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: Für Achsen mit inkrementellen Sensoren können Bewegungen erst nach einer erfolgreichen Referenzierungsfahrt kommandiert werden (auch als "Initialisierung" bezeichnet).

Das Verhalten der Achsen des Hexapods nach der Referenzierungsfahrt wird durch die Parameter ***Behaviour After Reference Move*** (ID 0x07030401) und ***Target For Motion After Reference Move*** (ID 0x07030402) festgelegt. Entsprechend der Parameterwerte können die Achsen der Plattform z. B. nach der Referenzierungsfahrt automatisch zu einer vorgegebenen Position bewegt werden.

- Wert des Parameters 0x07030401 = 0: Achse bleibt nach der Referenzierungsfahrt in der Referenzposition.
- Wert des Parameters 0x07030401 = 1: Achse fährt nach der Referenzierungsfahrt zur Zielposition, die durch Parameter 0x07030402 vorgegeben ist.

Für Achsen mit absolut messenden Sensoren ist keine Referenzierungsfahrt erforderlich. Die Verwendung des Befehls FRF

wird für diese Achsen trotzdem empfohlen. FRF startet für Achsen mit absolut messenden Sensoren keine Referenzierungsfahrt, sondern setzt die Zielpositionen auf die aktuellen Positionswerte. Zusätzlich werden die oben beschriebenen Parameterwerte wirksam, so dass die Achsen z. B. zu einer definierten "Ausgangsposition" bewegt werden können.

Während einer Referenzierungsfahrt bewegt sich der Hexapod auf unvorhersehbare Weise. Es findet **keine** Kollisionsprüfung und -vermeidung statt, selbst wenn eine Konfiguration zur Kollisionsvermeidung auf dem C-887 abgelegt wurde (z. B. mit der kostenpflichtigen PIVeriMove Hexapod Software zur Kollisionsprüfung). Verfahrbereichsgrenzen, die mit den Befehlen NLM (S. 248) und PLM (S. 251) für die Bewegungsplattform des Hexapods gesetzt wurden, werden während der Referenzierungsfahrt ignoriert.

Für die Achsen der Bewegungsplattform des Hexapods (X, Y, Z, U, V, W) erfolgt immer eine gemeinsame Referenzierungsfahrt. Daher reicht für den Start der Referenzierungsfahrt der Bewegungsplattform die Angabe einer einzigen Achse aus, z. B.:
FRF X

Für die Achsen der Bewegungsplattform des Hexapods (X, Y, Z, U, V, W) schaltet FRF auch den Servomodus ein.

FRF kann durch #24 (S. 156), STP (S. 267) und HLT (S. 204) abgebrochen werden.

Verwenden Sie FRF? (S. 186), um zu prüfen, ob die Referenzierungsfahrt erfolgreich war.

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
Hinweis:	Achsen gelten als "referenziert", wenn eine Referenzierungsfahrt erfolgreich durchgeführt wurde mit FRF (S. 185), oder wenn die Achsen mit absolut messenden Sensoren ausgestattet sind.

FSA (Fast Scan with Automated Alignment)

Beschreibung: Startet eine Scanprozedur zum Ermitteln des Intensitätsmaximums eines analogen Eingangssignals in einer Ebene. Die Suche besteht aus zwei Teilprozeduren:

„Grobanteil"; entspricht der Prozedur, die mit dem Befehl FSC (S. 190) gestartet wird

„Feinanteil"; entspricht der Prozedur, die mit dem Befehl AAP (S. 158) gestartet wird

Der Feinanteil wird nur ausgeführt, wenn der Grobanteil zuvor erfolgreich beendet wurde. Für detaillierte Beschreibungen der beiden Teilprozeduren siehe AAP und FSC.

Hinweis:

Für den Feinanteil wird der Scanbereich unter Verwendung der mit FSA vorgegebenen Werte für <Distance1> und <Distance2> neu gesetzt, so dass sich die Startposition des Feinanteils im Zentrum der vorgegebenen Fläche befindet. Dadurch kann sich der Scanbereich auf maximal das Doppelte des ursprünglichen Bereichs vergrößern.

- Stellen Sie sicher, dass sich die Bewegungsplattform auch außerhalb des ursprünglich vorgegebenen Scanbereichs gefahrlos bewegen kann.

Die Ebene, in der der Scanbereich liegt, kann durch eines der folgenden Achsenpaare definiert sein:

X Z

Y Z

X Y

FSA (und damit die Bewegung der Plattform) ist erfolgreich beendet, wenn der Feinanteil erfolgreich beendet ist, d. h. wenn drei aufeinanderfolgende Überprüfungen bestätigt haben, dass an der aktuellen Position das lokale Intensitätsmaximum vorliegt. Durch die Überprüfungen soll der Einfluß eines eventuell überlagerten Rauschsignals vermindert werden.

FSA wird in folgenden Fällen erfolglos beendet:

- Während des Grobanteils wurde der Intensitätsschwellenwert im Scanbereich nicht erreicht: die Bewegungsplattform kehrt zurück zur Ausgangsposition.
- Während des Feinanteils wurde kein Gradient über die Intensität gefunden: die Bewegungsplattform kehrt zurück zur Startposition des Feinanteils.
- Während des Feinanteils würde die vorgegebene Fläche überschritten: die Bewegungsplattform kehrt zurück zur Startposition des Feinanteils.
- #24 (S. 156), STP (S. 267) oder HLT (S. 204) wurde gesendet: die Bewegungsplattform bleibt an der aktuellen Position stehen.

Mit FSS? (S. 198) kann geprüft werden, ob eine Scanprozedur erfolgreich beendet wurde.

Um zu prüfen, ob eine Scanprozedur noch andauert, kann mit #5 (S. 153) der Bewegungsstatus der Achsen abgefragt werden.

Format:

FSA <Axis1ID> <Distance1> <Axis2ID> <Distance2>
["L" <Threshold>] ["S" <ScanLineDistance>]
["SA" <StepSize>] ["A" <AnalogInputID>]

Argumente:

<Axis1ID>: ist eine Achse des Controllers, Format: String
Zulässig sind die Achsen X, Y, und Z.
Während des Grobanteils wird in dieser Achse von Scanlinie zu Scanlinie um die Strecke <ScanlineDistance> gefahren.

<Distance1>: Seitenlänge des Scanbereichs entlang der Achse <Axis1ID>, Format: double

<Axis2ID>: ist eine Achse des Controllers, Format: String
Zulässig sind die Achsen X, Y, und Z.
Während des Grobanteils liegen die Scanlinien in dieser Achse.

<Distance2>: Seitenlänge des Scanbereichs entlang der Achse <Axis2ID>, Format: double

"L": erforderliches Schlüsselwort zur Eingabe von <Threshold>

<Threshold>: Intensitätsschwellenwert des analogen Eingangssignals, in V, Format: double

"S": erforderliches Schlüsselwort zur Eingabe von <ScanLineDistance>

<ScanLineDistance>: Abstand zwischen den Scanlinien, wird nur während des Grobanteils verwendet. Format: double

"SA": erforderliches Schlüsselwort zur Eingabe von <StepSize>

<StepSize>: Startwert für die Schrittweite, wird nur während des Feinanteils verwendet, Format: double

"A": erforderliches Schlüsselwort zur Eingabe von <AnalogInputID>

<AnalogInputID>: ist die Kennung des analogen Eingangssignals, dessen Intensitätsmaximum gesucht wird, Format: Integer

Antwort: Keine

Hinweise: Die physikalische Einheit, in der <Distance1>, <Distance2>, <ScanLineDistance> und <StepSize> anzugeben sind, kann mit PUN? (S. 253) abgefragt werden.

Die Werte für <Distance1> und <Distance2> müssen identisch sein.

Folgende Standardwerte werden verwendet, wenn die entsprechenden Argumente weggelassen werden:
<Threshold>: 1,0 V
<ScanLineDistance>: 0,01 mm
<StepSize>: 0,001 mm
<AnalogInputID>: 1
Diese Standardwerte wurden ausschließlich für die Hexapod-Modelle H-810, H-811 und H-206 geprüft. Für andere Hexapod-Modelle müssen die idealen Parameter experimentell ermittelt werden.

Je kleiner die Systemgeschwindigkeit mit VLS (S. 279) während einer Scanprozedur gesetzt ist, desto größer ist die Genauigkeit, mit der das Intensitätsmaximum gefunden wird.
Geschwindigkeiten im Bereich von unter 1 mm/s werden empfohlen.

Bei zu großen Werten für Strecken bewegt sich die Plattform während einer Scanprozedur auf einer undefinierten Bahn und kann verkippen. Dadurch sind Kollisionen und ein unbefriedigendes Ergebnis der Scanprozedur möglich. Maßnahmen zur Vermeidung des Verkippens:

- Wählen Sie für <Distance1> und <Distance2> Werte bis 0,2 mm.
- Richten Sie die Bewegungsplattform vor der Scanprozedur schon passend aus.
- Verwenden Sie passende Halterungen für die auszurichtenden Ein- und/oder Ausgänge des optischen Elements auf der Bewegungsplattform, so dass die Bewegung während der Scanprozedur nur über kleine Strecken erfolgt.

Wenn der Parameter ***Trajectory Source*** (ID 0x19001900) auf 1 gesetzt ist, muss das Dynamikprofil durch aufeinander folgende MOV-Befehle vorgegeben werden, und FSA ist nicht zulässig.
Weitere Informationen siehe "Bewegungen des Hexapods" (S. 34).

Beispiel: FSA Y 0.2 Z 0.2 L 1 S 0.01 SA 0.001 A 2

Startet eine Scanprozedur in der YZ-Ebene:

- Seitenlänge der quadratischen Fläche: 0,2 mm
- Intensitätsschwellenwert für den Grobanteil: 1 V
  Wenn 1 V erreicht wird, stoppt die Bewegungsplattform, anschließend wird der Scanbereich neu gesetzt, und der Feinanteil wird gestartet.
- Abstand zwischen den (in der Z-Achse liegenden) Scanlinien für den Grobanteil: 10 µm
- Startwert der Schrittweite für den Feinanteil: 1 µm
- Kennung des analogen Eingangskanals, dessen Intensitätsmaximum gesucht wird: 2

FSC (Fast Scan with Abort)

Beschreibung: Startet eine Scanprozedur, bei der über eine vorgegebene Fläche („Scanbereich") gescannt wird, bis das analoge Eingangssignal einen vorgegebenen Intensitätsschwellenwert erreicht.

Die mit FSC gestartete Scanprozedur entspricht dem „Grobanteil" der Scanprozedur, die mit dem Befehl FSA (S. 187) gestartet wird.

Die Ebene, in der der Scanbereich liegt, kann durch eines der folgenden Achsenpaare definiert sein:

X Z
Y Z
X Y
U W
V W
U V

Der Scanbereich ist um die aktuelle Position zum Zeitpunkt des Befehlsempfangs („Ausgangsposition") zentriert. Die Größe des Scanbereichs wird durch die Werte für <Distance1> und <Distance2> vorgegeben.

Die Scanprozedur startet in derjenigen Ecke des Scanbereichs, in der gilt:

- Für <Axis1ID>:
  Startposition1 = Ausgangsposition1 - <Distance1>/2
- Für <Axis2ID>:
  Startposition2 = Ausgangsposition2 - <Distance2>/2

Wenn der Intensitätsschwellenwert schon in der Ausgangsposition erreicht ist, wird die Startposition nicht angefahren.

<Axis1ID> gibt die Achse vor, in der von Scanlinie zu Scanlinie gefahren wird, <Axis2ID> gibt die Achse vor, in der die Scanlinien liegen. Der Abstand zwischen den Scanlinien kann vorgegeben werden.

FSC (und damit die Bewegung der Plattform) ist erfolgreich beendet, wenn das analoge Eingangssignal den vorgegebenen Intensitätsschwellenwert erreicht. Wenn aufgrund des Bremswegs nötig, fährt die Plattform nach dem Anhalten zurück zur Position, an der der Intensitätsschwellenwert erreicht war.

FSC wird in folgenden Fällen erfolglos beendet:

- Intensitätsschwellenwert wurde im Scanbereich nicht erreicht: die Bewegungsplattform kehrt zurück zur Ausgangsposition.
- #24 (S. 156), STP (S. 267) oder HLT (S. 204) wurde gesendet: die Bewegungsplattform bleibt an der aktuellen Position stehen.

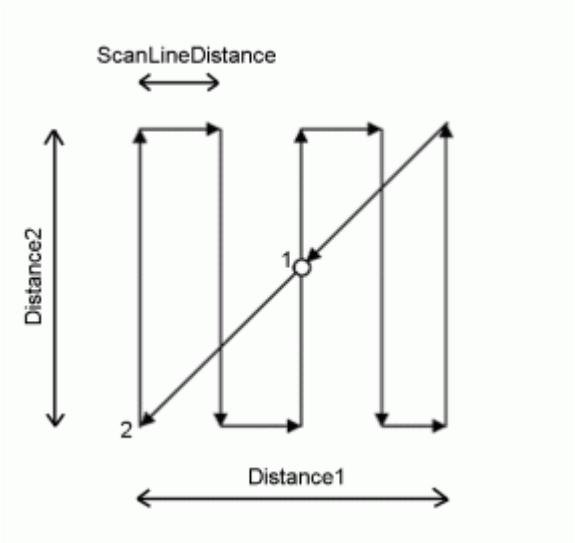


Tabelle 1: Bewegungsablauf, wenn der Intensitätsschwellenwert im Scanbereich nicht erreicht wurde

1 Ausgangsposition (= Endposition)
2 Startposition

Mit FSS? (S. 198) kann geprüft werden, ob eine Scanprozedur erfolgreich beendet wurde.

Um zu prüfen, ob eine Scanprozedur noch andauert, kann mit #5 (S. 153) der Bewegungsstatus der Achsen abgefragt werden.

Mit FSM (S. 195) kann eine ähnliche Scanprozedur gestartet werden. Im Gegensatz zu FSC scannt FSM den gesamten Scanbereich, so dass bei Vorliegen mehrerer lokaler Maxima das globale Maximum gefunden wird.

Format: FSC <Axis1ID> <Distance1> <Axis2ID> <Distance2> ["L" <Threshold>] ["S" <ScanLineDistance>] ["A" <AnalogInputID>]

Argumente: <Axis1ID>: ist die Achse, in der von Scanlinie zu Scanlinie um die Strecke <ScanlineDistance> gefahren wird. Format: String

Zulässig sind die Achsen X, Y, Z, U, V und W.

<Distance1>: Seitenlänge des Scanbereichs entlang der Achse <Axis1ID>, Format: double

<Axis2ID>: ist die Achse, in der die Scanlinien liegen, Format: String
Zulässig sind die Achsen X, Y, Z, U, V und W.

<Distance2>: Seitenlänge des Scanbereichs entlang der Achse <Axis2ID>, Format: double

"L": erforderliches Schlüsselwort zur Eingabe von <Threshold>

<Threshold>: Intensitätsschwellenwert des analogen Eingangssignals, in V, Format: double

"S": erforderliches Schlüsselwort zur Eingabe von <ScanLineDistance>

<ScanLineDistance>: Abstand zwischen zwei Scanlinien. Format: double

"A": erforderliches Schlüsselwort zur Eingabe von <AnalogInputID>

<AnalogInputID>: ist die Kennung des analogen Eingangssignals, dessen Intensitätsmaximum gesucht wird, Format: Integer

Antwort: Keine

Hinweise: Die physikalische Einheit, in der <Distance1>, <Distance2> und <ScanLineDistance> anzugeben sind, kann mit PUN? (S. 253) abgefragt werden.

Die Werte für <Distance1> und <Distance2> müssen identisch sein.

Folgende Standardwerte werden verwendet, wenn die entsprechenden Argumente weggelassen werden:
<Threshold>: 1,0 V
<ScanLineDistance>: 0,01
<AnalogInputID>: 1

Diese Standardwerte wurden ausschließlich für die Hexapod-Modelle H-810, H-811 und H-206 geprüft. Für andere Hexapod-Modelle müssen die idealen Parameter experimentell ermittelt werden.

Je kleiner die Systemgeschwindigkeit mit VLS (S. 279) während einer Scanprozedur gesetzt ist, desto größer ist die Genauigkeit, mit der der Schwellenwert der Intensität gefunden wird.
Geschwindigkeiten im Bereich von unter 1 mm/s werden empfohlen.

Bei zu großen Werten für Strecken oder Winkel bewegt sich die Plattform während einer Scanprozedur auf einer undefinierten Bahn und kann verkippen. Dadurch sind Kollisionen und ein unbefriedigendes Ergebnis der Scanprozedur möglich. Maßnahmen zur Vermeidung des Verkippens:

- Wählen Sie für <Distance1> und <Distance2> passende Werte. Für die Hexapod-Modelle H-810, H-811 und H-206 sollten 0,2 mm bzw. 0,2 Grad nicht überschritten werden; für andere Hexapod-Modelle sowie bei geänderten Einstellungen für Koordinatensysteme und Pivotpunkt müssen die idealen Werte experimentell ermittelt werden.
- Stellen Sie die Geschwindigkeit für die Bewegungsplattform des Hexapods so gering wie möglich ein (mit dem Befehl VLS).
- Richten Sie die Bewegungsplattform vor der Scanprozedur schon passend aus.
- Verwenden Sie passende Halterungen für die auszurichtenden Ein- und/oder Ausgänge des optischen Elements auf der Bewegungsplattform, so dass die Bewegung während der Scanprozedur nur über kleine Strecken oder Winkel erfolgt.

Wenn der Parameter ***Trajectory Source*** (ID 0x19001900) auf 1 gesetzt ist, muss das Dynamikprofil durch aufeinander folgende MOV-Befehle vorgegeben werden, und FSC ist nicht zulässig.

Weitere Informationen siehe "Bewegungen des Hexapods" (S. 34).

Beispiel: FSC Y 0.2 Z 0.2 L 1 S 0.05 A 2

Startet eine Scanprozedur in der YZ-Ebene:

- Seitenlänge der quadratischen Fläche: 0,2 mm
- Intensitätsschwellenwert: 1 V
- Abstand zwischen den (in der Z-Achse liegenden) Scanlinien: 50 µm
- Kennung des analogen Eingangskanals, dessen Intensitätsmaximum gesucht wird: 2

FSM (Fast Scan to Maximum)

Beschreibung: Startet eine Scanprozedur zum Ermitteln des globalen Intensitätsmaximums eines analogen Eingangssignals in einer Ebene.

Scannt eine vorgegebene Fläche („Scanbereich") vollständig auf die Intensität des analogen Eingangssignals ab. Damit wird verhindert, dass bei mehreren Intensitätsmaxima im Scanbereich statt des globalen Maximums lediglich ein lokales Maximum gefunden wird.

Die Ebene, in der der Scanbereich liegt, kann durch eines der folgenden Achsenpaare definiert sein:

X Z
Y Z
X Y
U W
V W
U V

Der Scanbereich ist um die aktuelle Position zum Zeitpunkt des Befehlsempfangs („Ausgangsposition") zentriert. Die Größe des Scanbereichs wird durch die Werte für <Distance1> und <Distance2> vorgegeben.
Die Scanprozedur startet in derjenigen Ecke des Scanbereichs, in der gilt:

- Für <Axis1ID>:
  Startposition1 = Ausgangsposition1 - <Distance1>/2
- Für <Axis2ID>:
  Startposition2 = Ausgangsposition2 - <Distance2>/2

<Axis1ID> gibt die Achse vor, in der von Scanlinie zu Scanlinie gefahren wird, <Axis2ID> gibt die Achse vor, in der die Scanlinien liegen. Der Abstand zwischen den Scanlinien kann vorgegeben werden.

FSM (und damit die Bewegung der Plattform) ist erfolgreich beendet, wenn die beiden folgenden Bedingungen erfüllt sind:

- Das analoge Eingangssignal hat den vorgegebenen Intensitätsschwellenwert im Scanbereich mindestens einmal erreicht.
- Die Bewegungsplattform ist von der Endposition des Scanbereichs zur Position mit maximaler Intensität zurückgekehrt.

FSM wird in folgenden Fällen erfolglos beendet:

- Intensitätsschwellenwert wurde im Scanbereich nicht erreicht: die Bewegungsplattform kehrt zurück zur Ausgangsposition.
- #24 (S. 156), STP (S. 267) oder HLT (S. 204) wurde gesendet: die Bewegungsplattform bleibt an der aktuellen Position stehen.

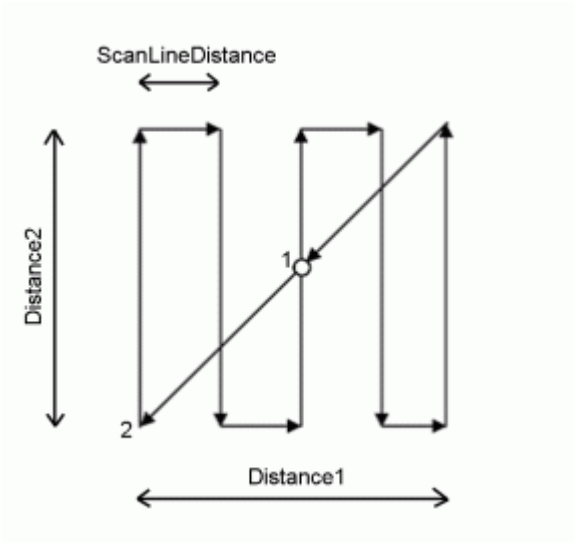


Tabelle 2: Bewegungsablauf, wenn der Intensitätsschwellenwert im Scanbereich nicht erreicht wurde

1 Ausgangsposition (= Endposition)
2 Startposition

Mit FSS? (S. 198) kann geprüft werden, ob eine Scanprozedur erfolgreich beendet wurde.

Um zu prüfen, ob eine Scanprozedur noch andauert, kann mit #5 (S. 153) der Bewegungsstatus der Achsen abgefragt werden.

Mit FSC (S. 190) kann eine ähnliche Scanprozedur gestartet werden. Im Gegensatz zu FSM ist FSC bereits erfolgreich beendet, wenn der Intensitätsschwellenwert erstmalig erreicht ist.

Format: FSM <Axis1ID> <Distance1> <Axis2ID> <Distance2> ["L" <Threshold>] ["S" <ScanLineDistance>] ["A" <AnalogInputID>]

Argumente: <Axis1ID>: ist die Achse, in der von Scanlinie zu Scanlinie um die Strecke <ScanlineDistance> gefahren wird. Format: String
Zulässig sind die Achsen X, Y, Z, U, V und W.

<Distance1>: Seitenlänge des Scanbereichs entlang der Achse <Axis1ID>, Format: double

<Axis2ID>: ist die Achse, in der die Scanlinien liegen, Format: String
Zulässig sind die Achsen X, Y, Z, U, V und W.

<Distance2>: Seitenlänge des Scanbereichs entlang der Achse <Axis2ID>, Format: double

"L": erforderliches Schlüsselwort zur Eingabe von <Threshold>

<Threshold>: Intensitätsschwellenwert des analogen Eingangssignals, in V, Format: double

"S": erforderliches Schlüsselwort zur Eingabe von <ScanLineDistance>

<ScanLineDistance>: Abstand zwischen zwei Scanlinien, Format: double

"A": erforderliches Schlüsselwort zur Eingabe von <AnalogInputID>

<AnalogInputID>: ist die Kennung des analogen Eingangssignals, dessen Intensitätsmaximum gesucht wird, Format: Integer

Antwort: Keine

Hinweise: Die physikalische Einheit, in der <Distance1>, <Distance2> und <ScanLineDistance> anzugeben sind, kann mit PUN? (S. 253) abgefragt werden.

Die Werte für <Distance1> und <Distance2> müssen identisch sein.

Folgende Standardwerte werden verwendet, wenn die entsprechenden Argumente weggelassen werden:
<Threshold>: 1,0 V
<ScanLineDistance>: 0,01
<AnalogInputID>: 1

Diese Standardwerte wurden ausschließlich für die Hexapod-Modelle H-810, H-811 und H-206 geprüft. Für andere Hexapod-Modelle müssen die idealen Parameter experimentell ermittelt werden.

Bei Intensitätsverteilungen mit Nebenmaxima ist FSM mit anschließendem AAP (S. 158) gegenüber FSC (S. 190) bzw. FSA (S. 187) zu bevorzugen.

Je kleiner die Systemgeschwindigkeit mit VLS (S. 279) während einer Scanprozedur gesetzt ist, desto größer ist die Genauigkeit, mit der das Intensitätsmaximum gefunden wird.
Geschwindigkeiten im Bereich von unter 1 mm/s werden empfohlen.

Bei zu großen Werten für Strecken oder Winkel bewegt sich die Plattform während einer Scanprozedur auf einer undefinierten Bahn und kann verkippen. Dadurch sind Kollisionen und ein unbefriedigendes Ergebnis der Scanprozedur möglich. Maßnahmen zur Vermeidung des Verkippens:

- Wählen Sie für <Distance1> und <Distance2> passende Werte. Für die Hexapod-Modelle H-810, H-811 und H-206 sollten 0,2 mm bzw. 0,2 Grad nicht überschritten werden; für andere Hexapod-Modelle sowie bei geänderten Einstellungen für Koordinatensysteme und Pivotpunkt müssen die idealen Werte experimentell ermittelt werden.
- Stellen Sie die Geschwindigkeit für die Bewegungsplattform des Hexapods so gering wie möglich ein (mit dem Befehl VLS).
- Richten Sie die Bewegungsplattform vor der Scanprozedur schon passend aus.
- Verwenden Sie passende Halterungen für die auszurichtenden Ein- und/oder Ausgänge des optischen Elements auf der Bewegungsplattform, so dass die Bewegung während der Scanprozedur nur über kleine Strecken oder Winkel erfolgt.

Wenn der Parameter ***Trajectory Source*** (ID 0x19001900) auf 1 gesetzt ist, muss das Dynamikprofil durch aufeinander folgende MOV-Befehle vorgegeben werden, und FSM ist nicht zulässig.

Weitere Informationen siehe "Bewegungen des Hexapods" (S. 34).

Beispiel: FSM Y 0.2 Z 0.2 L 1 S 0.05 A 2

Startet eine Scanprozedur in der YZ-Ebene:

- Seitenlänge der quadratischen Fläche: 0,2 mm
- Intensitätsschwellenwert: 1 V
- Abstand zwischen den (in der Z-Achse liegenden) Scanlinien: 50 µm
- Kennung des analogen Eingangskanals, dessen Intensitätsmaximum gesucht wird: 2

FSS? (Get Status of Fast Scan Routines)

Beschreibung: Fragt den Status der zuletzt gestarteten Scanprozedur ab.

Um zu prüfen, ob eine Scanprozedur noch andauert, kann mit #5 (S. 153) der Bewegungsstatus der Achsen abgefragt werden.

Format: FSS?

Argumente: Keine

Antwort: <uint> gibt den Status der zuletzt gestarteten Scanprozedur an:
1: Scanprozedur wurde erfolgreich beendet
0: Scanprozedur läuft noch oder wurde erfolglos beendet.

Für Details zur erfolgreichen oder erfolglosen Beendigung siehe die Beschreibungen der Scanprozeduren.

Hinweis: FSS? erfragt den Status von Scanprozeduren, die mit folgenden Befehlen gestartet werden:
AAP (S. 158), FIO (S. 177), FLM (S. 180), FLS (S. 182), FSA (S. 187), FSC (S. 190), FSM (S. 195)

Beispiel: Senden: AAP Y 0.1 Z 0.1 SA 0.001 N 3 A 2
Senden: FSS?
Empfangen: 0
Hinweis: Die Scanprozedur läuft noch oder wurde nicht erfolgreich beendet.
Senden: FSS?
Empfangen: 1
Hinweis: Die mit AAP gestartete Scanprozedur wurde erfolgreich beendet, d. h. eine festgelegte Anzahl von Überprüfungen (Anzahl ≠ null) hat bestätigt, dass an der aktuellen Position das lokale Intensitätsmaximum vorliegt.

GWD? (Get Wave Table Data)

Beschreibung: Fragt die Kurvenform für die angegebene Kurventabelle ab.

Format: GWD? [<StartPoint> <NumberOfPoints> [{<WaveTableID>}]]

Argumente: <StartPoint> ist der Startpunkt in der Kurventabelle, beginnt mit Index 1

<NumberOfPoints> bezeichnet die Anzahl der je Tabelle zu lesenden Punkte

<WaveTableID> bezeichnet eine Kurventabelle des Controllers

Antwort: Der Inhalt der Kurventabelle (Kurvenform) im GCS-Array-Format (siehe separates Handbuch für GCS Array, SM 146E und untenstehendes Beispiel.)

Hinweise: Abhängig von der Kurvenformdefinition mit WAV (S. 282) können die Kurventabellen unterschiedliche Längen haben.

- Fragen Sie nur Kurventabellen gleicher Länge im selben Befehl ab.
- Fragen Sie den Inhalt von Kurventabellen mit unterschiedlicher Länge mit separaten Befehlen ab.

Wenn Kurventabellen unterschiedlicher Länge gemeinsam abgefragt werden, enthält die Antwort maximal so viele Punkte wie die kürzeste Kurventabelle, oder die Abfrage erzeugt einen Fehler.

HDI? (Get Help For Interpretation Of DIA? Response)

Beschreibung: Zeigt Beschreibungen und physikalische Einheiten für die Messgrößen an, die mit dem Befehl DIA? (S. 165) abgefragt werden können.

Format: HDI?

Argumente: Keine

Antwort {<MeasureID>"="<Description>TAB<PhysUnit> LF}

wobei

<MeasureID> die Kennung der Messgröße ist

<Description> der Name der Messgröße ist

<PhysUnit> die physikalische Einheit der Messgröße ist.

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

Hinweis: Die HDR-Antwort listet alle für den C-887 möglichen Aufzeichnungsoptionen auf.

Beispiel:

```
hdr?
#RecordOptions
0=Nothing is recorded
1=Commanded position of axis
2=Real position of axis
3=Position error of axis
8=Measurement time
17=Input value of channel, calculated in Volt
18=Input value of channel, directly from
channel, without dimension
150=Input value of channel, calculated in Volt,
input calculated made with SIC
70=Commanded velocity of axis
71=Commanded acceleration of axis
73=Output of axis
74=Current proportion to the current error of
axis
75=Current integrated position error of axis
76=Current derivative position error of axis
80=Status register of axis
86=Fifo Size for continuous position mode
84=Real position of second sensor in axis
87=Commanded position for continuous position
mode
#TriggerOptions
0=No trigger. Exception: STE and IMP always
trigger data recorder
1=Trigger with next command that changes the
position, default setting
2=Trigger with next command, resets trigger
settings to 0
4=Trigger immediately, resets trigger settings
to 0
6=Trigger with next command that changes the
position, resets trigger settings to 0
#Parameters to be set with SPA
```

```
0x16000000=Data Recorder   Data Recorder Table
Rate
0x16000201=Data Recorder   Data Recorder Points
Per Table
#Additional information
Set data recorder configuration with DRC
Get data recorder configuration with DRC?
Get number of recorded points with DRL?
Get recorded data values with DRR?
Set data recorder trigger source with DRT
Get data recorder trigger source with DRT?
Set record table rate (in cycles) with RTR
Get record table rate (in cycles) with RTR?
Tell number of data recorder tables with TNR?
#Sources for Record Options
0 = X Y Z U V W 1 2 3 4 5 6
1 = X Y Z U V W 1 2 3 4 5 6
2 = X Y Z U V W 1 2 3 4 5 6
3 = 1 2 3 4 5 6
8 = X Y Z U V W 1 2 3 4 5 6
17 = 1 2 3 4
18 = 1 2 3 4
150 = 1 2 3 4
70 = 1 2 3 4 5 6
71 = 1 2 3 4 5 6
73 = 1 2 3 4 5 6
74 = 1 2 3 4 5 6
76 = 1 2 3 4 5 6
75 = 1 2 3 4 5 6
80 = 1 2 3 4 5 6
84 = 1 2 3 4 5 6
87 = 1 2 3 4 5 6
end of help
```

HIB? (Get State Of HID Button)

Beschreibung: Fragt den aktuellen Status der angegebenen Taste des angegebenen HID ab.

Format: HIB? [{<HIDeviceID> <HIDeviceButton>}]

Argumente: <HIDeviceID> ist ein HID, das an den Controller angeschlossen ist; nähere Angaben siehe unten.

<HIDeviceButton> ist eine Taste des HID; nähere Angaben siehe unten.

Antwort: {<HIDeviceID> <HIDeviceButton> "="<HIDButtonState>}

wobei

<HIDButtonState> den Status der Taste als ganzzahligen Wert angibt:
Die möglichen Werte hängen vom Tastentyp ab. Der Wertebereich für die einzelnen Tasten kann mit dem Befehl HIS? abgefragt werden. Wenn nur die Werte 0 und 1 zugelassen sind, haben sie folgende Bedeutung:
0 = Taste nicht gedrückt, 1 = Taste gedrückt
Die Bedeutung von Werten > 1 hängt vom HID ab.

Hinweise: Der C-887 unterstützt ein HID-Gerät (Kennung: 1) und zwei Tasten dieses HID-Geräts (Kennungen: 1 und 2).

Das HID-Gerät kann an einer der beiden USB-Schnittstellen des C-887 angeschlossen werden.

Wenn eine Bedieneinheit C-887.MC2 oder C-887.MC angeschlossen ist, kann mit HIB? der Status der beiden Drucktasten der Bedieneinheit abgefragt werden. Der Wert des Parameters ***HID Device Button Mode*** (ID 0x0E001600) bestimmt das Verhalten der Drucktasten:

- Parameterwert ist 0 (Standard): Die Drucktasten lösen Aktionen aus (Stop, Referenzierungsfahrt) gemäß den Beschreibungen in der Dokumentation der Bedieneinheit.
- Parameterwert ist 1: Die Drucktasten lösen keine Aktionen aus.

Der Wert des Parameters kann mit dem Befehl SPA (S. 258) geändert werden.

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

Hinweis: Die Antwort auf HLP? enthält die in der aktuellen Befehlsebene zugelassenen Befehle. Weitere Informationen siehe CCL (S. 162).

HLT (Halt Motion Smoothly)

Beschreibung: Stoppt die Bewegung der angegebenen Achsen sanft. Nähere Angaben siehe Hinweise unten.

Fehlercode 10 wird gesetzt.

#24 (S. 156) und STP (S. 267) 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 alle Achsenbewegungen, die durch Bewegungsbefehle, Fast-Alignment-Routinen, Scanprozeduren oder Funktionsgeneratorausgabe verursacht werden, und die Referenzierungsfahrt.

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

HLT stoppt Makros **nicht**.

HPA? (Get List Of Available Parameters)

Beschreibung: Antwortet mit einem Hilfe-String, der alle verfügbaren Parameter mit Kurzbeschreibungen enthält. Weitere Informationen siehe "Parameterübersicht" (S. 320).

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.

Hinweise: Der String hat folgendes Format:

<CmdLevel>TAB<MaxItem>TAB<DataType>TAB<FunctionGroupDescription>TAB<ParameterDescription>[{TAB<PossibleValue>"="<ValueDescription>}]

wobei

<CmdLevel> ist die Befehlsebene, die Schreibzugriff auf den Parameterwert erlaubt

<MaxItem> ist die maximale Anzahl der Elemente desselben Typs, auf die sich der Parameter auswirkt. (Die Bedeutung von „Element" hängt vom Parameter ab; es kann sich um eine Achse, ein Hexapod-Bein, einen Eingangssignalkanal oder das gesamte System handeln.)

<DataType> ist der Datentyp des Parameterwertes, er kann INT, FLOAT oder CHAR sein

<FunctionGroupDescription> ist der Name der Funktionsgruppe, der der Parameter angehört. (Parameter werden entsprechend ihrem Zweck zu Gruppen zusammengefasst, um ihre Beziehung zueinander zu verdeutlichen.)

<ParameterDescription> ist der Name des Parameters

<PossibleValue> ist ein Wert aus dem zulässigen Datenbereich

<ValueDescription> ist die Bedeutung des entsprechenden Wertes

Die Auflistung der Parameter variiert in Abhängigkeit vom installierten optionalen Zubehör. Die Auflistung kann sich außerdem zur Laufzeit des C-887 ändern, wenn die Zuweisung von Positionierertypen zu den Achsen A und B geändert wird, siehe auch "Betriebsparameter der Achsen A und B" (S. 32).

Für das Abfragen und Ändern von Parameterwerten siehe "Anpassen von Einstellungen" (S. 317).

IFC? (Get Current Interface Parameters)

Beschreibung: Fragt die Werte der Schnittstellenparameter für die Kommunikation aus dem flüchtigen Speicher ab.

Format:	IFC? [{<InterfacePam>}]
Argumente:	<InterfacePam> ist der abzufragende Schnittstellenparameter, mögliche Werte siehe unten.
Antwort:	{<InterfacePam>"="<PamValue> LF} wobei <PamValue> den Wert des Schnittstellenparameters aus dem flüchtigen Speicher angibt. <InterfacePam> kann RSPORT, RSBAUD, RSHSHK, IPADR, IPSTART, IPMASK, IPMAXCONN, MACADR oder TERMSTR sein. Für <InterfacePam> = RSPORT gibt <PamValue> den für die RS-232-Kommunikation genutzten Port aus: 1 Für <InterfacePam> = RSBAUD gibt <PamValue> die für die RS-232-Kommunikation verwendete Baudrate an. Für <InterfacePam> = RSHSHK gibt <PamValue> die Handshake-Einstellung für die RS-232-Kommunikation aus: 1 = RTS/CTS Für <InterfacePam> = IPSTART gibt <PamValue> die aktuelle Einstellung des Startverhaltens für die Konfiguration der IP-Adresse für die TCP/IP-Kommunikation an, 0 = Die mit IPADR definierte IP-Adresse wird verwendet. 1 = DHCP wird verwendet, um die IP-Adresse zu erhalten; Für <InterfacePam> = IPADR geben die ersten vier Teile von <PamValue> die IP-Adresse an, die aktuell für die TCP/IP-Kommunikation verwendet wird, der letzte Teil gibt den Port an; Für <InterfacePam> = IPMASK gibt <PamValue> die IP-Masken-Einstellung, die aktuell für die TCP/IP-Kommunikation verwendet wird, in der Form uint.uint.uint.uint an; Für <InterfacePam> = IPMAXCONN gibt <PamValue> die maximal zulässige Anzahl von Verbindungen für die TCP/IP-Kommunikation an; Für <InterfacePam> = MACADR gibt <PamValue> die unveränderliche, eindeutige Adresse der Netzwerkhardware im C-887 an; Für <InterfacePam> = TERMSTR gibt <PamValue> das Abschlusszeichen für die Befehle des GCS aus: 0 = LineFeed (ASCII-Zeichen 10)

IFS (Set Interface Parameters as Default Values)

Beschreibung: Speichert Schnittstellenparameter.

Ändert die Standardparameter für die Schnittstelle im permanenten Speicher, aber nicht die aktuell aktiven Parameter. Die mit IFS vorgenommenen Einstellungen werden beim nächsten Einschalten oder Neustart aktiviert.

Format: IFS <Pswd> {<InterfacePam> <PamValue>}

Argumente: <Pswd> ist das Kennwort zum Schreiben in den permanenten Speicher, Standardwert ist "100"

<InterfacePam> ist der zu ändernde Schnittstellenparameter, siehe unten

<PamValue> gibt den Wert des Schnittstellenparameters an, siehe unten

Antwort: Keine

Folgende Schnittstellenparameter können gesetzt werden:

RSBAUD
<PamValue> gibt die für die RS-232-Kommunikation zu verwendende Baudrate an. Mögliche Werte sind:
9600, 19200, 38400, 57600 und 115200. Standard ist 115200

IPADR
Die ersten vier Teile von <PamValue> geben die Standard-IP-Adresse für die TCP/IP-Kommunikation an, der letzte Teil gibt den zu verwendenden Standardport an, Standard ist 192.168.1.28:50000;
Hinweis: Während die IP-Adresse geändert werden kann, muss der Port stets 50000 sein!

IPSTART
<PamValue> definiert das Startverhalten zur Konfiguration der IP-Adresse für die TCP/IP-Kommunikation,
0 = Die mit IPADR definierte IP-Adresse wird verwendet
1 = DHCP oder AutoIP wird verwendet, um die IP-Adresse zu erhalten. (Standard);

IPMASK
<PamValue> gibt die für die TCP/IP-Kommunikation zu verwendende Subnetzmaske in der Form uint.uint.uint.uint an, Standard ist 255.255.255.0;

Antwort: Keine

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

Wenn IPSTART = 1 gesetzt ist, verwendet der C-887 die AutoIP-Funktionalität für die TCP/IP-Kommunikation in einem Netzwerk ohne DHCP-Server oder direkt mit dem PC. Mit AutoIP konfigurieren die Netzwerkteilnehmer automatisch ihre Schnittstellen, so dass die manuelle Abstimmung der IP-Adressen entfällt.

Weitere Schnittstellenparameter des C-887 sind schreibgeschützt. Die Standardeinstellungen dieser Parameter können mit IFS? (S. 208) abgefragt werden.

Weitere Informationen finden Sie in "Kommunikation über TCP/IP-Schnittstelle herstellen" (S. 74).

IFS? (Get Interface Parameters as Default Values)

Beschreibung: Fragt die im permanenten Speicher gespeicherten Parameterwerte der Schnittstellenkonfiguration ab (d. h. Standardeinstellungen)

Format: IFS? [{<InterfacePam>}]

Argumente: <InterfacePam> ist der abzufragende Schnittstellenparameter. Mögliche Werte siehe unten.

Antwort: {<InterfacePam>"="<PamValue> LF}

wobei

<PamValue> der Wert des Schnittstellenparameters im permanenten Speicher ist.

<InterfacePam> kann RSPORT, RSBAUD, RSHSHK, IPADR, IPSTART, IPMASK, IPMAXCONN, MACADR und TERMSTR sein

Die folgenden Schnittstellenparameter sind schreibgeschützt:

Für <InterfacePam> = RSPORT gibt <PamValue> den für die RS-232-Kommunikation genutzten Port aus:
1

Für <InterfacePam> = RSHSHK gibt <PamValue> die Handshake-Einstellung für die RS-232-Kommunikation aus:
1 = RTS/CTS

Für <InterfacePam> = IPMAXCONN gibt <PamValue> die maximal zulässige Anzahl von Verbindungen für die TCP/IP-Kommunikation aus:
1

Für <InterfacePam> = MACADR gibt <PamValue> die eindeutige Adresse der Netzwerkhardware im C-887 aus.

Für <InterfacePam> = TERMSTR gibt <PamValue> das Abschlusszeichen für die Befehle des GCS aus:
0 = LineFeed (ASCII-Zeichen 10)

Für alle weiteren Ausprägungen von <InterfacePam> siehe IFS (S. 207).

IMP (Start Impulse and Response Measurement)

Beschreibung: Startet einen Impuls und zeichnet die Impulsantwort für die angegebene Achse auf.

Die Datenrekorderkonfiguration, d. h. die Zuweisung der Datenquellen und der Aufzeichnungsoptionen zu den Rekordertabellen, kann mit DRC (S. 168) gesetzt werden.

Die aufgezeichneten Daten können mit dem Befehl DRR? (S. 172) gelesen werden.

Format: IMP <AxisID> <Amplitude>

Argumente: <AxisID> ist eine Achse des Controllers

<Amplitude> ist die Höhe des Impulses. Nähere Angaben siehe unten.

Antwort: Keine

Fehlersuche: Die Zielposition, die aus der angegebenen Impulshöhe resultiert, ist außerhalb der Grenzwerte.

Hinweise: Ein „Impuls" besteht aus einer relativen Bewegung mit der angegebenen Amplitude, gefolgt von einer gleich großen Bewegung in die entgegengesetzte Richtung. Der Impuls wird relativ zu der aktuellen Position ausgeführt.

Die Pulsweite des Impulses resultiert aus dem Wert des Parameters ***Pulse Length Factor*** (ID 0x0E000900) multipliziert mit der achsenabhängigen Zykluszeit (S. 376).

Die physikalische Einheit, in der <Amplitude> anzugeben ist, kann mit PUN? (S. 253) abgefragt werden.

Für die Achsen des Hexapods (X, Y, Z, U, V, W) gilt Folgendes:

- Vor dem Start jeder Bewegung erfolgt eine Prüfung, ob die Bewegungsplattform die kommandierte Zielposition tatsächlich erreichen kann. Mit VMO? (S. 280) können Sie abfragen, ob die Zielposition erreicht werden kann.
- Wenn der Parameter ***Trajectory Source*** (ID 0x19001900) auf 1 gesetzt ist, muss das Dynamikprofil durch aufeinander folgende MOV-Befehle vorgegeben werden. IMP ist nicht zulässig.

Weitere Informationen siehe "Bewegungen des Hexapods" (S. 34).

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. 281) 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

KCP (Copy Coordinate System)

Beschreibung: Erzeugt eine Kopie eines Koordinatensystems.

Format: KCP <CSNameSource> <CSNameCopy>

Argumente: <CSNameSource> ist der Name des Koordinatensystems, von dem eine Kopie erzeugt werden soll. PI_Base, PI_Levelling, ZERO und HEXAPOD können nicht kopiert werden.

<CSNameCopy> ist der Name der Kopie des Koordinatensystems.

Antwort: Keine

Hinweise: PI_Base, PI_Levelling, ZERO und HEXAPOD können nicht kopiert werden.

Möglichkeiten zum Anlegen der Kopie:

- <CSNameCopy> ist ein neuer Name. Die Kopie wird unter diesem Namen als neues Koordinatensystem angelegt.
- <CSNameCopy> ist der Name eines existierenden Koordinatensystems, das nicht verwendet wird. Das Koordinatensystem wird dadurch überschrieben.

Die Verkettung mit dem Vorgänger in einer Kette von Koordinatensystemen wird kopiert. Die Verkettung mit Nachfolgern wird nicht kopiert.

Die Kopie wird im flüchtigen Speicher erzeugt. Mit `WPA SKS` (S. 293) kann die Kopie in den permanenten Speicher geschrieben werden.

KEN (Activate Coordinate System)

Beschreibung: Aktiviert das angegebene Koordinatensystem. Der Umfang der Einstellungen, die durch das Aktivieren beeinflusst werden, hängt vom Typ des Koordinatensystems ab, siehe unten.

Positionsangaben für die Bewegungsplattform des Hexapods (Abfrage mit POS? (S. 252)) beziehen sich auf das aktive Betriebs-Koordinatensystem.

Wenn mit dem Work-und-Tool-Konzept gearbeitet wird:

- Das Work-und-Tool-Konzept verwendet eine Kombination aus zwei aktiven Betriebs-Koordinatensystemen. In der Regel besteht die Kombination aus jeweils einem aktiven Koordinatensystem der Typen KST und KSW. Wenn nur für einen der beiden Typen ein Koordinatensystem aktiv ist, wird für den anderen Typ automatisch ein Ersatz verwendet.
- Die mit POS? abgefragte aktuelle Position der bewegten Plattform des Hexapods ist als Position des Tool-Koordinatensystems im Work-Koordinatensystem zu verstehen.

Das Aktivieren von Koordinatensystemen mit KEN löst keine Bewegung aus.

Format: KEN <CSName>

Argumente: <CSName> ist der Name des Koordinatensystems, das aktiviert werden soll.

Antwort: Keine

Hinweise: Vor dem Aktivieren eines Koordinatensystems prüft KEN die korrekte Definition dieses Koordinatensystems inklusive seiner korrekten Verkettung. Wenn das Koordinatensystem nicht korrekt definiert ist, wird es nicht aktiviert.

Koordinatensysteme der folgenden Typen können mit KEN aktiviert werden: KSD, KSF, KSW, KST, KSB(USER), KSB(PI), KLF(USER), KLF(PI), KLD(USER), KLD(PI), ZERO

Aus den folgenden Gruppen von Koordinatensystemen ist jeweils genau ein Koordinatensystem bzw. genau eine Koordinatensystem-Kombination aktiv:

- Betriebs-Koordinatensystem: ein Koordinatensystem vom Typ ZERO oder KSF oder KSD oder - für das Work-und-Tool-Konzept - eine Kombination aus Koordinatensystemen der Typen KSW/KST oder ZERO/KST oder KSW/ZERO
- Achsorientierungs-Koordinatensystem (Typ KSB(PI) oder KSB(USER))
- Korrektur-Koordinatensystem (Typ KLD(USER) oder KLD(PI) oder KLF(USER) oder KLF(PI))

Das Aktivieren eines Koordinatensystems für eine der Gruppen deaktiviert gleichzeitig das Koordinatensystem bzw. die Koordinatensystem-Kombination, welche/s zuvor für diese Gruppe aktiv war.

Durch Senden von KEN ZERO wird das werkseitig aktivierte Betriebs-Koordinatensystem ZERO wieder aktiviert. Wenn die Befehlsebene 1 aktiv ist (siehe CCL (S. 162)), aktiviert KEN ZERO auch wieder das werkseitig aktivierte Korrektur-Koordinatensystem PI_Levelling, jedoch nicht das werkseitig aktivierte Achsorientierungs-Koordinatensystems PI_Base (dieses kann durch Senden von KEN PI_Base wieder aktiviert werden). Durch Senden von `DPA SKS` (S. 167) können alle werkseitig aktivierten Koordinatensysteme unabhängig von der aktuell aktiven Befehlsebene wieder aktiviert werden.

Durch das aktive Betriebs-Koordinatensystem werden Werte für die folgenden Einstellungen vorgegeben (beim Work-und-Tool-Konzept

durch die Kombination aus zwei Betriebs-Koordinatensystemen):

- NLM (S. 248): Untergrenze des Verfahrbereichs der Achse
- PLM (S. 251): Obergrenze des Verfahrbereichs der Achse
- SSL (S. 263): Aktivierungszustand der Verfahrbereichsgrenzen der Achse
- SPI (S. 260): Koordinaten des Pivotpunkts (nur für Koordinatensysteme der Typen KSF und ZERO)
- Wenn vom Controller unterstützt:
  SST (S. 264): Schrittweite für Bewegungen, die durch eine manuelle Bedieneinheit ausgelöst werden

Vor dem Aktivieren von Korrektur- und Achsorientierungs-Koordinatensystemen (Typen KLD(), KLF() und KSB()) ist der Wechsel auf Befehlsebene 1 erforderlich (siehe CCL).

Die Koordinatensysteme werden im flüchtigen Speicher aktiviert und deaktiviert. Mit `WPA SKS` (S. 293) kann der Aktivierungszustand in den permanenten Speicher geschrieben werden.

KEN? (Get Active Coordinate System)

Beschreibung: Listet die Namen der aktiven Koordinatensysteme auf und zeigt deren Typ an.

Format: KEN? [{<CSName>}]

Argumente: <CSName> ist der Name eines aktiven Koordinatensystems. Nicht zulässig: ZERO.

Wenn <CSName> weggelassen wird, werden alle aktiven Koordinatensysteme aufgelistet.

Antwort: {<CSName>"="<CSType>}

wobei

<CSType> den Typ des Koordinatensystems angibt.

Hinweise: KEN? fragt den flüchtigen Speicher ab.

Wenn das Betriebs-Koordinatensystem ZERO aktiv ist, wird es in der Antwort auf KEN? **nicht** angezeigt, und die Antwort enthält nur Folgendes:

- Das aktive Korrektur-Koordinatensystem, d.h. ein Koordinatensystem vom Typ KLD(PI) oder KLD(USER) oder KLF(PI) oder KLF(USER)

- Das aktive Achsorientierungs-Koordinatensystem, d.h. ein Koordinatensystem vom Typ KSB(PI) oder KSB(USER)

Wenn <CSName> in der Abfrage angegeben wird und das entsprechende Koordinatensystem nicht aktiv ist, sendet der C-887 eine leere Antwort und setzt einen Fehler (Abfrage des Fehlercodes mit ERR? (S. 176)).

KET? (Get Active Coordinate System Types)

Beschreibung: Listet die aktiven Typen von Koordinatensystemen auf und zeigt die Namen der entsprechenden Koordinatensysteme an.

Format: KET? [{<CSType>}]

Argumente: <CSType> ist ein aktiver Koordinatensystem-Typ. Mögliche Werte: KSW, KST, KSF, KSD, KLD(PI), KLD(USER), KLF(PI), KLF(USER), KSB(PI), KSB(USER)

Wenn <CSType> weggelassen wird, werden alle aktiven Koordinatensystem-Typen aufgelistet.

Antwort: {<CSType>"="<CSName>}

wobei

<CSName> den Namen des Koordinatensystems angibt.

Hinweise: KET? fragt den flüchtigen Speicher ab.

Wenn das Betriebs-Koordinatensystem vom Typ ZERO aktiv ist, wird dieser Typ in der Antwort auf KET? **nicht** angezeigt, und die Antwort enthält nur Folgendes:

- Den Typ des aktiven Korrektur-Koordinatensystems, d.h. KLD(PI) oder KLD(USER) oder KLF(PI) oder KLF(USER)
- Den Typ des aktiven Achsorientierungs-Koordinatensystems, d. h. KSB(PI) oder KSB(USER)

Wenn <CSType> in der Abfrage angegeben wird und kein Koordinatensystem des entsprechenden Typs aktiv ist, sendet der C-887 eine leere Antwort und setzt einen Fehler (Abfrage des Fehlercodes mit ERR? (S. 176)).

KLC? (Get Properties Of Work-And-Tool Combinations)

Beschreibung: Listet die Eigenschaften der im flüchtigen Speicher vorhandenen Kombinationen von Koordinatensystemen für das Work-und-Tool-Konzept auf.

Das Work-und-Tool-Konzept verwendet eine Kombination aus zwei aktiven Betriebs-Koordinatensystemen. In der Regel besteht die Kombination aus jeweils einem aktiven Koordinatensystem der Typen KST und KSW. Wenn nur für einen der beiden Typen ein Koordinatensystem aktiv ist, wird für den anderen Typ automatisch ein Ersatz verwendet. Ersatzweise verwendete Koordinatensysteme sind unter dem Namen "Zero" in der Antwort auf KLC? enthalten.

Eine Kombination wird im flüchtigen Speicher beim Aktivieren eines Koordinatensystems vom Typ KST oder KSW angelegt. Die Kombinationen sind im flüchtigen Speicher auch dann noch vorhanden, wenn die darin enthaltenen Koordinatensysteme vom Typ KST oder KSW nicht mehr aktiv sind.

Wenn mit KRM (S. 227) ein Koordinatensystem des Typs KST oder KSW gelöscht wird, sind die Kombinationen, in denen dieses Koordinatensystem enthalten war, nicht mehr in der Antwort auf KLC? enthalten.

Mit `WPA SKS` (S. 293) können die im flüchtigen Speicher vorhandenen Kombinationen in den permanenten Speicher geschrieben werden.

Die Eigenschaften der im flüchtigen Speicher vorhandenen Koordinatensysteme können mit KLS? (S. 223) abgefragt werden.

Format: KLC? [<CSName1>[<CSName2>[<Item1>[<Item2>]]]]

Argumente: <CSName1> ist der Name eines Koordinatensystems vom Typ KST oder KSW, das Bestandteil einer Kombination im flüchtigen Speicher ist.

<CSName2> ist der Name eines Koordinatensystems vom Typ KST oder KSW, das Bestandteil einer Kombination im flüchtigen Speicher ist.

<Item1> ist eine Eigenschaft der Achsen des Controllers für die abgefragte Kombination von Koordinatensystemen. Mögliche Werte:

- NLM: Untergrenze des Verfahrbereichs der Achse
- PLM: Obergrenze des Verfahrbereichs der Achse
- SSL: Aktivierungszustand der Verfahrbereichsgrenzen der Achse
- Wenn vom Controller unterstützt:
  SST: Schrittweite für Bewegungen, die durch eine manuelle Bedieneinheit ausgelöst werden

Die Einstellungen der Eigenschaften für die aktuell aktive Kombination können mit den entsprechenden Befehlen geändert und mit WPA gespeichert werden.

<Item2> ist eine Achse des Controllers, mögliche Werte: X, Y, Z, U, V, W

Wenn die Eigenschaften aller Kombinationen von Koordinatensystemen aufgelistet werden sollen, werden alle Argumente weggelassen.

Antwort: <String>

<String> enthält im XML-Format Informationen zu den im flüchtigen Speicher vorhandenen Kombinationen von Koordinatensystemen.

Der Aufbau der Antwort hängt von der Anzahl der Argumente im gesendeten Befehl ab. Mögliche Antworten je nach Anzahl der Argumente am Beispiel der Koordinatensysteme Node1, Node2, Node3 und Node4:

Gesendet:
```
KLC?
```
Antwort:
```
<CombinedCoordinateSystem>[SP][LF]
[TAB] <NODE1.NODE2 Name="NODE1.NODE2"
Work="NODE1" Tool="NODE2">[SP][LF]
[TAB] [TAB] <NLM X="-3.0" … W="-5.0"/>[SP][LF]
[TAB] [TAB] <PLM X="3.0" … W="5.0"/>[SP][LF]
[TAB] [TAB] <SSL X="1" Y="0" … W="1"/> [SP][LF]
[TAB] [TAB] <SST X="0.1" … W="0.2"/>[SP][LF]
[TAB] </NODE1.NODE2>[SP][LF]
[TAB] [TAB] <NODE1.NODE4  Name="NODE1.NODE4"
Work="NODE1" Tool="NODE4"> [SP][LF]
[TAB] [TAB] <NLM X="-4.0" … W="-3.4"/>[SP][LF]
[TAB] [TAB] <PLM X="2.0" … W="2.0"/>[SP][LF]
[TAB] [TAB] <SSL X="1" Y="0" … W="1"/> [SP][LF]
[TAB] [TAB] <SST X="0.2" … W="0.15"/>[SP][LF]
[TAB] </NODE1.NODE4>[SP][LF]
…
</CombinedCoordinateSystem>[LF]
```

Gesendet:
```
KLC? Node1
```
Antwort:
```
<CombinedCoordinateSystem>[SP][LF]
[TAB] <NODE1.NODE2 Name="NODE1.NODE2"
```

```
Work="NODE1" Tool="NODE2"> [SP][LF]
[TAB] [TAB] <NLM X="-3.0" … W="-5.0"/>[SP][LF]
[TAB] [TAB] <PLM X="3.0"  … W="5.0"/>[SP][LF]
[TAB] [TAB] <SSL X="1" Y="0" … W="1"/> [SP][LF]
[TAB] [TAB] <SST X="0.1" … W="0.2"/>[SP][LF]
[TAB] </NODE1.NODE2 >[SP][LF]
[TAB] <NODE1.NODE4 Name="NODE1.NODE4"
Work="NODE1" Tool="NODE4"> [SP][LF]
[TAB] [TAB] <NLM X="-1.0" … W="-7.0"/>[SP][LF]
[TAB] [TAB] <PLM X="1.1"  … W="10.0"/>[SP][LF]
[TAB] [TAB] <SSL X="1" Y="0" … W="1"/> [SP][LF]
[TAB] [TAB] <SST X="0.2" … W="0.21"/>[SP][LF]
[TAB] </NODE1.NODE4 >[SP][LF]
…
</CombinedCoordinateSystem>[LF]
```

Gesendet:

```
KLC? Node1 Node2
```

Antwort:

```
<CombinedCoordinateSystem>[SP][LF]
[TAB] <NODE1.NODE2 Name="NODE1.NODE2"
Work="NODE1" Tool="NODE2"> [SP][LF]
[TAB] [TAB] <NLM X="-3.0" … W="-5.0"/>[SP][LF]
[TAB] [TAB] <PLM X="3.0" …  W="5.0"/>[SP][LF]
[TAB] [TAB] <<SSL X="1" Y="0" … W="1"/> [SP][LF]
[TAB] [TAB] <SST X="0.1" … W="0.2"/>[SP][LF]
[TAB] </NODE1.NODE2 >[SP][LF]
</CombinedCoordinateSystem>[LF]
```

Gesendet:

```
KLC? Node1 Node2 PLM
```

Antwort:

```
<PLM X="3.0" … W="5.0"/>[LF]
```

Gesendet:

```
KLC? Node1 Node2 PLM X
```

Antwort:

```
X = 3.0 [LF]
```

KLD (Define Leveling Coordinate System By Specifying Values)

Beschreibung: Definiert ein Korrektur-Koordinatensystem vom Typ KLD(USER) zur dauerhaften Korrektur von Fehlern in der Ausrichtung des Hexapods (z. B. Einbaufehler).

Das Korrektur-Koordinatensystem wird auf Basis von Messwerten definiert (z. B. unter Verwendung eines Interferometers) und korrigiert die Linearverschiebung (Achsen X, Y, Z) und Achsneigung (Achsen U, V, W) der Bewegungsplattform des Hexapods.

Wenn die Linearverschiebung und Achsneigung nicht gemessen werden kann: Verwenden Sie KLF (S. 220) zum Definieren eines Korrektur-Koordinatensystems.

Das Koordinatensystem wird im flüchtigen Speicher definiert. Mit `WPA SKS` (S. 293) kann die Definition in den permanenten Speicher geschrieben werden.

Format: KLD <CSName> [{<AxisID> <Offset>}]

Argumente: <CSName> ist der Name des Koordinatensystems, das definiert werden soll.

<AxisID> ist eine Achse des Controllers. Mögliche Werte: X, Y, Z, U, V, W.

<Offset> ist ein Offset, der nach der Referenzierungsfahrt zum aktuellen Positionswert der Achse addiert wird; in physikalischen Einheiten.

Für Achsen, die im Befehl KLD nicht angegeben werden, wird der Offset auf null gesetzt.

Antwort: Keine

Hinweise: Vor dem Definieren eines Korrektur-Koordinatensystems ist der Wechsel auf Befehlsebene 1 erforderlich (siehe CCL (S. 162)).

Möglichkeiten zum Definieren eines Koordinatensystems mit KLD:

- <CSName> ist ein neuer Name. Das Korrektur-Koordinatensystem wird unter diesem Namen neu angelegt.
- <CSName> ist der Name eines existierenden Koordinatensystems, das nicht verwendet wird. Das Koordinatensystem wird durch die Definition mit KLD überschrieben.

Empfohlene Vorgehensweise zum Definieren und Aktivieren eines Korrektur-Koordinatensystems vom Typ KLD(USER):

1. Ausführen der Referenzierungsfahrt (siehe FRF (S. 185))

2. Messen der Abweichung von Position und Orientierung der bewegten Plattform des Hexapods von derjenigen Position und Orientierung, in der künftig gelten soll X = 0, Y = 0, Z = 0, U = 0, V = 0, W = 0 (Messung mit externem Messinstrument)
3. Wechsel auf Befehlsebene 1 durch Senden von `CCL 1 advanced`
4. Definieren des Korrektur-Koordinatensystems mit KLD unter Angabe der gemessenen Abweichungen für die Achsen der Bewegungsplattform (Offsetwerte)
5. Aktivieren des Korrektur-Koordinatensystems (siehe KEN (S. 211))
6. Optional: Festlegung des Verhaltens nach der Referenzierungsfahrt durch Setzen der Parameter ***Behaviour After Reference Move*** (ID 0x07030401) und ***Target For Motion After Reference Move*** (ID 0x07030402). Auf diese Weise können die Achsen der Plattform z. B. nach der Referenzierungsfahrt automatisch zur Nullposition bewegt werden.
   - Wert des Parameters 0x07030401 = 0: Die Achse bleibt nach der Referenzierungsfahrt in der Referenzposition
   - Wert des Parameters 0x07030401 = 1: Die Achse fährt nach der Referenzierungsfahrt zur Zielposition, die durch Parameter 0x07030402 vorgegeben ist
7. Speichern der Einstellungen durch Senden von `WPA SKS`

Die Offsetwerte, die in der Antwort auf KLS? (S. 223) für Koordinatensysteme vom Typ KLD(USER) angezeigt werden, ergeben sich durch Rückrechnung aus allen aktuell aktiven Koordinatensystemen. Sie können sich deshalb beim Wechsel der aktiven Koordinatensysteme ändern. Die in der Antwort auf KLT? (S. 226) enthaltenen Offsetwerte sind dagegen jeweils auf den angegebenen Vorgänger in der Kette bezogen und deshalb unabhängig von den aktuell aktiven Koordinatensystemen.

Für Korrektur-Koordinatensysteme, die mit KLD definiert werden, gilt Folgendes:

- Das Korrektur-Koordinatensystem ist immer direkter Nachfolger des werkseitig eingestellten Korrektur-Koordinatensystems PI_Levelling (automatische Verkettung).
- Das Korrektur-Koordinatensystem kann **nicht** mit KLN (S. 221) mit anderen Koordinatensystemen verkettet werden

DPA SKS (S. 167) aktiviert unabhängig von der aktuell aktiven Befehlsebene wieder das werkseitig aktivierte Korrektur-Koordinatensystem PI_Levelling, Details siehe KEN (S. 211).

KLF (Define Leveling Coordinate System At Current Position)

Beschreibung: Definiert ein Korrektur-Koordinatensystem vom Typ KLF(USER) zur dauerhaften Korrektur von Fehlern in der Ausrichtung des Hexapods (z. B. Einbaufehler).

Zum Definieren des Korrektur-Koordinatensystems wird die Bewegungsplattform nach der Referenzierungsfahrt in die Position und Orientierung kommandiert, in der künftig gelten soll X = 0, Y = 0, Z = 0, U = 0, V = 0, W = 0. Durch Senden von KLF wird ein Koordinatensystem mit Offsetwerten definiert, die nach der Referenzierungsfahrt zu den aktuellen Positionswerten der Achsen addiert werden; in physikalischen Einheiten.

Wenn Linearverschiebung (Achsen X, Y, Z) und Achsneigung (Achsen U, V, W) gemessen werden sollen: Verwenden Sie KLD (S. 218) zum Definieren eines Korrektur-Koordinatensystems.

Das Koordinatensystem wird im flüchtigen Speicher definiert. Mit `WPA SKS` (S. 293) kann die Definition in den permanenten Speicher geschrieben werden.

Format: KLF <CSName>

Argumente: <CSName> ist der Name des Koordinatensystems, das definiert werden soll.

Antwort: Keine

Hinweise: Vor dem Definieren eines Korrektur-Koordinatensystems ist der Wechsel auf Befehlsebene 1 erforderlich (siehe CCL (S. 162)).

Möglichkeiten zum Definieren eines Koordinatensystems mit KLF:

- <CSName> ist ein neuer Name. Das Korrektur-Koordinatensystem wird unter diesem Namen neu angelegt.
- <CSName> ist der Name eines existierenden Koordinatensystems, das nicht verwendet wird. Das Koordinatensystem wird durch die Definition mit KLF überschrieben.

Die Definition mit KLF ist nur möglich, wenn sich der Hexapod nicht bewegt.

Empfohlene Vorgehensweise zum Definieren und Aktivieren eines Korrektur-Koordinatensystems vom Typ KLF(USER):

1. Ausführen der Referenzierungsfahrt (siehe FRF (S. 185))
2. Anfahren derjenigen Position und Orientierung der Bewegungsplattform des Hexapods, in der künftig gelten soll X = 0, Y = 0, Z = 0, U = 0, V = 0, W = 0
3. Wechsel auf Befehlsebene 1 durch Senden von `CCL 1 advanced`
4. Definieren des Korrektur-Koordinatensystems mit KLF

5. Aktivieren des Korrektur-Koordinatensystems (siehe KEN (S. 211))
6. Optional: Festlegung des Verhaltens nach der Referenzierungsfahrt durch Setzen der Parameter ***Behaviour After Reference Move*** (ID 0x07030401) und ***Target For Motion After Reference Move*** (ID 0x07030402). Auf diese Weise können die Achsen der Plattform z. B. nach der Referenzierungsfahrt automatisch zur Nullposition bewegt werden.
   - Wert des Parameters 0x07030401 = 0: Die Achse bleibt nach der Referenzierungsfahrt in der Referenzposition
   - Wert des Parameters 0x07030401 = 1: Die Achse fährt nach der Referenzierungsfahrt zur Zielposition, die durch Parameter 0x07030402 vorgegeben ist
7. Speichern der Einstellungen durch Senden von `WPA SKS`

Die Offsetwerte, die in der Antwort auf KLS? (S. 223) für Koordinatensysteme vom Typ KLF(USER) angezeigt werden, ergeben sich durch Rückrechnung aus allen aktuell aktiven Koordinatensystemen. Sie können sich deshalb beim Wechsel der aktiven Koordinatensysteme ändern. Die in der Antwort auf KLT? (S. 226) enthaltenen Offsetwerte sind dagegen jeweils auf den angegebenen Vorgänger in der Kette bezogen und deshalb unabhängig von den aktuell aktiven Koordinatensystemen.

Für Korrektur-Koordinatensysteme, die mit KLF definiert werden, gilt Folgendes:

- Das Korrektur-Koordinatensystem ist immer direkter Nachfolger des werkseitig eingestellten Korrektur-Koordinatensystems PI_Levelling (automatische Verkettung).
- Das Korrektur-Koordinatensystem kann **nicht** mit KLN (S. 221) mit anderen Koordinatensystemen verkettet werden

DPA SKS (S. 167) aktiviert unabhängig von der aktuell aktiven Befehlsebene wieder das werkseitig aktivierte Korrektur-Koordinatensystem PI_Levelling, Details siehe KEN (S. 211).

KLN (Link Coordinate Systems)

Beschreibung:	Verbindet zwei Koordinatensysteme zu einer Kette aus Vorgänger und Nachfolger.
Format:	KLN <ChildCS> <ParentCS>
Argumente:	<ChildCS> ist der Name des Koordinatensystems, das in der Kette als Nachfolger an <ParentCS> angehängt werden soll. <ParentCS> ist der Name des Koordinatensystems, das der Vorgänger von <ChildCS> in der Kette sein soll.

Antwort: Keine

Hinweise: Jedes Koordinatensystem ist Bestandteil mindestens einer Kette. Grundsätzlicher Aufbau von Koordinatensystem-Ketten siehe "Koordinatensysteme" (S. 41).

In der Werkseinstellung sind folgende Koordinatensysteme zu einer Kette verbunden:

- Das Koordinatensystem HEXAPOD, das auf der Konfigurationsdatei mit den Geometriedaten des Hexapods basiert, ist der "Ursprung" aller Ketten und Vorgänger des Korrektur-Koordinatensystems PI_Levelling (unveränderbare Verkettung)
- PI_Levelling ist Vorgänger des Achsorientierungs-Koordinatensystems PI_Base
- PI_Base ist Vorgänger des Betriebs-Koordinatensystems ZERO

Für verkettete Koordinatensysteme gilt:

- Die tatsächlichen Offsetwerte für die Position der Achsen X, Y, Z, U, V, W ergeben sich jeweils aus den Offsetwerten der mit einem Koordinatensystem verketteten Vorgänger (Details siehe KLT? (S. 226)).
- Jedes Koordinatensystem hat genau einen Vorgänger und kann einen oder mehrere Nachfolger haben.
- Wenn ein Koordinatensystem aktiv ist, sind alle Vorgänger in der Kette ebenfalls in Verwendung und können nicht gelöscht oder überschrieben werden.
- Wenn ein (nicht verwendetes) Koordinatensystem gelöscht wird, werden sein Vorgänger und sein Nachfolger in der Kette miteinander verbunden.

Beschränkungen für den Aufbau von Ketten mit KLN:

- Ein Koordinatensystem kann nicht mit sich selbst verkettet werden.
- Ringverbindungen aus mindestens zwei Koordinatensystemen können zwar gebildet, aber nicht mit KEN (S. 211) aktiviert werden.
- Ein Koordinatensystem, das verwendet wird, kann nicht mit KLN als Nachfolger an ein anderes Koordinatensystem angehängt werden.
- An ein Koordinatensystem, das verwendet wird, kann mit KLN ein (nicht verwendetes) Koordinatensystem als Nachfolger angehängt werden.
- Vor dem Verketten von Achsorientierungs-Koordinatensystemen vom Typ KSB(USER) als Nachfolger ist der Wechsel auf Befehlsebene 1 erforderlich (siehe CCL (S. 162)).
- Koordinatensysteme der Typen KLD(PI), KLF(PI), KLD(USER), KLF(USER) und das Koordinatensystem HEXAPOD können nicht mit KLN verkettet werden.
- Das Koordinatensystem PI_Base kann nicht mit KLN als Nachfolger

an ein anderes Koordinatensystem angehängt werden.

- Koordinatensysteme vom Typ KSB(USER) können mit KLN nur mit anderen Koordinatensystemen vom Typ KSB(USER) verkettet oder als Nachfolger an das Koordinatensystem PI_Base angehängt werden.
- Das Koordinatensystem ZERO kann nicht mit KLN als Nachfolger an ein anderes Koordinatensystem angehängt werden.

Koordinatensysteme werden mit KLN im flüchtigen Speicher verkettet. Mit WPA SKS (S. 293) kann die Verkettung in den permanenten Speicher geschrieben werden.

KLN? (Get Coordinate System Chains)

Beschreibung: Listet die Bestandteile der vorhandenen Ketten von Koordinatensystemen auf.

Jedes Koordinatensystem ist Bestandteil mindestens einer Kette. Grundsätzlicher Aufbau von Koordinatensystem-Ketten siehe "Koordinatensysteme" (S. 41).

Format: KLN? [{<CSName>}]

Argumente: <CSName> ist der Name eines Koordinatensystems, dessen Vorgänger in der Kette aufgelistet werden sollen.

Wenn die Vorgänger aller Koordinatensysteme aufgelistet werden sollen, wird <CSName> weggelassen.

Antwort: {<CSName>"="<String>}

wobei

<String> die Namen der Vorgänger des Koordinatensystems in der Kette enthält. Der "Urspung" der Kette steht immer am Ende der Zeile.

Hinweise: Für eine übersichtliche Darstellung werden für Betriebs-Koordinatensysteme der Typen KSD, KSF, KSW und KST nur die Vorgänger bis zum Koordinatensystem ZERO aufgelistet. Die Vorgänger von ZERO können jedoch gezielt abgefragt werden und sind auch als separate Zeile in der Antwort enthalten, wenn <CSName> in der Abfrage weggelassen wird.

KLS? (Get Coordinate System Properties)

Beschreibung: Listet die Eigenschaften der im flüchtigen Speicher vorhandenen Koordinatensysteme auf.

Mit `WPA SKS` (S. 293) können die im flüchtigen Speicher vorhandenen Koordinatensysteme in den permanenten Speicher geschrieben werden.

Die Eigenschaften der im flüchtigen Speicher vorhandenen Kombinationen von Koordinatensystemen für das Work-und-Tool-Konzept können mit KLC? (S. 215) abgefragt werden.

Format: KLS? [<CSName>[<Item1>[<Item2>]]]

Argumente: <CSName> ist der Name eines Koordinatensystems. Nicht zulässig: HEXAPOD.

<Item1> ist eine Eigenschaft der Achsen des Controllers. Mögliche Werte:

Für alle Typen von Koordinatensystemen:

- POS: Offset für Positionen der Achsen

Nur für Koordinatensysteme der Typen KSD, KSF und ZERO:

- NLM: Untergrenze des Verfahrbereichs der Achse
- PLM: Obergrenze des Verfahrbereichs der Achse
- SSL: Aktivierungszustand der Verfahrbereichsgrenzen der Achse
- Wenn vom Controller unterstützt:
  SST: Schrittweite für Bewegungen, die durch eine manuelle Bedieneinheit ausgelöst werden

Nur für Koordinatensysteme der Typen KSF und ZERO:

- SPI: Koordinaten des Pivotpunkts

<Item2> ist eine Achse des Controllers, mögliche Werte: X, Y, Z, U, V, W

Wenn die Eigenschaften aller Koordinatensysteme aufgelistet werden sollen, werden alle Argumente weggelassen.

Antwort: <String>

<String> enthält im XML-Format Informationen zu den im flüchtigen Speicher vorhandenen Koordinatensystemen.

Der Aufbau von <String> hängt von der Anzahl der Argumente im gesendeten Befehl ab.

Wenn kein Argument oder nur der Name eines Koordinatensystems angegeben wird, listet <String> zu jedem in der Antwort enthaltenen Koordinatensystem die zutreffenden Angaben für <Item1> und <Item2> und zusätzlich folgende Informationen auf:

- Name des Koordinatensystems (Name="")
- Name des direkten Vorgängers des Koordinatensystems in der Kette (Parent="")

- Aktuelle Verwendung des Koordinatensystems;
  - Used="True": Das Koordinatensystem wird verwendet, d. h. es ist entweder selbst aktiv, oder es ist ein Vorgänger des aktiven Koordinatensystems in dessen Kette.
  - Used="False": Das Koordinatensystem wird nicht verwendet.
- Typ des Koordinatensystems (Type="")

Hinweis: Die Offsetwerte, die in der Antwort auf KLS? für Korrektur-Koordinatensysteme der Typen KLD(USER) und KLF(USER) angezeigt werden, ergeben sich durch Rückrechnung aus allen aktuell aktiven Koordinatensystemen. Sie können sich deshalb beim Wechsel der aktiven Koordinatensysteme ändern.

Beispiel: Das nachfolgende Beispiel zeigt die Antwort auf KLS? für die werkseitig vorhandenen Koordinatensysteme.

```
KLS?
<SingleCoordinateSystem>
     <ZERO Name="ZERO" Parent="PI_BASE" Used="True" Type="ZERO">
          <POS X="0.000000" Y="0.000000" Z="0.000000"
U="0.000000" V="0.000000" W="0.000000"/>
          <NLM X="-10.000100" Y="-10.000100" Z="-10.000100"
U="-1.000100" V="-1.000100" W="-1.000100"/>
          <PLM X="10.000100" Y="10.000100" Z="10.000100"
U="1.000100" V="1.000100" W="1.000100"/>
          <SSL X="1" Y="1" Z="1" U="1" V="1" W="1"/>
          <SPI R="0.000000" S="0.000000" T="481.000000"/>
          <SST X="0.010000" Y="0.010000" Z="0.010000"
U="0.010000" V="0.010000" W="0.010000"/>
     </ZERO>
     <PI_BASE Name="PI_BASE" Parent="PI_LEVELLING" Used="True"
Type="KSB(PI)">
          <POS X="0.000000" Y="0.000000" Z="0.000000"
U="0.000000" V="0.000000" W="0.000000"/>
     </PI_BASE>
     <PI_LEVELLING Name="PI_LEVELLING" Parent="HEXAPOD"
Used="True" Type="KLD(PI)">
          <POS X="0.000000" Y="0.000000" Z="0.000000"
U="0.000000" V="0.000000" W="0.000000"/>
     </PI_LEVELLING></SingleCoordinateSystem>
```

KLT? (Get Offset Resulting From A Chain)

Beschreibung: Listet für ein Koordinatensystem die Offsetwerte für die Position der Achsen X, Y, Z, U, V, W auf, die aus den Offsetwerten der mit diesem Koordinatensystem verketteten Vorgänger resultieren.

Für Betriebs-Koordinatensysteme gehen nur die als Vorgänger verketteten Betriebs-Koordinatensysteme bis zum Koordinatensystem ZERO in die Berechnung ein.

Für Achsorientierungs-Koordinatensysteme gehen nur die als Vorgänger verketteten Achsorientierungs-Koordinatensysteme bis zum Koordinatensystem HEXAPOD in die Berechnung ein.

Für Korrektur-Koordinatensysteme gehen nur die als Vorgänger verketteten Korrektur-Koordinatensysteme bis zum Koordinatensystem HEXAPOD in die Berechnung ein.

Format: KLT? [<StartCS> [<EndCS>]]

Argumente: <StartCS> ist der Name des Koordinatensystems, für das die aus seinen Vorgängern resultierenden Offsetwerte abgefragt werden sollen.

<EndCS> ist der Name eines als Vorgänger von <StartCS> verketteten Koordinatensystems, das als Ausgangspunkt der Offsetberechnung verwendet werden soll. Wenn <EndCS> weggelassen wird, hängt der Ausgangspunkt der Berechnung von der Angabe für <StartCS> ab:

- <StartCS> ist ein Betriebs-Koordinatensystem (Typ KSD, KSF, KST, KSW oder ZERO): Ausgangspunkt ist ZERO
- <StartCS> ist ein Achsorientierungs- oder Korrektur-Koordinatensystem (Typen KSB(PI), KSB(USER), KLD(PI), KLD(USER), KLF(PI), KLF(USER)): Ausgangspunkt ist HEXAPOD

Wenn die resultierenden Offsetwerte für alle Koordinatensysteme aufgelistet werden sollen, werden alle Argumente weggelassen.

Antwort: <String>

<String> enthält für jedes abgefragte Koordinatensystem eine Zeile mit folgenden Angaben:

- Name: Der Name des Koordinatensystems, für das die resultierenden Offsetwerte aufgelistet werden.
- EndCoordinateSystem: Der Name des Koordinatensystems, das als Ausgangspunkt der Berechnung der Offsetwerte verwendet wurde.
- X, Y, Z, U, V, W: Resultierender Offsetwert für die entsprechende Achse.

Hinweis: KLT? fragt den flüchtigen Speicher ab.

Die Eigenschaften der im flüchtigen Speicher vorhandenen Koordinatensysteme können mit KLS? (S. 223) abgefragt werden.

Beispiel: Das nachfolgende Beispiel zeigt die Antwort auf KLT? für die werkseitig vorhandenen Koordinatensysteme.

```
KLT?
Name=ZERO  EndCoordinateSystem=ZERO   X=0.000000 Y=0.000000
     Z=0.000000 U=0.000000 V=0.000000 W=0.000000
Name=PI_BASE    EndCoordinateSystem=HEXAPOD     X=0.000000
     Y=0.000000 Z=0.000000 U=0.000000 V=0.000000 W=0.000000
Name=PI_LEVELLING    EndCoordinateSystem=HEXAPOD     X=0.000000
     Y=0.000000 Z=0.000000 U=0.000000 V=0.000000 W=0.000000
```

KRM (Remove Coordinate System)

Beschreibung: Löscht ein Koordinatensystem.

Format: KRM <CSName>

Argumente: <CSName> ist der Name des Koordinatensystems, das gelöscht werden soll.

Antwort: Keine

Hinweise: Ein Koordinatensystem, das verwendet wird (d. h. das selbst aktiv ist oder als Vorgänger mit dem aktiven Koordinatensystem verkettet ist), kann nicht gelöscht werden.

Beim Löschen eines (nicht verwendeten) Koordinatensystems werden dessen Vorgänger und Nachfolger in der Kette von Koordinatensystemen miteinander verbunden.

Vor dem Löschen eines Korrektur-Koordinatensystems der Typen KLD(USER), KLF(USER) und KSB(USER) ist der Wechsel auf Befehlsebene 1 erforderlich (siehe CCL (S. 162)).

Das Koordinatensystem wird im flüchtigen Speicher gelöscht. Mit `WPA SKS` (S. 293) kann das Löschen in den permanenten Speicher übertragen werden.

KSB (Define Orientational Coordinate System)

Beschreibung: Definiert ein Achsorientierungs-Koordinatensystem vom Typ KSB(USER) zum dauerhaften Ändern der Richtung der Achsen X und/oder Y und/oder Z.

Die Richtung der Achsen wird durch Drehen des Koordinatensystems in 90°-Schritten wie folgt geändert:

- Drehen um X, d. h. Angabe eines Winkelwerts für U, ändert die Richtung der Achsen Y und Z
- Drehen um Y, d. h. Angabe eines Winkelwerts für V, ändert die Richtung der Achsen X und Z
- Drehen um Z, d. h. Angabe eines Winkelwerts für W, ändert die Richtung der Achsen X und Y

Das Koordinatensystem wird im flüchtigen Speicher definiert. Mit `WPA SKS` (S. 293) kann die Definition in den permanenten Speicher geschrieben werden.

Format: KSB <CSName> [{<AxisID> <Angle>}]

Argumente: <CSName> ist der Name des Koordinatensystems, das definiert werden soll.

<AxisID> ist eine Achse des Controllers. Mögliche Werte: U, V, W.

<Angle> ist der Winkel, um den die Achse gedreht werden soll. Mögliche Werte: 0, 90, 180, 270, -90, -180, -270 (Einheit: Grad).

Für Achsen, die im Befehl KSB nicht angegeben werden, wird der Winkel auf null gesetzt.

Antwort: Keine

Hinweise: Vor dem Definieren eines Achsorientierungs-Koordinatensystems ist der Wechsel auf Befehlsebene 1 erforderlich (siehe CCL (S. 162)).

Möglichkeiten zum Definieren eines Koordinatensystems mit KSB:

- <CSName> ist ein neuer Name. Das Achsorientierungs-Koordinatensystem wird unter diesem Namen neu angelegt und hat automatisch das aktive Achsorientierungs-Koordinatensystem als Vorgänger
- <CSName> ist der Name eines existierenden Koordinatensystems, das nicht verwendet wird. Das Koordinatensystem wird durch die Definition mit KSB überschrieben und automatisch als Nachfolger an das aktive Achsorientierungs-Koordinatensystem angehängt

Achsorientierungs-Koordinatensysteme vom Typ KSB(USER) können mit KLN (S. 221) mit anderen Koordinatensystemen vom Typ KSB(User) verkettet oder als Nachfolger an das Koordinatensystem PI_Base angehängt werden.

Achsorientierungs-Koordinatensysteme vom Typ KSB(USER) können mit KEN (S. 211) aktiviert werden. DPA SKS (S. 167) aktiviert wieder das werkseitig aktivierte Achsorientierungs-Koordinatensystem PI_Base, Details siehe KEN.

KSD (Define Operating Coordinate System By Specifying Values)

Beschreibung: Definiert ein Betriebs-Koordinatensystem vom Typ KSD.

Ein Koordinatensystem vom Typ KSD kann beliebig im Raum platziert und ausgerichtet werden. Siehe "Das Work-und-Tool-Konzept" und insbesondere "Betrachtung von Koordinatensystemen der Typen KSD und KSF aus dem Blickwinkel von "Work" und "Tool"" im Dokument "Koordinatensysteme für Hexapod-Mikroroboter" (C887T0007) sowie "Begriffserklärung" (S. 2) für weiterführende Erläuterungen.

Die Platzierung des Koordinatensystems im Raum wird durch Angabe von Offsetwerten für die Achsen X, Y, Z festgelegt. Die Ausrichtung des Koordinatensystems wird durch Angabe von Offsetwerten für die Achsen U, V, W festgelegt:

- Drehen um X, d. h. Angabe eines Offsets für U, ändert die Richtung der Achsen Y und Z
- Drehen um Y, d. h. Angabe eines Offsets für V, ändert die Richtung der Achsen X und Z
- Drehen um Z, d. h. Angabe eines Offsets für W, ändert die Richtung der Achsen X und Y

Wenn ein Betriebs-Koordinatensystem vom Typ KSD mit KEN (S. 211) aktiviert wurde:

- Das Koordinatensystem bestimmt die Positionsangaben für die Bewegungsplattform des Hexapods (Abfrage der aktuellen Position mit POS? (S. 252)).
- Die Koordinaten des Drehpunkts können **nicht** mit SPI (S. 260) geändert werden, und der mit SPI definierte Pivotpunkt wird **nicht** verwendet.

Wenn ein Koordinatensystem vom Typ KSD selbst nicht aktiv ist, jedoch als Vorgänger mit dem aktiven Betriebs-Koordinatensystem verkettet ist, gehen seine Offsetwerte in die Berechnung der Offsetwerte des aktiven Betriebs-Koordinatensystems ein (siehe KLT? (S. 226)).

Das Koordinatensystem wird im flüchtigen Speicher definiert. Mit `WPA SKS` (S. 293) kann die Definition in den permanenten Speicher geschrieben werden.

Format: KSD <CSName> [{<AxisID> <Offset>}]

Argumente: <CSName> ist der Name des Koordinatensystems, das definiert werden soll.

<AxisID> ist eine Achse des Controllers. Mögliche Werte: X, Y, Z, U, V, W.

<Offset> ist ein Offsetwert für die Platzierung oder Ausrichtung der Achse; in physikalischen Einheiten.

Für Achsen, die im Befehl KSD nicht angegeben werden, wird der Offset auf null gesetzt.

Antwort: Keine

Hinweise: Wenn ein Betriebskoordinatensystem vom Typ KSD aktiv ist:

- Neben den Offsetwerten für die Achsen X, Y, Z, U, V, W gibt das Koordinatensystem Werte für die folgenden Einstellungen vor:
  - NLM (S. 248): Untergrenze des Verfahrbereichs der Achse
  - PLM (S. 251): Obergrenze des Verfahrbereichs der Achse
  - SSL (S. 263): Aktivierungszustand der Verfahrbereichsgrenzen der Achse
  - Wenn vom Controller unterstützt:
    SST: Schrittweite für Bewegungen, die durch eine manuelle Bedieneinheit ausgelöst werden
- Die aktuellen Einstellungen können mit den entsprechenden Befehlen geändert werden.
- Die aktuellen Einstellungen können mit KLS? (S. 223) abgefragt werden.
- Die aktuellen Einstellungen werden mit `WPA SKS` (S. 293) für das Koordinatensystem gespeichert.
- Für Details zum Auslösen von Bewegungen siehe "Kommandieren von Bewegungen" im Dokument "Koordinatensysteme für Hexapod-Mikroroboter" (C887T0007).

Möglichkeiten zum Definieren eines Koordinatensystems mit KSD:

- <CSName> ist ein neuer Name. Das Betriebs-Koordinatensystem wird unter diesem Namen neu angelegt und hat automatisch das Betriebs-Koordinatensystem ZERO als Vorgänger.
- <CSName> ist der Name eines existierenden Koordinatensystems, das nicht verwendet wird. Das Koordinatensystem wird durch die Definition mit KSD überschrieben.
  - Wenn das überschriebene Koordinatensystem ebenfalls vom Typ KSD war, bleibt die Verkettung mit seinem Vorgänger und Nachfolger erhalten.
  - Wenn das überschriebene Koordinatensystem nicht vom Typ KSD war, wird sein Typ in KSD geändert, und es wird an ZERO als neuen Vorgänger angehängt.

Betriebs-Koordinatensysteme der Typen KSD, KSF, KST, KSW können mit KLN (S. 221) mit anderen Betriebs-Koordinatensystemen dieser Typen verkettet oder wieder als Nachfolger an das Koordinatensystem ZERO angehängt werden.

`KEN ZERO` und `DPA SKS` aktivieren wieder das werkseitig aktivierte Betriebs-Koordinatensystem ZERO, Details siehe KEN (S. 211).

KSF (Define Operating Coordinate System At Current Position)

Beschreibung: Definiert ein Betriebs-Koordinatensystem vom Typ KSF an der aktuellen Position der Bewegungsplattform des Hexapods.

Siehe "Das Work-und-Tool-Konzept" und insbesondere "Betrachtung von Koordinatensystemen der Typen KSD und KSF aus dem Blickwinkel von "Work" und "Tool"" im Dokument "Koordinatensysteme für Hexapod-Mikroroboter" (C887T0007) sowie "Begriffserklärung" (S. 2) für weiterführende Erläuterungen.

Wenn ein Betriebs-Koordinatensystem vom Typ KSF mit KEN (S. 211) aktiviert wurde, bestimmt es die Positionsangaben für die Bewegungsplattform des Hexapods (Abfrage der aktuellen Position mit POS? (S. 252)), und der mit SPI (S. 260) definierte Pivotpunkt wird als Drehpunkt für Rotationen verwendet.

Wenn ein Koordinatensystem vom Typ KSF selbst nicht aktiv ist, jedoch als Vorgänger mit dem aktiven Betriebs-Koordinatensystem verkettet ist, gehen seine Offsetwerte in die Berechnung der Offsetwerte des aktiven Betriebs-Koordinatensystems ein (siehe KLT? (S. 226)).

Die Definition mit KSF ist nur möglich, wenn sich der Hexapod nicht bewegt.

Das Koordinatensystem wird im flüchtigen Speicher definiert. Mit `WPA SKS` (S. 293) kann die Definition in den permanenten Speicher geschrieben werden.

Format: KSF <CSName>

Argumente: <CSName> ist der Name des Koordinatensystems, das definiert werden soll.

Antwort: Keine

Hinweise: Wenn ein Koordinatensystem vom Typ KSF definiert wird, werden seine Pivotpunktkoordinaten (R, S, T, siehe SPI (S. 260)) auf die Werte gesetzt, die für das aktuell aktive Betriebs-Koordinatensystem gültig sind. Wenn das aktive Betriebs-Koordinatensystem den mit SPI gesetzten Pivotpunkt nicht unterstützt (Typen KSD, KST/KSW), werden die Pivotpunktkoordinaten des KSF-Koordinatensystems auf R = S = T = 0 gesetzt.

Wenn ein Betriebs-Koordinatensystem vom Typ KSF aktiv ist:

- Neben den Offsetwerten für die Achsen X, Y, Z, U, V, W gibt das Koordinatensystem Werte für die folgenden Einstellungen vor:
  - NLM (S. 248): Untergrenze des Verfahrbereichs der Achse
  - PLM (S. 251): Obergrenze des Verfahrbereichs der Achse
  - SSL (S. 263): Aktivierungszustand der Verfahrbereichsgrenzen der Achse
  - SPI (S. 260): Pivotpunktkoordinaten R, S, T
  - Wenn vom Controller unterstützt:
    SST: Schrittweite für Bewegungen, die durch eine manuelle Bedieneinheit ausgelöst werden
- Die aktuellen Einstellungen können mit den entsprechenden Befehlen geändert werden. Die Pivotpunktkoordinaten können mit SPI nur geändert werden, wenn für die aktuelle Position der Plattform gilt U = V = W = 0.
- Die aktuellen Einstellungen können mit KLS? (S. 223) abgefragt werden.
- Die aktuellen Einstellungen werden mit `WPA SKS` (S. 293) für das Koordinatensystem gespeichert.
- Für Details zum Auslösen von Bewegungen siehe "Kommandieren von Bewegungen" im Dokument "Koordinatensysteme für Hexapod-Mikroroboter" (C887T0007).

Möglichkeiten zum Definieren eines Koordinatensystems mit KSF:

- <CSName> ist ein neuer Name. Das Betriebs-Koordinatensystem wird unter diesem Namen neu angelegt und hat automatisch das Betriebs-Koordinatensystem ZERO als Vorgänger.
- <CSName> ist der Name eines existierenden Koordinatensystems, das nicht verwendet wird. Das Koordinatensystem wird durch die Definition mit KSF überschrieben und automatisch als Nachfolger an das Betriebs-Koordinatensystem ZERO angehängt.

Betriebs-Koordinatensysteme der Typen KSD, KSF, KST, KSW können mit KLN (S. 221) mit anderen Betriebs-Koordinatensystemen dieser Typen verkettet oder wieder als Nachfolger an das Koordinatensystem ZERO angehängt werden.

`KEN ZERO` und `DPA SKS` aktivieren wieder das werkseitig aktivierte Betriebs-Koordinatensystem ZERO, Details siehe KEN (S. 211).

KST (Define "Tool" Operating Coordinate System)

Beschreibung: Definiert ein Betriebs-Koordinatensystem vom Typ KST ("Tool-Koordinatensystem") für das Work-und-Tool-Konzept.

Siehe "Das Work-und-Tool-Konzept" im Dokument "Koordinatensysteme für Hexapod-Mikroroboter" (C887T0007) sowie "Begriffserklärung" (S. 2) für weiterführende Erläuterungen.

Die Platzierung des Tool-Koordinatensystems wird durch Angabe von Offsetwerten für die Achsen X, Y, Z festgelegt. Die Ausrichtung des Tool-Koordinatensystems wird durch Angabe von Offsetwerten für die Achsen U, V, W festgelegt:

- Drehen um X, d. h. Angabe eines Offsets für U, ändert die Richtung der Achsen Y und Z
- Drehen um Y, d. h. Angabe eines Offsets für V, ändert die Richtung der Achsen X und Z
- Drehen um Z, d. h. Angabe eines Offsets für W, ändert die Richtung der Achsen X und Y

Wenn ein Betriebs-Koordinatensystem vom Typ KST aktiv ist, wird mit dem Work-und-Tool-Konzept gearbeitet, und Folgendes gilt:

- Zusätzlich zum Tool-Koordinatensystem muss ein Work-Koordinatensystem aktiv sein. Wenn kein Work-Koordinatensystem vom Typ KSW aktiv ist, wird als Ersatz ein automatisch erzeugtes Work-Koordinatensystem verwendet.
- Die Koordinaten des Drehpunkts können **nicht** mit SPI (S. 260) geändert werden, und der mit SPI definierte Pivotpunkt wird **nicht** verwendet.

Wenn ein Koordinatensystem vom Typ KST selbst nicht aktiv ist, jedoch als Vorgänger mit dem aktiven Betriebs-Koordinatensystem verkettet ist, gehen seine Offsetwerte in die Berechnung der Offsetwerte des aktiven Betriebs-Koordinatensystems ein (siehe KLT? (S. 226)).

Das Koordinatensystem wird im flüchtigen Speicher definiert. Mit `WPA SKS` (S. 293) kann die Definition in den permanenten Speicher geschrieben werden.

Format: KST <CSName> [{<AxisID> <Offset>}]

Argumente: <CSName> ist der Name des Koordinatensystems, das definiert werden soll.

<AxisID> ist eine Achse des Controllers. Mögliche Werte: X, Y, Z, U, V, W.

<Offset> ist ein Offsetwert für die Platzierung oder Ausrichtung der Achse; in physikalischen Einheiten.

Für Achsen, die im Befehl KST nicht angegeben werden, wird der Offset auf null gesetzt.

Antwort: Keine

Hinweise: Wenn eine Kombination aus Koordinatensystemen der Typen KSW/KST oder ZERO/KST oder KSW/ZERO aktiv ist:

- Neben den Offsetwerten für die Achsen X, Y, Z, U, V, W gibt die Kombination Werte für die folgenden Einstellungen vor:
  - NLM (S. 248): Untergrenze des Verfahrbereichs der Achse
  - PLM (S. 251): Obergrenze des Verfahrbereichs der Achse
  - SSL (S. 263): Aktivierungszustand der Verfahrbereichsgrenzen der Achse
  - Wenn vom Controller unterstützt:
    SST: Schrittweite für Bewegungen, die durch eine manuelle Bedieneinheit ausgelöst werden
- Die aktuellen Einstellungen können mit den entsprechenden Befehlen geändert werden.
- Die aktuellen Einstellungen können mit KLC? (S. 215) abgefragt werden.
- Die aktuellen Einstellungen werden mit `WPA SKS` (S. 293) für die Koordinatensystem-Kombination gespeichert.
- Für Details zum Auslösen von Bewegungen siehe "Kommandieren von Bewegungen" im Dokument "Koordinatensysteme für Hexapod-Mikroroboter" (C887T0007).

Möglichkeiten zum Definieren eines Koordinatensystems mit KST:

- <CSName> ist ein neuer Name. Das Betriebs-Koordinatensystem wird unter diesem Namen neu angelegt und hat automatisch das Betriebs-Koordinatensystem ZERO als Vorgänger.
- <CSName> ist der Name eines existierenden Koordinatensystems, das nicht verwendet wird. Das Koordinatensystem wird durch die Definition mit KST überschrieben.
  - Wenn das überschriebene Koordinatensystem ebenfalls vom Typ KST war, bleibt die Verkettung mit seinem Vorgänger und Nachfolger erhalten.
  - Wenn das überschriebene Koordinatensystem nicht vom Typ KST war, wird sein Typ in KST geändert, und es wird an ZERO als neuen Vorgänger angehängt.

Betriebs-Koordinatensysteme der Typen KSD, KSF, KST, KSW können mit KLN (S. 221) mit anderen Betriebs-Koordinatensystemen dieser Typen verkettet oder wieder als Nachfolger an das Koordinatensystem ZERO angehängt werden.

`KEN ZERO` und `DPA SKS` aktivieren wieder das werkseitig aktivierte Betriebs-Koordinatensystem ZERO, Details siehe KEN (S. 211).

KSW (Define "Work" Operating Coordinate System)

Beschreibung: Definiert ein Betriebs-Koordinatensystem vom Typ KSW ("Work-Koordinatensystem") für das Work-und-Tool-Konzept.

Siehe "Das Work-und-Tool-Konzept" im Dokument "Koordinatensysteme für Hexapod-Mikroroboter" (C887T0007) sowie "Begriffserklärung" (S. 2) für weiterführende Erläuterungen.

Die Platzierung des Work-Koordinatensystems wird durch Angabe von Offsetwerten für die Achsen X, Y, Z festgelegt. Die Ausrichtung des Work-Koordinatensystems wird durch Angabe von Offsetwerten für die Achsen U, V, W festgelegt:

- Drehen um X, d. h. Angabe eines Offsets für U, ändert die Richtung der Achsen Y und Z
- Drehen um Y, d. h. Angabe eines Offsets für V, ändert die Richtung der Achsen X und Z
- Drehen um Z, d. h. Angabe eines Offsets für W, ändert die Richtung der Achsen X und Y

Wenn ein Betriebs-Koordinatensystem vom Typ KSW aktiv ist, wird mit dem Work-und-Tool-Konzept gearbeitet, und Folgendes gilt:

- Zusätzlich zum Work-Koordinatensystem muss ein Tool-Koordinatensystem aktiv sein. Wenn kein Tool-Koordinatensystem vom Typ KST aktiv ist, wird als Ersatz ein automatisch erzeugtes Tool-Koordinatensystem verwendet.
- Die Koordinaten des Drehpunkts können **nicht** mit SPI (S. 260) geändert werden, und der mit SPI definierte Pivotpunkt wird **nicht** verwendet.

Wenn ein Koordinatensystem vom Typ KSW selbst nicht aktiv ist, jedoch als Vorgänger mit dem aktiven Betriebs-Koordinatensystem verkettet ist, gehen seine Offsetwerte in die Berechnung der Offsetwerte des aktiven Betriebs-Koordinatensystems ein (siehe KLT? (S. 226)).

Das Koordinatensystem wird im flüchtigen Speicher definiert. Mit `WPA SKS` (S. 293) kann die Definition in den permanenten Speicher geschrieben werden.

Format: KSW <CSName> [{<AxisID> <Offset>}]

Argumente: <CSName> ist der Name des Koordinatensystems, das definiert werden soll.

<AxisID> ist eine Achse des Controllers. Mögliche Werte: X, Y, Z, U, V, W.

<Offset> ist ein Offsetwert für die Platzierung oder Ausrichtung der Achse; in physikalischen Einheiten.

Für Achsen, die im Befehl KSW nicht angegeben werden, wird der Offset auf null gesetzt.

Antwort: Keine

Hinweise: Wenn eine Kombination aus Koordinatensystemen der Typen KSW/KST oder ZERO/KST oder KSW/ZERO aktiv ist:

- Neben den Offsetwerten für die Achsen X, Y, Z, U, V, W gibt die Kombination Werte für die folgenden Einstellungen vor:
  - NLM (S. 248): Untergrenze des Verfahrbereichs der Achse
  - PLM (S. 251): Obergrenze des Verfahrbereichs der Achse
  - SSL (S. 263): Aktivierungszustand der Verfahrbereichsgrenzen der Achse
  - Wenn vom Controller unterstützt:
    SST: Schrittweite für Bewegungen, die durch eine manuelle Bedieneinheit ausgelöst werden
- Die aktuellen Einstellungen können mit den entsprechenden Befehlen geändert werden.
- Die aktuellen Einstellungen können mit KLC? (S. 215) abgefragt werden.
- Die aktuellen Einstellungen werden mit `WPA SKS` (S. 293) für die Koordinatensystem-Kombination gespeichert.
- Für Details zum Auslösen von Bewegungen siehe "Kommandieren von Bewegungen" im Dokument "Koordinatensysteme für Hexapod-Mikroroboter" (C887T0007).

Möglichkeiten zum Definieren eines Koordinatensystems mit KSW:

- <CSName> ist ein neuer Name. Das Betriebs-Koordinatensystem wird unter diesem Namen neu angelegt und hat automatisch das Betriebs-Koordinatensystem ZERO als Vorgänger.
- <CSName> ist der Name eines existierenden Koordinatensystems, das nicht verwendet wird. Das Koordinatensystem wird durch die Definition mit KSW überschrieben.
  - Wenn das überschriebene Koordinatensystem ebenfalls vom Typ KSW war, bleibt die Verkettung mit seinem Vorgänger und Nachfolger erhalten.
  - Wenn das überschriebene Koordinatensystem nicht vom Typ KSW war, wird sein Typ in KSW geändert, und es wird an ZERO als neuen Vorgänger angehängt.

Betriebs-Koordinatensysteme der Typen KSD, KSF, KST, KSW können mit KLN (S. 221) mit anderen Betriebs-Koordinatensystemen dieser Typen verkettet oder wieder als Nachfolger an das Koordinatensystem ZERO angehängt werden.

`KEN ZERO` und `DPA SKS` aktivieren wieder das werkseitig aktivierte Betriebs-Koordinatensystem ZERO, Details siehe KEN (S. 211).

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

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> [{<String>}]
MAC START <macroname> [{<String>}]

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 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 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 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> [{<String>}]

Wiederholt das angegebene Makro <uint> Mal. Eine neue Ausführung wird gestartet, wenn die letzte beendet ist.

<String> steht für den Wert einer im Makro enthaltenen lokalen Variablen. 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. Pro Befehlszeile sind maximal 256 Zeichen zulässig. <String> kann direkt oder über den Wert einer anderen Variable angegeben werden. Nähere Angaben siehe "Variablen" (S. 135).

MAC START <macroname> [{<String>}]
Startet eine Ausführung des angegebenen Makros. <String> hat 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.

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

Makros können lokale und globale Variablen beinhalten. Die Namen der in einem Makro verwendeten lokalen Variablen müssen eine fortlaufende Reihe bilden. Beispiel für erlaubte Benennung: 1, 2, 3, 4. Nicht erlaubt ist z .B. die Benennung mit 1, 2, 5, 6. Weitere Informationen finden Sie im Abschnitt "Variablen" (S. 135).

Ein laufendes Makro sendet keine Antworten an eine Schnittstelle.

In Abhängigkeit vom Parameterwert 0x72 (Ignore Macro Error) bestehen die folgenden Möglichkeiten, wenn ein Fehler durch ein laufendes Makro verursacht wird:

0 = Makroausführung wird gestoppt
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-887 zur Verfügung gestellten Befehle können nur in Makros verwendet werden:
ADD (S. 161), CPY (S. 163), DEL (S. 165), JRC (S. 210), MEX (S. 241) und WAC (S. 281).

Ein Makro kann ein anderes Makro starten. Die Höchstzahl der Verschachtelungsebenen beträgt 10. 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; nur der letzte Befehl wird, unabhängig von seiner Quelle, ausgeführt.

Die Makroausführung kann durch #24 (S. 156) und STP (S. 267) 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. 155) können Sie abfragen, ob ein Makro aktuell auf dem Controller ausgeführt wird.

Warnung: Die Anzahl von Schreibzyklen im permanenten Speicher ist begrenzt.

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.

Hinweise: Ein detaillierter Hilfetext kann für folgende GCS-Befehle angezeigt werden:
FDR, SIC, WPA, WAV, WTR, WGO, WAV?

Beispiel: Senden: `MAN? WPA`
Empfangen:

```
WPA <Password> [{<ItemID> <PamID>}] Save
Parameters To Non-Volatile Memory
#AvailablePasswords
<Pswd> <Param_Setting>
100 All Parameters, Settings of Coordinate
System Configuration and Assignment of Axes A, B
101 All Parameters
SKS Settings of Coordinate System Configuration
A12 Assignment of Axes A, B
end of help
```

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. 281).

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

Fehlersuche:

- Zielposition außerhalb des aktuellen Arbeitsraums.
- Unzulässige Achsenkennung
  Weitere Informationen finden Sie im Abschnitt "Kommandierbare Elemente" (S. 27).
- Servomodus ist Aus für eine der angegebenen Achsen.
- Für mindestens eine Achse wurde die Referenzierungsfahrt nicht erfolgreich abgeschlossen.

Hinweise: Der Servomodus muss bei Verwendung dieses Befehls eingeschaltet sein (geregelter Betrieb).

Die physikalische Einheit, in der <Position> anzugeben ist, kann mit PUN? (S. 253) abgefragt werden.

Um festzustellen, ob eine Bewegung abgeschlossen ist, wird empfohlen, #5 (S. 153) zu senden.

Die Bewegung kann durch #24 (S. 156), STP (S. 267) und HLT (S. 204) abgebrochen werden.

Für die Achsen des Hexapods (X, Y, Z, U, V, W) gilt Folgendes:

- Vor dem Start jeder Bewegung erfolgt eine Prüfung, ob die Bewegungsplattform die kommandierte Zielposition tatsächlich erreichen kann. Mit VMO? (S. 280) können Sie abfragen, ob die Zielposition erreicht werden kann.
- Je nach Einstellung des Parameters ***Trajectory Source*** (ID 0x19001900) wird das Dynamikprofil für die Achsen des Hexapods (X, Y, Z, U, V, W) durch eine der beiden folgende Quellen vorgegeben:
  - Profilgenerator des C-887
  - Zyklische Übertragung von Zielpositionen durch aufeinander folgende MOV-Befehle

Weitere Informationen siehe "Bewegungen des Hexapods" (S. 34).

Beispiel 1: Senden: MOV X 10 U 5
Hinweis: Achse X bewegt sich nach 10 (Zielposition in mm), Achse U bewegt sich nach 5 (Zielposition in °)

Beispiel 2: Senden: MOV X 4 Y 2.3 Z -3 U -5.3 V 3 W 1
Hinweis: Mit einem einzigen Bewegungsbefehl können Zielpositionen für alle sechs Achsen des Hexapods gesetzt werden.

Beispiel 3: Senden: MOV Z 100
Senden: ERR?
Empfangen: 7
Hinweis: Die Achse bewegt sich nicht. Der Fehlercode „7" in der Antwort auf den Befehl ERR? (S. 176) zeigt an, dass die im Bewegungsbefehl angegebene Zielposition außerhalb der Grenzwerte ist

MOV? (Get Target Position)

Beschreibung: Fragt die letzte gültige kommandierte Zielposition ab.

Format: MOV? [{<AxisID>}]

Argumente: <AxisID> ist eine Achse des Controllers.

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

wobei

<float> die letzte kommandierte Zielposition in physikalischen Einheiten ist.

Fehlersuche: Unzulässige Achsenkennung

Hinweise: Die Zielposition kann durch unterschiedliche Quellen geändert werden, siehe "Bewegungen des Hexapods" (S. 34).
MOV? fragt die kommandierten Positionen ab. Verwenden Sie POS? (S. 252), um die aktuellen Positionen abzufragen.

MRT (Set Target Relative In Tool Coordinate System)

Beschreibung: Bewegt die angegebene Achse relativ im Tool-Koordinatensystem.

Siehe "Das Work-und-Tool-Konzept" im Dokument "Koordinatensysteme für Hexapod-Mikroroboter" (C887T0007) sowie "Begriffserklärung" (S. 2) für weiterführende Erläuterungen.

Während der Ermittlung der anzufahrenden Zielposition aus den Werten für <Distance> wird zuerst die Translation berechnet und danach die Rotation.

Der Servomodus muss für die kommandierte Achse vor dem Einsatz dieses Befehls eingeschaltet sein (geregelter Betrieb).

Format: MRT {<AxisID> <Distance>}

Argumente: <AxisID> ist eine Achse des Controllers. Mögliche Werte: X, Y, Z, U, V, W.

<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

Fehlersuche:
- Zielposition außerhalb des aktuellen Arbeitsraums.
- Parameter ***Trajectory Source*** (ID 0x19001900) ist auf 1 gesetzt (muss bei Verwendung von MRT jedoch auf 0 gesetzt sein).
- Servomodus ist Aus für eine der angegebenen Achsen.
- Für mindestens eine Achse wurde die Referenzierungsfahrt nicht erfolgreich abgeschlossen.

Hinweise: Wenn nicht mit dem Work-und-Tool-Konzept gearbeitet wird, werden Bewegungen mit MRT in dem Tool-Koordinatensystem ausgeführt, das entsprechend dem aktiven Betriebs-Koordinatensystem besteht (siehe "Betrachtung von Koordinatensystemen der Typen KSD und KSF aus dem Blickwinkel von "Work" und "Tool"" im Dokument "Koordinatensysteme für Hexapod-Mikroroboter" (C887T0007)).

Die physikalische Einheit, in der <Distance> anzugeben ist, kann mit PUN? (S. 253) abgefragt werden.

Um festzustellen, ob eine Bewegung abgeschlossen ist, wird empfohlen, #5 (S. 153) zu senden.

Die Bewegung kann durch #24 (S. 156), STP (S. 267) und HLT (S. 204) abgebrochen werden.

MRW (Set Target Relative In Work Coordinate System)

Beschreibung: Bewegt die angegebene Achse relativ im Work-Koordinatensystem.

Siehe "Das Work-und-Tool-Konzept" im Dokument "Koordinatensysteme für Hexapod-Mikroroboter" (C887T0007) sowie "Begriffserklärung" (S. 2) für weiterführende Erläuterungen.

Während der Ermittlung der anzufahrenden Zielposition aus den Werten für <Distance> wird zuerst die Translation berechnet und danach die Rotation.

Der Servomodus muss für die kommandierte Achse vor dem Einsatz dieses Befehls eingeschaltet sein (geregelter Betrieb).

Format: MRW {<AxisID> <Distance>}

Argumente: <AxisID> ist eine Achse des Controllers. Mögliche Werte: X, Y, Z, U, V, W.

<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

Fehlersuche:

- Zielposition außerhalb des aktuellen Arbeitsraums.
- Parameter ***Trajectory Source*** (ID 0x19001900) ist auf 1 gesetzt (muss bei Verwendung von MRW jedoch auf 0 gesetzt sein).
- Servomodus ist Aus für eine der angegebenen Achsen.
- Für mindestens eine Achse wurde die Referenzierungsfahrt nicht erfolgreich abgeschlossen.

Hinweise: Wenn nicht mit dem Work-und-Tool-Konzept gearbeitet wird, werden Bewegungen mit MRW in dem Work-Koordinatensystem ausgeführt, das entsprechend dem aktiven Betriebs-Koordinatensystem besteht (siehe "Betrachtung von Koordinatensystemen der Typen KSD und KSF aus dem Blickwinkel von "Work" und "Tool"" im Dokument "Koordinatensysteme für Hexapod-Mikroroboter" (C887T0007)).

Die physikalische Einheit, in der <Distance> anzugeben ist, kann mit PUN? (S. 253) abgefragt werden.

Um festzustellen, ob eine Bewegung abgeschlossen ist, wird empfohlen, #5 (S. 153) zu senden.

Die Bewegung kann durch #24 (S. 156), STP (S. 267) und HLT (S. 204) abgebrochen werden.

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

Fehlersuche:

- Zielposition außerhalb des aktuellen Arbeitsraums.
- Unzulässige Achsenkennung
  Weitere Informationen finden Sie im Abschnitt "Kommandierbare Elemente" (S. 27).
- Servomodus ist Aus für eine der angegebenen Achsen.
- Für mindestens eine Achse wurde die Referenzierungsfahrt nicht erfolgreich abgeschlossen.

Hinweise: Der Servomodus muss bei Verwendung dieses Befehls eingeschaltet sein (geregelter Betrieb).

Die physikalische Einheit, in der <Distance> anzugeben ist, kann mit PUN? (S. 253) abgefragt werden.

Um festzustellen, ob eine Bewegung abgeschlossen ist, wird empfohlen, #5 (S. 153) zu senden.

Die Bewegung kann durch #24 (S. 156), STP (S. 267) und HLT (S. 204) abgebrochen werden.

Für die Achsen des Hexapods (X, Y, Z, U, V, W) gilt Folgendes:

- Vor dem Start jeder Bewegung erfolgt eine Prüfung, ob die Bewegungsplattform die kommandierte Zielposition tatsächlich erreichen kann. Mit VMO? (S. 280) können Sie abfragen, ob die Zielposition erreicht werden kann.
- Wenn der Parameter ***Trajectory Source*** (ID 0x19001900) auf 1 gesetzt ist, muss das Dynamikprofil durch aufeinander folgende MOV-Befehle vorgegeben werden. MVR ist nicht zulässig.

Weitere Informationen siehe "Bewegungen des Hexapods" (S. 34).

Beispiel:

Senden: MOV X 0.5
Hinweis: Dies ist eine absolute Bewegung.
Senden: POS? X
Empfangen: X=0.500000
Senden: MOV? X
Empfangen: X=0.500000
Senden: MVR X 2
Hinweis: Dies ist eine relative Bewegung.
Senden: POS? X
Empfangen: X=2.500000
Senden: MVR X 2000

Hinweis: Neue Zielposition von Achse X würde den Bewegungsbereich überschreiten. Befehl wird ignoriert, d. h. die Zielposition bleibt unverändert und die Achse bewegt sich nicht.
Senden: MOV? X
Empfangen: X=2.500000
Senden: POS? X
Empfangen: X=2.500000

NAV (Set Number of Readout Values to be Averaged)

Beschreibung: Legt die Anzahl der Auslesewerte des analogen Eingangs fest, über die der Mittelwert gebildet wird.

Format: NAV {<AnalogInputID> <NumberOfReadings>}

Argumente: <AnalogInputID> ist die Kennung des analogen Eingangskanals

<NumberOfReadings> ist Anzahl der Auslesewerte des Analogsignals

Fehlersuche: Zu hohe Anzahl an Auslesewerten
Unzulässige Kennung des analogen Eingangskanals

Hinweise: Der Standardwert für <NumberOfReadings> ist 1; der zulässige Wertebereich ist 1 bis 10000.

Mit NAV kann der Einfluss von Rauschen am analogen Eingang auf Abfragen (TAV? (S. 270), TAD? (S. 270)) oder auf Scanprozeduren (z. B. AAP (S. 158)) vermindert werden. Je größer das Rauschen des analogen Eingangsignals ist, desto höher sollte mit NAV die Anzahl der Auslesewerte des Analogsignals gesetzt werden, über die der Mittelwert gebildet wird.

Für Fast-Alignment-Routinen, d. h. alle Routinen, die mit den Befehlen FDR und FDG definiert und mit dem Befehl FRS gestartet werden, wird **kein** Mittelwert über die Auslesewerte des analogen Eingangs gebildet. NAV ist für diese Routinen wirkungslos.

Beispiel: Senden: NAV 2 200
Senden: TAV? 2
Hinweis: Die Antwort auf den Befehl TAV? 2 ist der Mittelwert von 200 Auslesewerten des analogen Eingangskanals 2.

NAV? (Get Number of Readout Values to be Averaged)

Beschreibung: Fragt die Anzahl der Auslesewerte des analogen Eingangs ab, über die der Mittelwert gebildet wird.

Format: NAV? [{<AnalogInputID>}]

Argumente: <AnalogInputID> ist die Kennung des analogen Eingangskanals

Antwort: {<AnalogInputID>"="<int> LF}

wobei

<int> die Anzahl der Auslesewerte ist.

Die Antwort besteht aus einem Linefeed, wenn der Controller keinen analogen Eingangskanal enthält.

NLM (Set Low Position Soft Limit)

Beschreibung: Setzt die Untergrenze des Achsenstellwegs im geregelten Betrieb ("Verfahrbereichsgrenze").

Format: NLM {<AxisID> <LowLimit>}

Argumente: <AxisID> ist eine Achse des Controllers

<LowLimit> ist die Position für die Untergrenze des Stellwegs in physikalischen Einheiten

Antwort: Keine

Hinweise: Für die Achsen der Bewegungsplattform des Hexapods werden die Standardwerte für NLM durch das aktive Betriebs-Koordinatensystem vorgegeben (beim Work-und-Tool-Konzept durch die Kombination aus zwei Betriebs-Koordinatensystemen).

Der mit NLM gesetzte Wert muss kleiner als der aktuelle Positionswert sein. Für die Achsen X, Y, Z, U, V und W sind nur negative Werte erlaubt.

Die Verfahrbereichsgrenzen werden mit SSL (S. 263) aktiviert und deaktiviert.

Verfahrbereichsgrenzen können nur gesetzt werden, wenn sich die Achse nicht bewegt (Abfrage mit #5 (S. 153)).

Die physikalische Einheit, in der <LowLimit> anzugeben ist, kann mit PUN? (S. 253) abgefragt werden.

Wenn der Pivotpunkt mit SPI (S. 262) geändert wird, werden die Verfahrbereichsgrenzen für die Rotationsachsen U, V und W nicht angepasst.

Beispiel:

```
Senden: NLM? X
Empfangen: X = -22.5
Senden: POS? X
Empfangen: X = -10
Senden: NLM X -5
Senden: ERR?
Empfangen: 27  -  (error 27 - "Soft limit out of range")
Senden: NLM? X
Empfangen: X = -22.5
```

NLM? (Get Low Position Soft Limit)

Beschreibung: Fragt die Position der Verfahrbereichsgrenze ab, die die Untergrenze des Achsenstellwegs im geregelten Betrieb bestimmt.

Format: NLM? [{<AxisID>}]

Argumente: <AxisID> ist eine Achse des Controllers

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

wobei

<LowLimit> die Position für die Untergrenze des Stellwegs in physikalischen Einheiten ist.

ONT? (Get On-Target State)

Beschreibung: Fragt den On-Target-Status der angegebenen Achse ab.

Werden alle Argumente weggelassen, wird der Status aller Achsen abgefragt.

Format: ONT? [{<AxisID>}]

Argumente: <AxisID> ist eine Achse des Controllers.

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

wobei

<uint> = "1" wenn die angegebene Achse den Zielwert erreicht hat, anderenfalls "0".

Fehlersuche: Unzulässige Achsenkennung

Hinweise: ONT? fragt den Status der Achsen der Bewegungsplattform des Hexapods (X bis W) sowie der Achsen A und B ab.

Für die Achsen der Bewegungsplattform des Hexapods (X bis W) prüft der C-887 bei der Abfrage mit ONT? nur, ob das Ende des Dynamikprofils erreicht ist, aber **nicht** den tatsächlichen Bewegungsstatus. Den tatsächlichen Bewegungsstatus aller Achsen zeigt der C-887 bei der Abfrage mit #5 (S. 153) an, und für die Hexapod-Beine und die Achsen A und B außerdem durch Statusregisterbits, Details siehe "Bewegungsstatus, Einschwingfenster, Einschwingzeit" (S. 46).

PLM (Set High Position Soft Limit)

Beschreibung:	Setzt die Obergrenze des Achsenstellwegs im geregelten Betrieb ("Verfahrbereichsgrenze").
Format:	PLM {<AxisID> <HighLimit>}
Argumente:	<AxisID> ist eine Achse des Controllers <HighLimit> ist die Position für die Obergrenze des Stellwegs in physikalischen Einheiten.
Antwort:	Keine
Hinweise:	Für die Achsen der Bewegungsplattform des Hexapods werden die Standardwerte für PLM durch das aktive Betriebs-Koordinatensystem vorgegeben (beim Work-und-Tool-Konzept durch die Kombination aus zwei Betriebs-Koordinatensystemen). Der mit PLM gesetzte Wert muss größer als der aktuelle Positionswert sein. Für die Achsen X, Y, Z, U, V und W sind nur positive Werte erlaubt. Die Verfahrbereichsgrenzen werden mit SSL (S. 263) aktiviert und deaktiviert. Verfahrbereichsgrenzen können nur gesetzt werden, wenn sich die Achse nicht bewegt (Abfrage mit #5 (S. 153)). Die physikalische Einheit, in der <HighLimit> anzugeben ist, kann mit PUN? (S. 253) abgefragt werden. Wenn der Pivotpunkt mit SPI (S. 262) geändert wird, werden die Verfahrbereichsgrenzen für die Rotationsachsen U, V und W nicht angepasst.
Beispiel:	Senden: PLM? X Empfangen: X = 22.5 Senden: POS? X Empfangen: X = 10 Senden: PLM X 5 Senden: ERR? Empfangen: 27 - (error 27 - "Soft limit out of range") Senden: PLM? X Empfangen: X = 22.5

PLM? (Get High Position Soft Limit)

Beschreibung: Fragt die Position der Verfahrbereichsgrenze ab, die die Obergrenze des Achsenstellwegs im geregelten Betrieb bestimmt.

Format: PLM? [{<AxisID>}]

Argumente: <AxisID> ist eine Achse des Controllers

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

wobei

<HighLimit> die Position für die Obergrenze des Stellwegs in physikalischen Einheiten ist.

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

Hinweise: Dieser Befehl ist funktionsgleich mit #3 (S. 152), der bevorzugt werden sollte, wenn der Controller zeitaufwändige Aufgaben ausführt.

Die aktuelle Position der Achsen X, Y, Z, U, V und W wird aus den gemessenen Positionen der einzelnen Beine berechnet.

Zwischen Einschalten des Controllers und der Referenzierung des Hexapods mit FRF (S. 185) ist die aktuelle Position des Hexapods und der Achsen A und B unbekannt. Dennoch liefert die Antwort auf POS? für alle Achsen den Positionswert 0.

Die physikalische Einheit, in der die Achsenposition angegeben wird, kann mit PUN? (S. 253) abgefragt werden.

PUN? (Get Axis Unit)

Beschreibung:	Fragt die aktuelle Einheit der Achse ab. Werden alle Argumente weggelassen, so wird die aktuelle Einheit für alle Achsen abgefragt.
Format:	PUN? [{<AxisID>}]
Argumente:	<AxisID> ist eine Achse des Controllers.
Antwort:	{<AxisID>"="<string> LF} wobei <string> die aktuelle Einheit der Achse ist.
Fehlersuche:	Unzulässige Achsenkennung
Hinweis:	Für die Positionsangaben der Achsen gelten folgende Einheiten: X, Y, Z: Millimeter U, V, W: Grad A, B: Millimeter oder Grad Die Einheiten sind nicht veränderbar.

RBT (Reboot System)

Beschreibung:	Startet das System neu. Der Controller verhält sich wie nach dem Einschalten.
Format:	RBT
Argumente:	Keine
Antwort:	Keine

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? (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

Hinweise: RON? gibt immer 1 zurück. Für Achsen, deren Position durch inkrementelle Sensoren gemessen wird, bedeutet das:

- Bewegungsbefehle werden nur nach erfolgreicher Referenzierung ausgeführt.
- Die Referenzierung muss durch eine Referenzierungsfahrt erfolgen.

Mit dem C-887 erfolgt die Referenzierungsfahrt zum Referenzschalter (Start mit FRF (S. 185)).

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: RTR setzt den Wert des Parameters ***Data Recorder Table Rate*** (ID 0x16000000) im flüchtigen Speicher. Die Einstellung kann mit dem Befehl WPA (S. 293) im permanenten Speicher des C-887 gespeichert werden, so dass sie nach dem Einschalten oder Neustart des C-887 sofort verfügbar ist.

Werte zwischen 1 und 10000 sind möglich. Der Wert des Parameters ist standardmäßig auf 10 eingestellt (entspricht 1 kHz).

Wenn der C-887 Fast-Alignment-Routinen (S. 2) ausführt, setzt er die Aufzeichnungsrate auf den Faktor 4.

Die Dauer der Aufzeichnung kann wie folgt berechnet werden:

Aufz. Dauer = Zykluszeit * RTR Wert *Anzahl der Punkte

wobei

die Zykluszeit des Datenrekorders 100 µs beträgt

die Anzahl der Punkte für den C-887 maximal 262144 beträgt

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

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).

Hinweis: Fragt den Parameterwert ***Data Recorder Table Rate*** im flüchtigen Speicher ab (ID 0x16000000).

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

SAI? (Get List Of Current Axis Identifiers)

Beschreibung: Fragt die Achsenkennung ab.

Siehe auch "Kommandierbare Elemente" (S. 27).

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.

Hinweis: Die Achsen A und B können „deaktiviert" werden, indem ihnen mit dem Befehl CST (S. 163) der Positionierertyp „NOSTAGE" zugewiesen wird.

Deaktivierte Achsen werden nicht angezeigt.

Ausnahme: In den Antworten auf die Befehle CST? (S. 164) und SAI? ALL sind die deaktivierten Achsen enthalten.

SCT (Set Cycle Time)

Beschreibung: Legt die Zykluszeit für das Ausführen eines Dynamikprofils fest.

Wird verwendet bei der zyklischen Übertragung von Zielpositionen, siehe auch "Bewegungen des Hexapods" (S. 34).

Format: SCT "T" <CycleTime>

Argumente: "T" ist das erforderliche Schlüsselwort für das Argument <CycleTime>

<CycleTime> ist die Zykluszeit in ms, Format: float.

Fehlersuche: Zulässiger Wertebereich überschritten

Hinweise: Der zulässige Wertebereich für <CycleTime> ist 1 bis 10000 ms; Standardwert ist 1 ms.

Die mit SCT eingestellte Zykluszeit ist nur wirksam, wenn der Parameter ***Trajectory Source*** (ID 0x19001900) den Wert 1 hat (= das Dynamikprofil wird durch aufeinander folgende MOV-Befehle festgelegt).

Die Zykluszeit wird verwendet, um während der Bewegung die Geschwindigkeit so zu berechnen, dass die vorgegebenen Punkte des Dynamikprofils jeweils genau am Ende des Zeitintervalls erreicht werden.

Wenn der Parameter ***Trajectory Execution*** (ID 0x19001901) den Wert 1 hat, ist durch die Zwischenspeicherung der Dynamikprofilpunkte sichergestellt, dass das Dynamikprofil im entsprechenden Zeittakt ausgeführt wird.

Wenn der Parameter ***Trajectory Execution*** (ID 0x19001901) den Wert 0 hat, werden die MOV-Befehle sofort nach dem Senden ausgeführt.

- Stellen Sie sicher, dass die MOV-Befehle im der Zykluszeit entsprechenden Zeittakt gesendet werden, um das Dynamikprofil einzuhalten.

Beispiel: Senden: SCT T 30

Setzt die Zykluszeit für das Ausführen eines Dynamikprofils, das durch aufeinander folgende MOV-Befehle vorgegeben wird, auf 30 ms.

SCT? (Get Cycle Time)

Beschreibung: Fragt die aktuelle Zykluszeit für das Abfahren eines definierten Dynamikprofils ab.

Format: SCT? [<T>]

Argumente: "T" dient als Schlüsselwort und kann für die Abfrage weggelassen werden.

Antwort: T=<float> LF

<float> ist die Zykluszeit in ms.

SGA (Set Gain)

Beschreibung: Legt den Verstärkungsfaktor für den angegebenen analogen Eingangskanal fest.

Format: SGA {<AnalogInputID> <Gain>}

Argumente: <AnalogInputID> ist die Kennung des analogen Eingangskanals

<Gain> ist der Verstärkungsfaktor

Fehlersuche: Unzulässiger Wert

Hinweis: Der Standardwert für <Gain> ist 1; andere Werte sind nicht zulässig. Der Befehl SGA ist nur aus Kompatibilitätsgründen vorhanden.

Für die Kennungen der analogen Eingangskanäle siehe "Kommandierbare Elemente" (S. 27).

SGA? (Get Gain)

Beschreibung: Fragt den Verstärkungsfaktor für den angegebenen analogen Eingangskanal ab.

Format: SGA? [{<AnalogInputID>}]

Argumente: <AnalogInputID> ist die Kennung des analogen Eingangskanals, siehe SGA (S. 258)

Antwort: {<AnalogInputID>"="<int> LF}

wobei

<int> der Verstärkungsfaktor des analogen Eingangskanals ist.

SPA (Set Volatile Memory Parameters)

Beschreibung: Setzt einen Parameter des angegebenen Elements im flüchtigen Speicher (RAM) auf einen bestimmten Wert. Parameteränderungen gehen verloren, wenn der Controller abgeschaltet oder neugestartet wird.

Format: SPA {<ItemID> <PamID> <PamValue>}

Argumente: <ItemID> ist das Element, für das ein Parameter im flüchtigen Speicher geändert wird. Nähere Angaben siehe unten.

<PamID> ist die Parameterkennung, kann im Hexadezimal- oder Dezimalformat geschrieben werden. Nähere Angaben siehe unten.

<PamValue> ist der Wert, auf den der Parameter des angegebenen Elements gesetzt wird.

Antwort: Keine

Beachten Sie, dass dieser Befehl für die Einstellung hardwarespezifischer Parameter gilt. Falsche Werte können eventuell zu fehlerhaftem Betrieb oder zur Beschädigung Ihrer Hardware führen!

Mit HPA? (S. 204) erhalten Sie eine Liste der verfügbaren Parameter.

Fehlersuche: Unzulässige Elementkennung, falsche Parameter-ID, Wert im unzulässigen Bereich, zu niedrige Befehlsebene für Schreibzugriff

Parameterwerte können nur gesetzt werden, wenn sich die Achse nicht bewegt (Abfrage mit #5 (S. 153)).

Verfügbare Element-IDs und Parameter-IDs: Das Element kann eine Achse, ein Hexapod-Bein (Antrieb oder Sensor), ein Eingangssignalkanal oder das gesamte System sein; der Elementtyp ist vom Parameter abhängig.
Siehe "Parameterübersicht" (S. 320) für den entsprechenden Elementtyp, bezüglich der Elementkennungen siehe "Kommandierbare Elemente" (S. 27).

Die gültigen Parameter-IDs werden in der "Parameterübersicht" (S. 320) angegeben.

Beispiel: Senden: SPA 1 0x16000000 8
Hinweis: Setzt die Aufzeichnungsrate des Datenrekorders für den Controller auf 8, Parameter-ID im Hexadezimalformat geschrieben
Senden: SPA 1 369098752 2
Hinweis: Setzt die Aufzeichnungsrate des Datenrekorders für den Controller auf 2, Parameter-ID im Dezimalformat geschrieben

SPA? (Get Volatile Memory Parameters)

Beschreibung: Fragt den Wert eines Parameters für ein angegebenes Element aus dem flüchtigen Speicher (RAM) ab.

Mit HPA? (S. 204) erhalten Sie eine Liste der verfügbaren Parameter.

Format:	SPA? [{<ItemID> <PamID>}]
Argumente:	<ItemID> ist das Element, für das ein Parameter im flüchtigen Speicher abgefragt werden soll. Nähere Angaben siehe unten. <PamID> ist die Parameterkennung, kann im Hexadezimal- oder Dezimalformat geschrieben werden. Nähere Angaben siehe unten.
Antwort:	{<ItemID> <PamID>"="<PamValue> LF} wobei <PamValue> der Wert des angegebenen Parameters für das angegebene Element ist.
Fehlersuche:	Unzulässige Elementkennung, falsche Parameterkennung
Verfügbare Element-IDs und Parameter-IDs:	Das Element kann eine Achse, ein Hexapod-Bein (Antrieb oder Sensor), ein Eingangssignalkanal oder das gesamte System sein; der Elementtyp ist vom Parameter abhängig. Siehe "Parameterübersicht" (S. 320) für den entsprechenden Elementtyp, bezüglich der Elementkennungen siehe "Kommandierbare Elemente" (S. 27). Die gültigen Parameter-IDs werden in der "Parameterübersicht" (S. 320) angegeben.

SPI (Set Pivot Point)

Beschreibung:	Verschiebt den Pivotpunkt aus dem Ursprung des Koordinatensystems heraus in X- und/oder Y- und/oder Z-Richtung, wenn die beiden folgenden Bedingungen erfüllt sind: ▪ Als Betriebs-Koordinatensystem ist das Koordinatensystem ZERO oder ein Koordinatensystem vom Typ KSF aktiv. ▪ Für die Rotationskoordinaten der Bewegungsplattform gilt U = V = W = 0 SPI setzt die Pivotpunktkoordinaten im flüchtigen Speicher.
Format:	SPI {<PPCoordinate> <Position>}
Argumente:	<PPCoordinate> ist eine Pivotpunktkoordinate, siehe unten. <Position> ist der Wert der Pivotpunktkoordinate, siehe unten.

Antwort: Keine

Fehlersuche: Mindestens eine der Rotationskoordinaten U, V und W ist ungleich 0

Das aktive Koordinatensystem ist nicht ZERO und nicht vom Typ KSF.

Hinweise: <PPCoordinate> kann R, S und T sein. Als Alias-Kennungen für R, S und T können auch X, Y und Z verwendet werden.

<Position> wird in mm angegeben. Die Standardwerte werden durch das aktive Betriebs-Koordinatensystem vorgegeben (nur für Koordinatensysteme der Typen KSF und ZERO).

Weitere Informationen zum Pivotpunkt siehe "Begriffserklärung" (S. 2).

Bei einer Verlegung des Pivotpunkts mit SPI ändern sich die möglichen Stellwege für die Rotationsachsen U, V und W und damit der Arbeitsraum. Folgende Werte werden jedoch **nicht** an die geänderten Stellwege angepasst:

- Antworten auf TMN? (S. 271) und TMX? (S. 272)
- Verfahrbereichsgrenzen, die mit NLM (S. 248) und PLM (S. 251) gesetzt sind

Verwenden Sie VMO? (S. 280), um abzufragen, ob die Zielposition erreicht werden kann.

Beispiel:

Senden: SPI?
Empfangen: R=0
S=0
T=0
Senden: SPI S 2
Senden: SPI?

Empfangen: R=0
S=2
T=0
Senden: SPI Z 2
Senden: SPI?
Empfangen: R=0
S=2
T=2

SPI? (Get Pivot Point)

Beschreibung:	Fragt die Pivotpunktkoordinaten ab.
Format:	SPI? [{<PPCoordinate>}]
Argumente:	<PPCoordinate> ist eine Pivotpunktkoordinate, siehe unten
Antwort:	{<PPCoordinate>"="<Position> LF} wobei <Position> der Wert der Pivotpunktkoordinate in physikalischen Einheiten ist.
Hinweis:	<PPCoordinate> kann R, S und T sein. Als Alias-Kennungen für R, S und T können auch X, Y und Z verwendet werden.

SRG? (Query Status Register Value)

Beschreibung:	Gibt Registerwerte für die abgefragten Elemente und Register zurück.
Format:	SRG? [{<ItemID> <RegisterID>}]
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.
Mögliche IDs und Antwortwerte:	<ItemID> kann 1 bis 8 sein. Die IDs 1 bis 8 entsprechen den Hexapod-Beinen 1 bis 6 sowie den Achsen A und B in dieser Reihenfolge. <RegisterID> ist immer 1. <Value> ist die bit-codierte Antwort. Für die Beschreibung der Bits siehe "Statusregister für Hexapod-Beine und Achsen A und B" (S. 386).

Hinweise: Die Statusregisterbits, die Sie mit SRG? abfragen, können Sie auch mit dem Datenrekorder des C-887 (S. 99) aufzeichnen, Aufzeichnungsoption 80 (Status register of axis).

Mit den Befehlen #4 (S. 152) und STA? (S. 266) können Sie das Systemstatusregister (S. 386) abfragen.

SSL (Set Soft Limit)

Beschreibung: Aktiviert bzw. deaktiviert die Verfahrbereichsgrenzen, die mit NLM (S. 248) und PLM (S. 251) gesetzt werden.

Format: SSL {<AxisID> <SoftLimitsOn>}

Argumente: <AxisID> ist eine Achse des Controllers

<SoftLimitsOn> ist der Status der Verfahrbereichsgrenzen:

0 = Verfahrbereichsgrenzen deaktiviert
1 = Verfahrbereichsgrenzen aktiviert

Antwort: Keine

Hinweis: Für die Achsen der Bewegungsplattform des Hexapods werden die Standardwerte für SSL durch das aktive Betriebs-Koordinatensystem vorgegeben (beim Work-und-Tool-Konzept durch die Kombination aus zwei Betriebs-Koordinatensystemen).

Verfahrbereichsgrenzen können nur aktiviert/deaktiviert werden, wenn sich die Achse nicht bewegt (Abfrage mit #5 (S. 153)).

Beispiel: Senden: SSL X 1
Die Verfahrbereichsgrenzen für Achse X sind aktiviert.

Senden: SSL Y 0 Z 1 W 1
Die Verfahrbereichsgrenzen sind für die Achse Y deaktiviert und für die Achsen Z und W aktiviert.

SSL? (Get Soft Limit Status)

Beschreibung: Fragt den Status der Verfahrbereichsgrenzen ab, die mit NLM (S. 248) und PLM (S. 251) gesetzt werden.

Werden alle Argumente weggelassen, so wird der Status für alle Achsen abgefragt.

Format: SSL? [{<AxisID>}]

Argumente: <AxisID> ist eine Achse des Controllers

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

wobei

<SoftLimitsOn> der Status der Verfahrbereichsgrenzen ist:
0 = Verfahrbereichsgrenzen deaktiviert
1 = Verfahrbereichsgrenzen aktiviert

Fehlersuche: Unzulässige Achsenkennung

SSN? (Get Device Serial Number)

Beschreibung: Fragt die Seriennummer des C-887 ab.

Format: SSN?

Argumente: Keine

Antwort: <SerialNumber> ist die Seriennummer des Geräts.

SST (Set Step Size)

Beschreibung: Setzt die Strecke ("Schrittweite") für Bewegungen der angegebenen Achse, die durch eine manuelle Bedieneinheit ausgelöst werden.

Format: SST {<AxisID> <StepSize>}

Argumente: <AxisID> ist eine Achse des Controllers

<StepSize> ist die Strecke, Format: float

Antwort: Keine

Fehlersuche: Unzulässiger Wert
Unzulässige Achsenkennung

Hinweis: Die physikalische Einheit, in der <StepSize> anzugeben ist, kann mit PUN? (S. 253) abgefragt werden.

Der zulässige Wertebereich für <StepSize> ist 0 bis 0,5; der Standardwert ist 0,01.

Die mit SST eingestellte Schrittweite wird für Bewegungen der Achsen der Bewegungsplattform des Hexapods (X, Y, Z, U, V, W) verwendet, die durch eine manuelle Bedieneinheit C-887.MC2 oder C-887.MC (S. 24) ausgelöst werden.

Beispiel: Senden: SST Y 0.002 U 0.05
Senden: SST?
Empfangen: X=0.01
Y=0.002
Z=0.01
U=0.05
V=0.01
W=0.01

Senden: SST X 0.09 W 0.09
Senden: SST?
Empfangen: X=0.09
Y=0.002
Z=0.01
U=0.05
V=0.01
W=0.09

SST? (Get Step Size)

Beschreibung: Fragt die Strecke ("Schrittweite") für Bewegungen der angegebenen Achse ab, die durch eine manuelle Bedieneinheit ausgelöst werden.

Format: SST? [{<AxisID>}]

Argumente: <AxisID> ist eine Achse des Controllers

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

wobei

<StepSize> die Strecke in physikalischen Einheiten ist, siehe SST (S. 264).

STA? (Query Status Register Value)

Beschreibung: Fragt die Systemstatusinformation ab.

Format: STA?

Argumente: Keine

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

Hinweise: Die Antwort ist die Summe der Codes in Hexadezimalformat. Für die Beschreibung der Bits und ein Beispiel siehe "Systemstatusregister" (S. 386).

Dieser Befehl ist funktionsgleich mit #4 (S. 152).

Zusätzliche Statusinformationen, die **nicht** mit #4 und STA? abgefragt werden können: Für jedes der Beine 1 bis 6 des Hexapods und für die Achsen A und B hat der C-887 jeweils ein Statusregister (S. 386). Sie können die Bits dieser Register mit dem Befehl SRG? (S. 262) abfragen und mit dem Datenrekorder des C-887 (S. 99) aufzeichnen, Aufzeichnungsoption 80 (Status register of axis).

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. 168) gesetzt werden.

Die aufgezeichneten Daten können mit dem Befehl DRR? (S. 172) 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: Die Zielposition, die aus der angegebenen Sprunggröße resultiert, ist außerhalb der Grenzwerte.

Hinweise: Ein "Sprung" besteht aus einer relativen Bewegung mit der angegebenen Amplitude. Der Schritt wird relativ zu der aktuellen Position ausgeführt.

Die physikalische Einheit, in der <Amplitude> anzugeben ist, kann mit PUN? (S. 253) abgefragt werden.

Für die Achsen des Hexapods (X, Y, Z, U, V, W) gilt Folgendes:

- Vor dem Start jeder Bewegung erfolgt eine Prüfung, ob die Bewegungsplattform die kommandierte Zielposition tatsächlich erreichen kann. Mit VMO? (S. 280) können Sie abfragen, ob die Zielposition erreicht werden kann.
- Wenn der Parameter ***Trajectory Source*** (ID 0x19001900) auf 1 gesetzt ist, muss das Dynamikprofil durch aufeinander folgende MOV-Befehle vorgegeben werden. STE ist nicht zulässig.

Weitere Informationen siehe "Bewegungen des Hexapods" (S. 34).

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. 156).

Format: STP

Argumente: Keine

Antwort: Keine

Fehlersuche: Kommunikationsstörung

Hinweise: STP stoppt alle Achsenbewegungen, die durch Bewegungsbefehle, Fast-Alignment-Routinen, Scanprozeduren oder Funktionsgeneratorausgabe verursacht werden, und die Referenzierungsfahrt.

STP stoppt Makros.

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

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: Für die Achsen der Bewegungsplattform des Hexapods (X, Y, Z, U, V, W) wird der Servomodus immer gemeinsam ein- oder ausgeschaltet. Daher reicht zum Setzen des Servomodus für die Achsen der Bewegungsplattform die Angabe einer einzigen Achse aus, z. B.:
SVO X 1

Der Servomodus wird in folgenden Fällen automatisch eingeschaltet:

- Einschalten oder Neustart des C-887 – Servomodus wird für alle Achsen eingeschaltet
- Für die Achsen der Bewegungsplattform des Hexapods (X, Y, Z, U, V, W): Starten einer Referenzierungsfahrt mit FRF (S. 185)
- Für die Achsen A und B: Zuweisen eines Positionierertyps mit CST (S. 163)

Bewegungen der Achsen können ausschließlich bei eingeschaltetem Servomodus ausgelöst werden.

Der Servomodus kann mit SVO nur ausgeschaltet werden, wenn sich die Achse nicht bewegt (Abfrage mit #5 (S. 153)).

SVO? (Get Servo Mode)

Beschreibung:	Fragt den Servomodus für die angegebenen Achsen ab. Werden keine Argumente angegeben, wird der Servomodus aller Achsen abgefragt.
Format:	SVO? [{<AxisID>}]
Argumente:	<AxisID> ist eine Achse des Controllers.
Antwort:	{<AxisID>"="<ServoState> LF} wobei <ServoState> der aktuelle Servomodus der Achse ist: 0 = Servomodus aus (ungeregelter Betrieb) 1 = Servomodus an (geregelter Betrieb)
Fehlersuche:	Unzulässige Achsenkennung

TAC? (Tell Analog Channels)

Beschreibung:	Fragt die Anzahl installierter Analogleitungen ab.
Format:	TAC?
Argumente:	Keine
Antwort:	<uint> gibt die Gesamtanzahl der Analogleitungen (Ein- und Ausgänge) an.
Hinweis:	Alle in der Antwort auf TAC? enthaltenen Analogleitungen sind analoge Eingangskanäle. Für die Kennungen der analogen Eingangskanäle siehe "Kommandierbare Elemente" (S. 27).

TAD? (Get ADC Value Of Input Signal)

Beschreibung: Fragt den aktuellen Wert des A/D Wandlers des angegebenen Eingangssignalkanals ab. Mit diesem Befehl kann überprüft werden, ob ein Sensorüberlauf vorliegt.

Format: TAD? [{<InputSignalID>}]

Argumente: <InputSignalID> ist ein Eingangssignalkanal des Controllers

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

wobei

<uint> der aktuelle A/D Wert ist, dimensionslos

Hinweise: Die Antwort auf TAD? zeigt den digitalisierten Signalwert ohne Filterung und Linearisierung an.

Je größer das Rauschen des analogen Eingangsignals ist, desto höher sollte mit NAV (S. 247) die Anzahl der Auslesewerte des Analogsignals gesetzt werden, über die der Mittelwert gebildet wird.

Für die Kennungen der analogen Eingangskanäle siehe "Kommandierbare Elemente" (S. 27).

TAV? (Get Analog Input Voltage)

Beschreibung: Fragt die Spannung am Analogeingang ab.

Format: TAV? [{<AnalogInputID>}]

Argumente: <AnalogInputID> ist die Kennung des analogen Eingangskanals; nähere Angaben siehe unten.

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

wobei

<float> die aktuelle Spannung am Analogeingang ist, in Volt

Hinweis: Je größer das Rauschen des analogen Eingangssignals ist, desto höher sollte mit NAV (S. 247) die Anzahl der Auslesewerte des Analogsignals gesetzt werden, über die der Mittelwert gebildet wird.

Für die Kennungen der analogen Eingangskanäle siehe "Kommandierbare Elemente" (S. 27).

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.

Hinweise: Die Stellwege in X, Y, Z, U, V, W sind voneinander abhängig. Je nach der aktuellen Position der Bewegungsplattform des Hexapods kann der tatsächlich verfügbare Stellweg für die Achsen X, Y, Z, U, V und W geringer ausfallen als in der Antwort auf TMN? angegeben. Die Antwort auf TMN? entspricht dem tatsächlich verfügbaren Stellweg einer Achse nur dann, wenn die folgenden Bedingungen erfüllt sind:

- Alle anderen Achsen stehen auf Nullposition.
- Die werkseitig eingestellten Koordinatensysteme sind aktiv.
- Die Standardeinstellungen für die Pivotpunktkoordinaten gelten.

Bei einer Verlegung des Pivotpunkts mit SPI (S. 260) ändern sich die verfügbaren Stellwege für die Rotationsachsen U, V und W. Die Werte für die kleinsten kommandierbaren Positionen werden jedoch **nicht** an die geänderten Stellwege angepasst.

Verwenden Sie VMO? (S. 280), um abzufragen, ob eine Zielposition erreicht werden kann.

Wenn Sie mit benutzerdefinierten Koordinatensystemen arbeiten, den Pivotpunkt verschieben und/oder kombinierte Bewegungen mehrerer Achsen ausführen wollen: Verwenden Sie TRA? (S. 273), um die absolute Position abzufragen, die maximal kommandiert werden kann, wenn sich die Plattform des Hexapods entlang eines vorgegebenen Richtungsvektors bewegt.

Die physikalische Einheit, in der die kleinste kommandierbare Position angegeben wird, kann mit PUN? (S. 253) abgefragt werden.

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.

Hinweise: Die Stellwege in X, Y, Z, U, V, W sind voneinander abhängig. Je nach der aktuellen Position der Bewegungsplattform des Hexapods kann der tatsächlich verfügbare Stellweg für die Achsen X, Y, Z, U, V und W geringer ausfallen als in der Antwort auf TMX? angegeben. Die Antwort auf TMX? entspricht dem tatsächlich verfügbaren Stellweg einer Achse nur dann, wenn die beiden folgenden Bedingungen erfüllt sind:

- Alle anderen Achsen stehen auf Nullposition.
- Die werkseitig eingestellten Koordinatensysteme sind aktiv.
- Die Standardeinstellungen für die Pivotpunktkoordinaten gelten.

Bei einer Verlegung des Pivotpunkts mit SPI (S. 260) ändern sich die verfügbaren Stellwege für die Rotationsachsen U, V und W. Die Werte für die größten kommandierbaren Positionen werden jedoch **nicht** an die geänderten Stellwege angepasst.

Verwenden Sie VMO? (S. 280), um abzufragen, ob eine Zielposition erreicht werden kann.

Wenn Sie mit benutzerdefinierten Koordinatensystemen arbeiten, den Pivotpunkt verschieben und/oder kombinierte Bewegungen mehrerer Achsen ausführen wollen: Verwenden Sie TRA? (S. 273), um die absolute Position abzufragen, die maximal kommandiert werden kann, wenn sich die Plattform des Hexapods entlang eines vorgegebenen Richtungsvektors bewegt.

Die physikalische Einheit, in der die größte kommandierbare Position angegeben wird, kann mit PUN? (S. 253) abgefragt werden.

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:	Die Antwort gibt den Wert des Parameters ***Data Recorder Channel Number*** (ID 0x16000300) an. Weitere Informationen finden Sie unter "Datenrekorder" (S. 99).

TRA? (Get Maximum Commandable Position For Direction Vector)

Beschreibung:	Fragt die absolute Position ab, die maximal kommandiert werden kann, wenn sich die Plattform des Hexapods entlang des Richtungsvektors bewegt, der durch die angegebenen Achsenanteile definiert wird. Die maximal kommandierbare Position wird ausgehend von der aktuellen Position berechnet und kann nur abgefragt werden, wenn sich die Plattform des Hexapods nicht bewegt. Anmerkung: „Maximal“ bezieht sich auf den Betrag des Positionswerts.

Deshalb wird in dieser Beschreibung auch die größtmögliche Auslenkung in negative Richtung als „maximale“ Position bezeichnet (und nicht als „minimale“ Position).

Format: TRA? {<AxisID> <Component>}

Argumente: <AxisID> ist eine Achse des Controllers. Mögliche Werte: X, Y, Z, U, V, W.

<Component> ist der Anteil der Achse am Richtungsvektor. Muss für mindestens eine abgefragte Achse verschieden von null sein. Kann ein negatives Vorzeichen haben.
Achsen, die in der Abfrage nicht angegeben werden, haben keinen Anteil am Richtungsvektor und sind in der Antwort nicht enthalten.

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

wobei

<float> die für die Achse maximal kommandierbare absolute Position ist, wenn sich die Plattform des Hexapods entlang des angegebenen Richtungsvektors bewegt; in physikalischen Einheiten.

Hinweise: Die aktuellen Einstellungen für die Verfahrbereichsgrenzen (siehe NLM (S. 248), PLM (S. 251), SSL (S. 263)) und, falls vom aktiven Betriebs-Koordinatensystem verwendet, für den mit SPI definierten Pivotpunkt (siehe SPI (S. 260)) werden in die Berechnung einbezogen.

Die Antwort auf TRA? zeigt rundungsbedingt möglicherweise eine Position an, die nicht erreicht werden kann. Das Kommandieren einer solchen Position schlägt fehl und erzeugt den Fehlercode 7 ("Position out of limits"). Sie können deshalb die Antwort auf TRA? mit einem Faktor so begrenzen, dass nur Positionen angezeigt werden, die auch tatsächlich kommandiert werden können:

- Setzen Sie mit SPA den Parameter ***Reduction Factor for TRA? Response*** (0x19006000) auf einen geeigneten Wert zwischen 0 und 1.

Die physikalische Einheit, in der die maximal kommandierbare Position angegeben wird, kann mit PUN? (S. 253) abgefragt werden.

Beispiel: Die Bewegungsplattform soll sich in Richtung des Vektors (X, Z) = (2, 4) bewegen. Mit dem Befehl PLM ist die obere Verfahrbereichsgrenze für die Achse X auf den Wert 1 gesetzt.
Senden Sie:
TRA? X 2 Z 4
Die Antwort gibt die maximale absolute Position an, die von der aktuellen Position aus in Richtung des Vektors (X, Z) = (2, 4) angefahren werden kann:
X=1.00000
Z=1.99869

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

TWG? (Get Number of Wave Generators)

Beschreibung:	Fragt die Anzahl der im Controller verfügbaren Funktionsgeneratoren ab.
Format:	TWG?
Argumente:	Keine
Antwort	<uint> ist die Anzahl der verfügbaren Funktionsgeneratoren

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. 135). 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. 135).

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. 161).

VAR? (Get Variable Values)

Beschreibung: Gibt Variablenwerte zurück.

Wird VAR? mit CPY (S. 163), JRC (S. 210), MEX (S. 241) oder WAC (S. 281) kombiniert, muss die Antwort auf VAR? ein einzelner Wert sein (und nicht mehr).

Nähere Angaben zu lokalen und globalen Variablen siehe "Variablen" (S. 135).

Format: VAR? [{<Variable>}]

Argumente: <Variable> ist der Name der abzufragenden Variablen. Nähere Angaben zu Namenskonventionen siehe "Variablen" (S. 135).

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.

Hinweis: Innerhalb eines Makros kann VAR? nur in Kombination mit CPY (S. 163), JRC (S. 210), MEX (S. 241) oder WAC (S. 281) sinnvoll verwendet werden.

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: Der Befehl ist nur für die Achsen A und B zulässig.

<Velocity> muss ≥ 0 sein. Die Obergrenze hängt vom Positionierertyp ab, der den Achsen A und B zugewiesen ist (siehe CST (S. 163)).

Die mit VEL gesetzte Geschwindigkeit wird nur im flüchtigen Speicher (RAM) gespeichert.

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.

VER? (Get Versions Of Firmware And Drivers)

Beschreibung: Fragt die Versionen der Firmware des C-887 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.

Beispiel: Bedeutung der Antwort-Zeilen für den C-887:

`FW: V 2.3.1.1` Version der Firmwarekomponente FW

`Macro: V 0.15.0.0` Version der Makrofunktionalität

`OS: V #1 SMP PREEMPT RT Fri Jan 22 12:54:58 CET 2016` Version des Betriebssystems des C-887

`Hexdata: V 1.0.0.0` Version der Konfigurationsdatei mit den Geometriedaten für den Hexapod, auf den der C-887 abgestimmt ist

`PIStages: V 200.7 19/10/2017 15:1:28` Version der auf dem C-887 gespeicherten Positioniererdatenbank PIStages2.dat

`FPGA/DSP:  V 0.28 V 0.97` Version der Motoransteuerung

`ADC:  C887E0015 SN116054450 V3 11/10/2016` Version des A/D-Wandlers für die analogen Eingänge 5 und 6; nur Modelle C-887.521, .523, .531, .533

`C-887.MC: V 256 SN 118015195` Version der manuellen Bedieneinheit; nur wenn eine Bedieneinheit am C-887 angeschlossen ist

`IDChip: H-811.I2 SN117007310 21/2/2017` Auf dem ID-Chip des Hexapods gespeicherte Angaben für Typ, Seriennummer und Herstelldatum des Hexapods

`Collision: V 2.15` Version der Bibliothek für die kostenpflichtige PIVeriMove Hexapod Software zur Kollisionsprüfung; wird auch angezeigt, wenn die Kollisionsprüfung nicht freigeschaltet ist

VLS (Set System Velocity)

Beschreibung: Setzt die Geschwindigkeit für die Bewegungsplattform des Hexapods.

Format: VLS <SystemVelocity>

Argumente: <SystemVelocity> ist der Geschwindigkeitswert in physikalischen Einheiten.

Antwort: Keine

Hinweise: Die Einheit für <SystemVelocity> ist mm/s.

Die Untergrenze für <SystemVelocity> ist durch die kleinste Schrittweite des Hexapods bedingt (modellabhängig) und wird durch den Wert des Parameters ***Minimum System Velocity*** (ID 0x19001501) vorgegeben. Der Parameter kann mit SPA? (S. 259) abgefragt werden.

Die Geschwindigkeit kann mit VLS nur gesetzt werden, wenn sich der Hexapod nicht bewegt (Achsen X, Y, Z, U, V, W; Abfrage mit #5 (S. 153)).

Für die Achsen A und B kann die Geschwindigkeit mit VEL (S. 277) gesetzt werden.

VLS? (Get System Velocity)

Beschreibung:	Fragt die Geschwindigkeit der Bewegungsplattform des Hexapods ab, die mit VLS (S. 279) gesetzt ist.
Format:	VLS?
Argumente:	Keine
Antwort:	<SystemVelocity> ist der Geschwindigkeitswert in physikalischen Einheiten, siehe VLS.

VMO? (Virtual Move)

Beschreibung:	Prüft, ob die Bewegungsplattform des Hexapods eine vorgegebene Position von der aktuellen Position aus anfahren kann. VMO? löst keine Bewegung aus.
Format:	VMO? {<AxisID> <Position>}
Argumente:	<AxisID> ist eine Achse des Controllers, siehe unten <Position> ist ein zu prüfender Zielpositionswert
Antwort:	<uint> gibt an, ob die Bewegungsplattform die aus den angegebenen Zielpositionswerten resultierende Position anfahren kann: 0 = vorgegebene Position kann nicht angefahren werden 1 = vorgegebene Position kann angefahren werden
Hinweise:	Zulässig sind die Achsen X, Y, Z, U, V, W. VMO? prüft Folgendes: ▪ Liegen die Stützstellen des berechneten Profils und die Zielposition außerhalb der Stellwegsgrenzen, die mit TMN? (S. 271) und TMX? (S. 272) oder TRA? (S. 273) abgefragt werden können? ▪ Sind die mit NLM (S. 248) und PLM (S. 251) gesetzten Verfahrbereichsgrenzen mit SSL (S. 263) aktiviert, und wenn ja, liegen die Stützstellen und die Zielposition außerhalb dieser Verfahrbereichsgrenzen? ▪ Sind die einzelnen Beine in der Lage, die Plattform zu den notwendigen Stützstellen und zur vorgegebenen Zielposition zu bewegen? ▪ Wenn eine Konfiguration zur Kollisionsvermeidung auf dem C-887 abgelegt wurde (z. B. mit der kostenpflichtigen PIVeriMove Hexapod Software zur Kollisionsprüfung): Treten Kollisionen auf zwischen den folgenden Gruppen?

- Umgebung inkl. Grundplatte des Hexapods
- Hexapod-Beine
- Bewegungsplattform des Hexapods inkl. Last

Um eine zuverlässige Antwort zu erhalten, senden Sie VMO? erst nach einer erfolgreichen Referenzierungsfahrt (Start mit FRF (S. 185)) und nur dann, wenn sich der Hexapod nicht bewegt (Abfrage mit #5) (S. 153).

VST? (Get Connectable Stages)

Beschreibung:	Listet Positionierertypen auf, die an den C-887 angeschlossen werden können.
Format:	VST?
Argumente:	Keine
Antwort:	<string> ist eine Liste aller Positionierertypen, die in den verfügbaren Positioniererdatenbanken enthalten sind und mit dem C-887 verwendet werden können.
Hinweise:	Weitere Informationen zu Positioniererdatenbanken finden Sie in "Betriebsparameter der Achsen A und B" (S. 32). Die mit VST? aufgelisteten Positionierertypen können mit dem Befehl CST (S. 163) den Achsen A und B zugewiesen werden.

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. 241).
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.

WAV (Set Waveform Definition)

Beschreibung: Definiert eine Kurvenform des angegebenen Typs für die angegebene Kurventabelle.

Um eine flexible Definition zu ermöglichen, kann eine Kurvenform (Kurventabelleninhalt) durch Aneinanderreihen von "Segmenten" aufgebaut werden. Jedes Segment wird mit einem separaten WAV-Befehl definiert. Mit dem Argument <AppendWave> (siehe unten) wird ein Segment an den bestehenden Kurventabelleninhalt angehängt. Um einzelne Segmente oder deren Reihenfolge zu ändern, muss die gesamte Kurvenform Segment für Segment neu erstellt werden.

Ein Segment kann auf vordefinierten Kurvenformen basieren (siehe Argument <WaveType> unten).

Kurvenformen können nicht geändert werden, während sie vom Funktionsgenerator ausgegeben werden. Vor dem Ändern einer Kurvenform mit WAV muss die Funktionsgeneratorausgabe aus der zugehörigen Kurventabelle gestoppt werden.

Die Kurvenformwerte sind absolute Werte.

Die Dauer eines Ausgabezyklus für die Kurvenform lässt sich folgendermaßen berechnen:

Ausgabedauer = Servozykluszeit * WTR-Wert * Anzahl der Punkte

wobei

die Servozykluszeit für den C-887 durch den Parameter 0x0E000200 angegeben wird (in Sekunden)

der WTR-Wert (Ausgaberate des Funktionsgenerators) die Anzahl der Servozyklen angibt, über die sich die Ausgabe eines Kurvenpunkts zeitlich erstreckt; ganzzahliges Vielfaches von 10 (Minimum und Standard: 10, Maximum: 1000)

die Anzahl der Punkte der Länge der Kurventabelle entspricht (Summe der Längen aller Segmente in dieser Tabelle)

Weitere Informationen siehe "Funktionsgenerator" (S. 102).

Format: WAV <WaveTableID> <AppendWave> <WaveType> <WaveTypeParameters>

Argumente: <WaveTableID> ist die Kurventabellenkennung.

<AppendWave> kann „X“ oder „&“ sein:
„X“ löscht die Kurventabelle und beginnt am ersten Punkt der Tabelle zu schreiben.
„&“ hängt das definierte Segment an den vorhandenen Kurventabelleninhalt an, um die Kurvenform zu verlängern.

<WaveType>: Der Typ der zum Definieren des Segments verwendeten Kurve. Dies kann einer der folgenden Kurventypen sein:
„PNT“ (benutzerdefinierte Kurve)
„SIN_P“ (invertierte Kosinuskurve)
„RAMP“ (Rampenkurve)
„LIN“ (Kurve in Form einer einzelnen Abtastzeile)

<WaveTypeParameters> steht für die Parameter der Kurve:

Für „PNT“:

<WaveStartPoint> <WaveLength> {<WavePoint>}

<WaveStartPoint>: Der Index des Startpunkts. Muss 1 sein.

<WaveLength>: Die Anzahl der in die Kurventabelle zu schreibenden Punkte (= Segmentlänge).

<WavePoint>: Der Wert eines einzelnen Punktes.

Für „SIN_P“:

<SegLength> <Amp> <Offset> <WaveLength> <StartPoint> <CurveCenterPoint>

<SegLength>: Die Länge des Kurventabellensegments in Punkten. Nur die durch <SegLength> angegebene Anzahl von Punkten wird in die Kurventabelle geschrieben. Ist der Wert von <SegLength> größer als der Wert von <WaveLength>, werden die fehlenden Punkte im Segment mit dem Endpunktwert der Kurve aufgefüllt.

<Amp>: Die Amplitude der Sinuskurve.

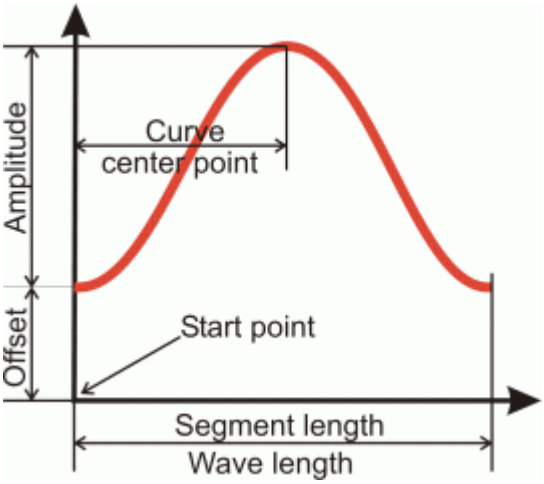
<Offset>: Der Offset der Sinuskurve.

<WaveLength>: Die Länge der Sinuskurve in Punkten.

<StartPoint>: Der Index des Startpunkts der Sinuskurve im Segment. Gibt die Phasenverschiebung an. Der kleinstmögliche Wert ist 0.

<CurveCenterPoint>: Der Index des Mittelpunkts der Sinuskurve. Bestimmt, ob die Kurve symmetrisch ist oder nicht. Der kleinstmögliche Wert ist 0.

Beispiel (weitere Beispiele siehe "Kurvenform definieren" (S. 105)):



Für „RAMP“:

<SegLength> <Amp> <Offset> <WaveLength> <StartPoint> <SpeedUpDown> <CurveCenterPoint>

<SegLength>: Die Länge des Kurventabellensegments in Punkten. Nur die durch <SegLength> angegebene Anzahl von Punkten wird in die Kurventabelle geschrieben. Ist der Wert von <SegLength> größer als der Wert von <WaveLength>, werden die fehlenden Punkte im Segment mit dem Endpunktwert der Kurve aufgefüllt.

<Amp>: Die Amplitude der Rampenkurve.

<Offset>: Der Offset der Rampenkurve.

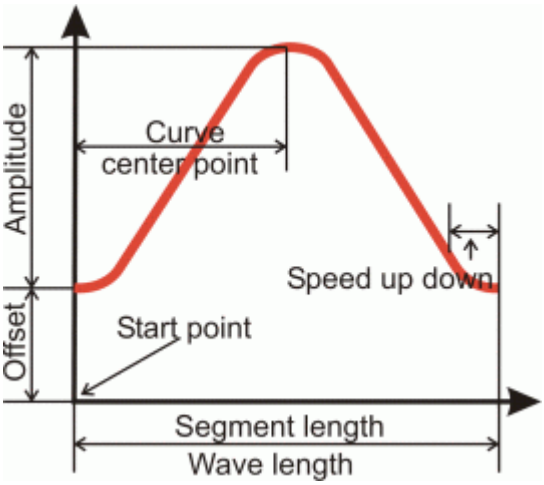
<WaveLength>: Die Länge der Rampenkurve in Punkten.

<StartPoint>: Der Index des Startpunkts der Rampenkurve im Segment. Gibt die Phasenverschiebung an. Der kleinstmögliche Wert ist 0.

<SpeedUpDown>: Die Anzahl der Punkte für Beschleunigung und Verzögerung.

<CurveCenterPoint>: Der Index des Mittelpunkts der Rampenkurve. Bestimmt, ob die Kurve symmetrisch ist oder nicht. Der kleinstmögliche Wert ist 0.

Beispiel (weitere Beispiele siehe "Kurvenform definieren" (S. 105)):



Für „LIN“:

<SegLength> <Amp> <Offset> <WaveLength> <StartPoint> <SpeedUpDown>

<SegLength>: Die Länge des Kurventabellensegments in Punkten. Nur die durch <SegLength> angegebene Anzahl von Punkten wird in die Kurventabelle geschrieben. Ist der Wert von <SegLength> größer als der Wert von <WaveLength>, werden die fehlenden Punkte im Segment mit dem Endpunktwert der Kurve aufgefüllt.

<Amp>: Die Amplitude der Abtastzeile.

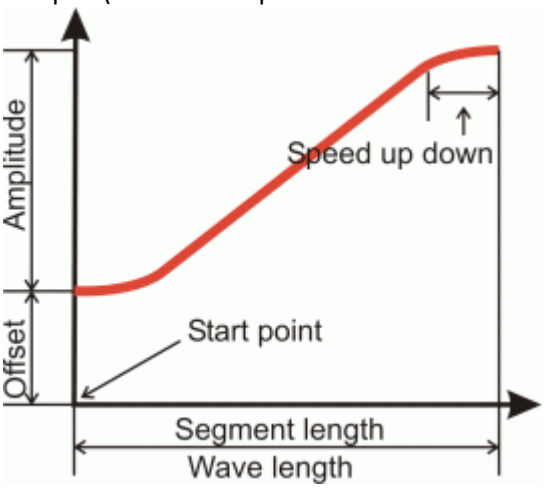
<Offset>: Der Offset der Abtastzeile.

<WaveLength>: Die Länge der Kurve (einzelne Abtastzeile) in Punkten.

<StartPoint>: Der Index des Startpunkts der Abtastzeile im Segment. Der kleinstmögliche Wert ist 0.

<SpeedUpDown>: Die Anzahl der Punkte für Beschleunigung und Verzögerung.

Beispiel (weitere Beispiele siehe "Kurvenform definieren" (S. 105)):



Antwort: Keine

Fehlersuche: Ungültige Kurventabellenkennung

Die Gesamtanzahl der Punkte für die Kurvenform (die aus mehreren Segmenten bestehen kann) überschreitet die Anzahl der verfügbaren Speicherpunkte.

Hinweise: Bei der Definition einer Kurvenform prüft der C-887 **nicht** die Zielpositionen und die resultierenden Geschwindigkeiten. Die Prüfung erfolgt erst während der Funktionsgeneratorausgabe (S. 115). Wenn die Bewegung nicht möglich ist, stoppt der C-887 die Funktionsgeneratorausgabe und damit die Bewegung abrupt.

Um mit WAV geeignete Kurvenformen zu definieren und einen Abbruch der Funktionsgeneratorausgabe zu vermeiden, beachten Sie Folgendes:

- Berechnen Sie die Kurvenform extern möglichst exakt (z. B. mit NI LabVIEW, MATLAB oder Python), bevor Sie die Kurvenformdefinition mit WAV in den C-887 übertragen.
- Beachten Sie, dass die Kurvenform die Geschwindigkeit während der Bewegungen beeinflusst. Definieren Sie die Kurvenform so, dass die Spezifikationen der angeschlossenen Mechanik während der Funktionsgeneratorausgabe eingehalten werden. Die Geschwindigkeit wird unter anderem durch folgende Faktoren begrenzt:
  - Typ der Mechanik
  - Kombination der zu bewegenden Achsen
  - Aktuelle Einstellungen für Koordinatensystem und Drehpunkt
  - Amplitude der Bewegung
- Wenn die Kurvenform periodisch ausgegeben werden soll (d. h., mehr als ein Ausgabezyklus in Folge), muss das Ende der Kurvenform in Position und Geschwindigkeit identisch sein mit dem Anfang der Kurvenform.
- Wenn Sie eine Kurvenform aus mehreren Segmenten zusammensetzen, muss die Anfangsposition und Anfangsgeschwindigkeit eines anzuhängenden Segments jeweils an die Endposition und Endgeschwindigkeit des Vorgängersegments angepasst sein.
- Definieren Sie die Kurvenform möglichst so, dass sie mit der Standardausgaberate ausgegeben werden kann. Bei größeren Ausgaberaten interpoliert der C-887 fehlende Positionswerte, wodurch die ausgegebene Kurvenform möglicherweise nicht mehr exakt der Definition entspricht.

WAV? (Get Waveform Definition)

Beschreibung: Fragt den Wert eines Kurvenparameters für eine angegebene Kurventabelle ab.

Weitere Informationen siehe "Funktionsgenerator" (S. 102).

Format: WAV? [{<WaveTableID> <WaveParameterID>}]

Argumente: <WaveTableID> ist die Kurventabellenkennung.

<WaveParameterID> ist die Kurvenparameter-ID:
1 = aktuelle Kurventabellenlänge als Anzahl der Punkte

Antwort: {<WaveTableID> <WaveParameterID>"="<float> LF}

wobei

<float> von <WaveParameterID> abhängt; gibt die aktuelle Anzahl der Kurvenformpunkte in der Kurventabelle für <WaveParameterID> = 1 an

Fehlersuche: Ungültige Kurventabellenkennung

WCL (Clear Wave Table Data)

Beschreibung: Löscht den Inhalt der angegebenen Kurventabelle.

Solange ein Funktionsgenerator aktiv ist, ist es nicht möglich, die damit verbundene Kurventabelle zu löschen.

Weitere Informationen siehe "Funktionsgenerator" (S. 102).

Format: WCL {<WaveTableID>}

Argumente: <WaveTableID> ist die Kurventabellenkennung.

Antwort: Keine

WGC (Set Number Of Wave Generator Cycles)

Beschreibung: Setzt die Anzahl der Ausgabezyklen für den angegebenen Funktionsgenerator (die Ausgabe selbst wird mit WGO (S. 289) gestartet).

Weitere Informationen siehe "Funktionsgenerator" (S. 102).

Format: WGC {<WaveGenID> <Cycles>}

Argumente: <WaveGenID> ist die Funktionsgeneratorkennung

<Cycles> ist die Anzahl der Funktionsgenerator-Ausgabezyklen.

Antwort: Keine

Hinweise: Mit <WaveGenID> = 0 werden alle Funktionsgeneratoren addressiert.

Wenn Cycles = 0, erfolgt die Ausgabe der Kurvenform ohne Begrenzung, bis sie mit WGO oder #24 (S. 156) oder STP (S. 267) oder HLT (S. 204) gestoppt wird.

WGC? (Get Number Of Wave Generator Cycles)

Beschreibung: Fragt die Anzahl der Ausgabezyklen ab, die für den angegebenen Funktionsgenerator gesetzt sind.

Weitere Informationen siehe "Funktionsgenerator" (S. 102).

Format: WGC? [{<WaveGenID>}]

Argumente: <WaveGenID> ist die Funktionsgeneratorkennung

Antwort: {<WaveGenID>"="<Cycles> LF}

wobei

<Cycles> die Anzahl der Funktionsgenerator-Ausgabezyklen ist, die mit WGC (S. 288) gesetzt sind.

WGO (Set Wave Generator Start/Stop Mode)

Beschreibung: Startet und stoppt den angegebenen Funktionsgenerator. Beim Start der Funktionsgeneratorausgabe wird automatisch ein Datenaufzeichnungszyklus gestartet.

Alle Funktionsgeneratoren, deren Ausgabe gleichzeitig aktiv sein soll, müssen im selben Befehl gestartet werden.

Für die Achsen der Bewegungsplattform des Hexapods, deren Funktionsgenerator **nicht** gestartet wurde, wird immer die zuletzt gültige Zielposition kommandiert.

Die Anzahl der Ausgabezyklen kann durch WGC (S. 288) begrenzt werden.

Mit WTR (S. 296) können Sie die einzelnen Ausgabezyklen der Kurvenform verlängern.

Die Funktionsgeneratorausgabe wird auch fortgesetzt, wenn die PC-Software, mit der sie gestartet wurde, beendet wird.

Der Befehl #9 kann zum Abfragen des aktuellen Aktivierungszustands der Funktionsgeneratoren verwendet werden. Mit WGO? können die zuletzt kommandierten Startoptionen für den Funktionsgenerator abgefragt werden. Mit WGS? (S. 292) können weitere Statusinformationen über den Funktionsgenerator abgefragt werden.

Weitere Informationen siehe "Funktionsgenerator" (S. 102).

Format: WGO {<WaveGenID> <StartMode>}

Argumente: <WaveGenID> ist die Funktionsgeneratorkennung

<StartMode> ist der Startmodus für den angegebenen Funktionsgenerator, kann im Hexadezimal- oder Dezimalformat angegeben werden. Mögliche Werte:

0: Funktionsgeneratorausgabe wird gestoppt. Sie können die Funktionsgeneratorausgabe auch mit #24 (S. 156) oder STP (S. 267) oder HLT (S. 204) stoppen.

Bit 0 = 0x1 (Hexadezimalformat) oder 1 (Dezimalformat): sofortiger Start der Funktionsgeneratorausgabe, synchronisiert durch Servozyklus

Antwort: Keine

Fehlersuche: Ungültige Funktionsgeneratorkennung

Mit dem Funktionsgenerator ist keine Kurventabelle verbunden. Verbinden Sie mit WSL (S. 295) eine Kurventabelle.

Wenn die Funktionsgeneratorausgabe aktiv ist, sind Befehle zum Starten oder Konfigurieren von Bewegungen sowie das Ausführen entsprechender Makros **nicht** zulässig.

Während der Funktionsgeneratorausgabe prüft der C-887 ständig, ob die Bewegung noch möglich ist. In folgenden Fällen stoppt der C-887 die Bewegung abrupt und setzt einen Fehlercode:

- Die auszugebenden Zielpositionen können nicht erreicht werden.
- Die erforderliche Geschwindigkeit kann nicht erreicht werden.
- Die Bewegung würde eine Kollision verursachen.

➢ Fragen Sie mit dem Befehl `WGS?` den aktuellen Status der Funktionsgeneratoren ab, insbesondere den Index der Kurvenformpunkte, an denen ein Fehler aufgetreten ist.

➢ Fragen Sie mit dem Befehl `ERR?` (S. 176) den Fehlercode des zuletzt aufgetretenen Fehlers ab.

WGO? (Get Wave Generator Start/Stop Mode)

Beschreibung: Fragt den Start-/Stoppmodus des angegebenen Funktionsgenerators ab.

Weitere Informationen siehe "Funktionsgenerator" (S. 102).

Format: WGO? [{<WaveGenID>}]

Argumente: <WaveGenID> ist die Funktionsgeneratorkennung

Antwort: {<WaveGenID>"="<StartMode> LF}

wobei

<StartMode> der zuletzt kommandierte Startmodus des Funktionsgenerators im Dezimalformat ist.Weitere Informationen siehe WGO (S. 289).

Hinweise: <StartMode> ist nicht nur 0, wenn `WGO <WaveGenID> 0` gesendet wurde, sondern auch, wenn die Funktionsgeneratorausgabe geendet hat oder mit #24 (S. 156) oder STP oder HLT (S. 204) gestoppt wurde.

WGR (Starts Recording In Sync With Wave Generator)

Beschreibung: Startet die Datenaufzeichnung, wenn der Funktionsgenerator aktiv ist.

Weitere Informationen siehe "Funktionsgenerator" (S. 102) und "Datenrekorder" (S. 99).

Format: WGR

Argumente: Keine

Antwort: Keine

Hinweise: Nach dem Senden von WGR startet die Datenaufzeichnung mit dem nächsten Ausgabezyklus des Funktionsgenerators.

Die aufgezeichneten Daten können mit DRR? (S. 172) gelesen werden.

Das Starten der Funktionsgeneratorausgabe mit WGO (S. 289) startet gleichzeitig einen ersten Datenaufzeichnungszyklus.

Weitere Triggeroptionen zum Starten der Datenaufzeichnung siehe DRT (S. 175).

WGS? (Get Status Information On Wave Generator)

Beschreibung: Fragt Statusinformationen über den angegebenen Funktionsgenerator ab.

Weitere Informationen siehe "Funktionsgenerator" (S. 102).

Format: WGS? [<WaveGenID> [<ItemID>]]

Argumente: <WaveGenID> ist die Funktionsgeneratorkennung

<ItemID> ist die Kennung einer abzufragenden Größe. Mögliche Kennungen:

STATUS
Fragt den Status des Funktionsgenerators ab.
Mögliche Antwortwerte, in hexadezimalem Format:
0x0 = Funktionsgenerator nicht aktiv (Ausgabe läuft nicht)
0x1 = Funktionsgenerator aktiv (Ausgabe läuft)

ITERATIONS
Fragt die Anzahl der Ausgabezyklen seit dem letzten Start des Funktionsgenerators ab.
Das Stoppen des Funktionsgenerators hält den Zähler an. Das Zurücksetzen des Zählers erfolgt beim erneuten Starten des Funktionsgenerators mit dem Befehl WGO.

ERRORTYPE
Fragt den Fehlercode des zuletzt während der Ausgabe aufgetretenen Fehlers ab.
Mögliche Antwortwerte:
0 = No error occurred
7 = Position out of limits
8 = Velocity out of limits
91 = Move not possible, would cause collision

ERRORINDEX
Fragt den Index des Kurvenformpunktes ab, an dem der Fehler aufgetreten ist.

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

wobei

<Value> den aktuellen Wert der abgefragten Größe angibt.

WMS? (Get Maximum Number of Wave Table Points)

Beschreibung: Fragt die Anzahl der verfügbaren Speicherpunkte für die angegebene Kurventabelle ab.

Format: WMS? [{<WaveTableID>}]

Argumente: <WaveTableID> ist die Kurventabellenkennung.

Antwort: {<WaveTableID>"="<NumberOfPoints> LF}

wobei

<NumberOfPoints> ist die Anzahl der Speicherpunkte, die für die Kurventabelle verfügbar sind (Summe der bereits mit WAV (S. 282) definierten Punkte und der noch ungenutzten Punkte).

WPA (Save Parameters To Non-Volatile Memory)

Beschreibung: Schreibt die aktuellen Einstellungen aus dem flüchtigen in den permanenten Speicher.

Die mit WPA gespeicherten Einstellungen werden beim Einschalten oder Neustart des C-887 automatisch aus dem permanenten Speicher in den flüchtigen Speicher geladen.

Hinweis: Fehlerhafte Einstellungen können zur Fehlfunktion des Systems führen. Vergewissern Sie sich, dass die aktuellen Einstellungen korrekt sind, bevor Sie den Befehl WPA ausführen.

Einstellungen im flüchtigen Speicher, die nicht mit WPA gespeichert wurden, gehen verloren, wenn der C-887 ausgeschaltet oder neugestartet wird bzw. wenn Einstellungen zurückgesetzt werden.

Format: WPA <Pswd> [{<ItemID> <PamID>}]

Argumente: <Pswd> ist das Kennwort zum Schreiben in den permanenten Speicher. Nähere Angaben siehe unten.

<ItemID> ist das Element, für das ein Parameter aus dem flüchtigen Speicher im permanenten Speicher gespeichert werden soll. Nähere Angaben siehe unten.

<PamID> ist die Parameterkennung, kann im Hexadezimal- oder Dezimalformat geschrieben werden. Nähere Angaben siehe unten.

Antwort: Keine

Fehlersuche: Unzulässige Elementkennung, falsche Parameter-ID, ungültiges Kennwort

Hinweise: Wenn Parametereinstellungen gespeichert werden sollen:

- Parameterwerte können im flüchtigen Speicher mit dem Befehl SPA (S. 258) geändert werden.
- Ein Element kann eine Achse, ein Hexapod-Bein, ein Eingangssignalkanal oder das gesamte System sein. Der

Elementtyp hängt vom Parameter ab. Weitere Informationen siehe "Anpassen von Einstellungen" (S. 317).

- Mit der Abfrage HPA? (S. 204) erhalten Sie eine Liste aller verfügbaren Parameter. Gültige Parameter-IDs finden Sie auch in der Parameterübersicht (S. 320).

Neben den Einstellungen der Parameter können mit WPA die Einstellungen für Koordinatensysteme in den permanenten Speicher geschrieben werden (Details siehe untenstehende Tabelle).

Das Speichern mit WPA überschreibt **nicht** die Werkseinstellungen, die mit DPA (S. 167) wiederhergestellt werden können.

Hinweis: Vermeiden Sie es, den C-887 während der WPA-Prozedur auszuschalten.

Gültige Kennwörter zum Schreiben in den permanenten Speicher:

100	Speichert die aktuell gültigen Werte aller Parameter, die aktuell gültigen Einstellungen für Koordinatensysteme (Details siehe Kennwort SKS) und die aktuelle Zuweisung von Positionierertypen zu den Achsen A und B
101	Speichert die aktuell gültigen Parameterwerte. Die Angabe von <ItemID> und <PamID> ist optional.
SKS	Die Angabe von <ItemID> und <PamID> entfällt bei Verwendung des Kennworts SKS. Speichert die aktuell gültigen Einstellungen für Koordinatensysteme: ▪ Eigenschaften der im flüchtigen Speicher vorhandenen Koordinatensysteme und Kombinationen von Koordinatensystemen, siehe KLS? und KLC? ▪ Aktivierungszustand von Koordinatensystemen, siehe KEN ▪ Verkettung von Koordinatensystemen, siehe KLN Wenn ZERO aktiv ist: Die aktuellen Werte für NLM, PLM, SSL, SPI und SST werden nicht gespeichert. Damit wird sichergestellt, dass KEN ZERO die Werkseinstellungen für das Betriebs-Koordinatensystem wieder im vollen Umfang aktiviert.
A12	Zuweisung von Positionierertypen zu den Achsen A und B

Gültige Kennwörter können mit `MAN? WPA` abgefragt werden.

WSL (Set Connection Of Wave Table To Wave Generator)

Beschreibung: Auswahl der Kurventabelle: Verbindet eine Kurventabelle mit einem Funktionsgenerator oder trennt die Verbindung des ausgewählten Generators zu einer Kurventabelle.

Zwei oder mehr Generatoren können mit derselben Kurventabelle verbunden sein, ein Generator kann jedoch nicht mit mehreren Kurventabellen verbunden sein.

Das Löschen des Kurventabelleninhalts mit WCL (S. 288) hat keine Auswirkung auf die WSL-Einstellungen.

Solange ein Funktionsgenerator aktiv ist, ist es nicht möglich, seine Kurventabellenverbindung zu ändern.

Weitere Informationen siehe "Funktionsgenerator" (S. 102).

Format: WSL {<WaveGenID> <WaveTableID>}

Argumente: <WaveGenID> ist die Funktionsgeneratorkennung

<WaveTableID> ist die Kurventabellenkennung. Wenn <WaveTableID> = 0, wird die Verbindung des ausgewählten Generators zu einer Kurventabelle getrennt.

Antwort: Keine

WSL? (Get Connection Of Wave Table To Wave Generator)

Beschreibung: Fragt die aktuellen Einstellungen der Kurventabellenverbindung für den angegebenen Funktionsgenerator ab.

Weitere Informationen siehe "Funktionsgenerator" (S. 102).

Format: WSL? [{<WaveGenID>}]

Argumente: <WaveGenID> ist die Funktionsgeneratorkennung

Antwort: {<WaveGenID>"="<WaveTableID> LF}

wobei

<WaveTableID> die Kurventabellenkennung ist. Wenn <WaveTableID> = 0, ist keine Kurventabelle mit dem Funktionsgenerator verbunden.

WTR (Set Wave Generator Table Rate)

Beschreibung:	Setzt die Ausgaberate des Funktionsgenerators und den Interpolationstyp.
Format:	WTR {<WaveGenID> <WaveTableRate> <InterpolationType>}
Argumente:	<WaveGenID> ist die Funktionsgeneratorkennung. Nähere Angaben siehe unten. <WaveTableRate> ist Ausgaberate des Funktionsgenerators (Einheit: Anzahl der Servozyklen); muss ein Ganzzahlwert sein, der größer als null ist <InterpolationType> Verfügbare Interpolationstypen siehe unten.
Antwort:	Keine
Hinweise:	Für die einzelnen Funktionsgeneratoren des C-887 können unterschiedliche Ausgaberaten eingestellt werden. Die Ausgaberate wird für alle Funktionsgeneratoren auf denselben Wert gesetzt, wenn <WaveGenID> den Wert null hat. Mit WTR können die einzelnen Ausgabezyklen der Kurvenform verlängert werden. Die Dauer eines Ausgabezyklus für die Kurvenform kann wie folgt berechnet werden: Ausgabedauer = Servozykluszeit * WTR-Wert * Anzahl der Punkte wobei die Servozykluszeit für den C-887 durch den Parameter 0x0E000200 angegeben wird (in Sekunden) der WTR-Wert (Ausgaberate des Funktionsgenerators) die Anzahl der Servozyklen angibt, über die sich die Ausgabe eines Kurvenpunkts zeitlich erstreckt; ganzzahliges Vielfaches von 10 (Minimum und Standard: 10, Maximum: 1000) die Anzahl der Punkte der Länge der Kurvenform (d. h. der Länge der Kurventabelle) entspricht WTR setzt auch den Interpolationstyp, der für die Funktionsgeneratorausgabe mit einer Ausgaberate > 10 verwendet wird. Anmerkung: Durch die lineare Interpolation können zwar Positionssprünge vermieden werden, aber die Kurvenform ist dann an den Stützstellen nicht mehr differenzierbar. Daraus können größere Geschwindigkeits- und Beschleunigungssprünge resultieren, die

Schwingungen der Mechanik anregen. Verwenden Sie deshalb möglichst die Standardausgaberate (10).

Weitere Informationen siehe "Funktionsgenerator" (S. 102). Ein Anwendungsbeispiel finden Sie unter "Funktionsgenerator konfigurieren" (S. 113).

Verfügbare Interpolationstypen:

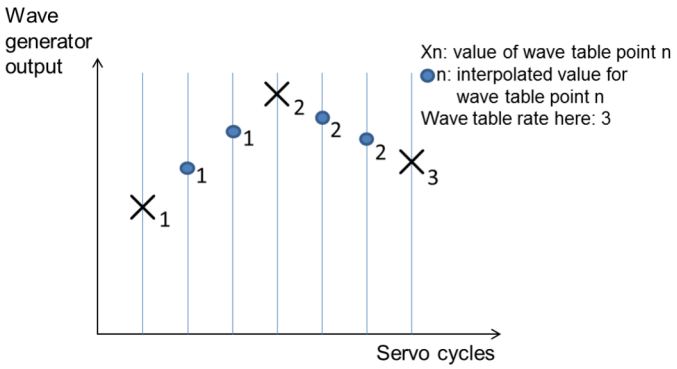
Folgende Interpolationstypen stehen zur Verfügung:
0 = keine Interpolation
1 = Gerade (Standard)

Beispiele:

Beachten Sie: Der C-887 unterstützt **nicht** die in den folgenden Beispielen verwendete Ausgaberate (3). Die Beispiele sollen lediglich das Prinzip der Interpolation zeigen.

Interpolationstyp: Gerade (1; Standard)
Die Ausgaberate für Funktionsgenerator 5 wird auf 3 gesetzt, mit linearer Interpolation:

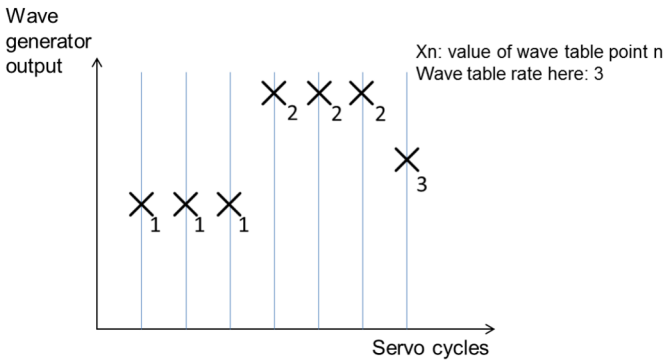
WTR 5 3 1



Interpolationstyp: keine Interpolation (0)

Die Ausgaberate für Funktionsgenerator 5 wird auf 3 gesetzt, ohne Interpolation:

WTR 5 3 0



WTR? (Get Wave Generator Table Rate)

Beschreibung: Fragt die aktuelle Ausgaberate des Funktionsgenerators und den verwendeten Interpolationstyp ab.

Weitere Informationen siehe "Funktionsgenerator" (S. 102). Ein Anwendungsbeispiel finden Sie unter "Funktionsgenerator konfigurieren" (S. 113).

Format: WTR? [{<WaveGenID>}]

Argumente: <WaveGenID> ist die Funktionsgeneratorkennung

Antwort: {<WaveGenID>"="<WaveTableRate> <InterpolationType> LF}

wobei

<WaveTableRate> die Ausgaberate des Funktionsgenerators ist (Einheit: Anzahl der Servozyklen)

<InterpolationType> der Interpolationstyp ist, der auf Ausgaben zwischen Kurventabellenpunkten angewendet wird, wenn die Ausgaberate größer als der Mindestwert ist. Verfügbare Interpolationstypen siehe WTR (S. 296).

8.5 Fehlercodes

Die hier aufgelisteten Fehlercodes sind Bestandteil des PI General Command Set. Einige der Fehlercodes sind für Ihren Controller möglicherweise nicht relevant und werden daher nie ausgegeben.

Controllerfehler

0	PI_CNTR_NO_ERROR	No error
1	PI_CNTR_PARAM_SYNTAX	Parameter syntax error
2	PI_CNTR_UNKNOWN_COMMAND	Unknown command
3	PI_CNTR_COMMAND_TOO_LONG	Command length out of limits or command buffer overrun
4	PI_CNTR_SCAN_ERROR	Error while scanning
5	PI_CNTR_MOVE_WITHOUT_REF_OR_NO_SERVO	Unallowable move attempted on unreferenced axis, or move attempted with servo off
6	PI_CNTR_INVALID_SGA_PARAM	Parameter for SGA not valid
7	PI_CNTR_POS_OUT_OF_LIMITS	Position out of limits
8	PI_CNTR_VEL_OUT_OF_LIMITS	Velocity out of limits
9	PI_CNTR_SET_PIVOT_NOT_POSSIBLE	Attempt to set pivot point while U,V and W not all 0
10	PI_CNTR_STOP	Controller was stopped by command
11	PI_CNTR_SST_OR_SCAN_RANGE	Parameter for SST or for one of the embedded scan algorithms out of range
12	PI_CNTR_INVALID_SCAN_AXES	Invalid axis combination for fast scan
13	PI_CNTR_INVALID_NAV_PARAM	Parameter for NAV out of range
14	PI_CNTR_INVALID_ANALOG_INPUT	Invalid analog channel
15	PI_CNTR_INVALID_AXIS_IDENTIFIER	Invalid axis identifier
16	PI_CNTR_INVALID_STAGE_NAME	Unknown stage name
17	PI_CNTR_PARAM_OUT_OF_RANGE	Parameter out of range
18	PI_CNTR_INVALID_MACRO_NAME	Invalid macro name
19	PI_CNTR_MACRO_RECORD	Error while recording macro
20	PI_CNTR_MACRO_NOT_FOUND	Macro not found
21	PI_CNTR_AXIS_HAS_NO_BRAKE	Axis has no brake
22	PI_CNTR_DOUBLE_AXIS	Axis identifier specified more than once
23	PI_CNTR_ILLEGAL_AXIS	Illegal axis
24	PI_CNTR_PARAM_NR	Incorrect number of parameters
25	PI_CNTR_INVALID_REAL_NR	Invalid floating point number
26	PI_CNTR_MISSING_PARAM	Parameter missing
27	PI_CNTR_SOFT_LIMIT_OUT_OF_RANGE	Soft limit out of range
28	PI_CNTR_NO_MANUAL_PAD	No manual pad found
29	PI_CNTR_NO_JUMP	No more step-response values
30	PI_CNTR_INVALID_JUMP	No step-response values recorded

31	PI_CNTR_AXIS_HAS_NO_REFERENCE	Axis has no reference sensor
32	PI_CNTR_STAGE_HAS_NO_LIM_SWITCH	Axis has no limit switch
33	PI_CNTR_NO_RELAY_CARD	No relay card installed
34	PI_CNTR_CMD_NOT_ALLOWED_FOR_STAGE	Command not allowed for selected stage(s)
35	PI_CNTR_NO_DIGITAL_INPUT	No digital input installed
36	PI_CNTR_NO_DIGITAL_OUTPUT	No digital output configured
37	PI_CNTR_NO_MCM	No more MCM responses
38	PI_CNTR_INVALID_MCM	No MCM values recorded
39	PI_CNTR_INVALID_CNTR_NUMBER	Controller number invalid
40	PI_CNTR_NO_JOYSTICK_CONNECTED	No joystick configured
41	PI_CNTR_INVALID_EGE_AXIS	Invalid axis for electronic gearing, axis can not be slave
42	PI_CNTR_SLAVE_POSITION_OUT_OF_RANGE	Position of slave axis is out of range
43	PI_CNTR_COMMAND_EGE_SLAVE	Slave axis cannot be commanded directly when electronic gearing is enabled
44	PI_CNTR_JOYSTICK_CALIBRATION_FAILED	Calibration of joystick failed
45	PI_CNTR_REFERENCING_FAILED	Referencing failed
46	PI_CNTR_OPM_MISSING	OPM (Optical Power Meter) missing
47	PI_CNTR_OPM_NOT_INITIALIZED	OPM (Optical Power Meter) not initialized or cannot be initialized
48	PI_CNTR_OPM_COM_ERROR	OPM (Optical Power Meter) Communication Error
49	PI_CNTR_MOVE_TO_LIMIT_SWITCH_FAILED	Move to limit switch failed
50	PI_CNTR_REF_WITH_REF_DISABLED	Attempt to reference axis with referencing disabled
51	PI_CNTR_AXIS_UNDER_JOYSTICK_CONTROL	Selected axis is controlled by joystick
52	PI_CNTR_COMMUNICATION_ERROR	Controller detected communication error
53	PI_CNTR_DYNAMIC_MOVE_IN_PROCESS	MOV! motion still in progress
54	PI_CNTR_UNKNOWN_PARAMETER	Unknown parameter
55	PI_CNTR_NO_REP_RECORDED	No commands were recorded with REP
56	PI_CNTR_INVALID_PASSWORD	Password invalid
57	PI_CNTR_INVALID_RECORDER_CHAN	Data Record Table does not exist
58	PI_CNTR_INVALID_RECORDER_SRC_OPT	Source does not exist; number too low or too high
59	PI_CNTR_INVALID_RECORDER_SRC_CHAN	Source Record Table number too low or too high
60	PI_CNTR_PARAM_PROTECTION	Protected Param: current Command Level (CCL) too low
61	PI_CNTR_AUTOZERO_RUNNING	Command execution not possible while Autozero is running
62	PI_CNTR_NO_LINEAR_AXIS	Autozero requires at least one linear axis
63	PI_CNTR_INIT_RUNNING	Initialization still in progress
64	PI_CNTR_READ_ONLY_PARAMETER	Parameter is read-only
65	PI_CNTR_PAM_NOT_FOUND	Parameter not found in non-volatile memory

66	PI_CNTR_VOL_OUT_OF_LIMITS	Voltage out of limits
67	PI_CNTR_WAVE_TOO_LARGE	Not enough memory available for requested wave curve
68	PI_CNTR_NOT_ENOUGH_DDL_MEMORY	Not enough memory available for DDL table; DDL can not be started
69	PI_CNTR_DDL_TIME_DELAY_TOO_LARGE	Time delay larger than DDL table; DDL can not be started
70	PI_CNTR_DIFFERENT_ARRAY_LENGTH	The requested arrays have different lengths; query them separately
71	PI_CNTR_GEN_SINGLE_MODE_RESTART	Attempt to restart the generator while it is running in single step mode
72	PI_CNTR_ANALOG_TARGET_ACTIVE	Motion commands and wave generator activation are not allowed when analog target is active
73	PI_CNTR_WAVE_GENERATOR_ACTIVE	Motion commands are not allowed when wave generator is active
74	PI_CNTR_AUTOZERO_DISABLED	No sensor channel or no piezo channel connected to selected axis (sensor and piezo matrix)
75	PI_CNTR_NO_WAVE_SELECTED	Generator started (WGO) without having selected a wave table (WSL).
76	PI_CNTR_IF_BUFFER_OVERRUN	Interface buffer did overrun and command couldn't be received correctly
77	PI_CNTR_NOT_ENOUGH_RECORDED_DATA	Data Record Table does not hold enough recorded data
78	PI_CNTR_TABLE_DEACTIVATED	Data Record Table is not configured for recording
79	PI_CNTR_OPENLOOP_VALUE_SET_WHEN_SERVO_ON	Open-loop commands (SVA, SVR) are not allowed when servo is on
80	PI_CNTR_RAM_ERROR	Hardware error affecting RAM
81	PI_CNTR_MACRO_UNKNOWN_COMMAND	Not macro command
82	PI_CNTR_MACRO_PC_ERROR	Macro counter out of range
83	PI_CNTR_JOYSTICK_ACTIVE	Joystick is active
84	PI_CNTR_MOTOR_IS_OFF	Motor is off
85	PI_CNTR_ONLY_IN_MACRO	Macro-only command
86	PI_CNTR_JOYSTICK_UNKNOWN_AXIS	Invalid joystick axis
87	PI_CNTR_JOYSTICK_UNKNOWN_ID	Joystick unknown
88	PI_CNTR_REF_MODE_IS_ON	Move without referenced stage
89	PI_CNTR_NOT_ALLOWED_IN_CURRENT_MOTION_MODE	Command not allowed in current motion mode
90	PI_CNTR_DIO_AND_TRACING_NOT_POSSIBLE	No tracing possible while digital IOs are used on this HW revision. Reconnect to switch operation mode.
91	PI_CNTR_COLLISION	Move not possible, would cause collision
92	PI_CNTR_SLAVE_NOT_FAST_ENOUGH	Stage is not capable of following the master. Check the gear ratio.
93	PI_CNTR_CMD_NOT_ALLOWED_WHILE_AXIS_IN_MOTION	This command is not allowed while the affected axis or its master is in motion.

94	PI_CNTR_OPEN_LOOP_JOYSTICK_ENABLED	Servo cannot be switched on when open-loop joystick control is activated.
95	PI_CNTR_INVALID_SERVO_STATE_FOR_PARAMETER	This parameter cannot be changed in current servo mode.
96	PI_CNTR_UNKNOWN_STAGE_NAME	Unknown stage name
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_HEADER	uudecode: incomplete header
207	PI_CNTR_UUDECODE_NOTHING_TO_DECODE	uudecode: nothing to decode
208	PI_CNTR_UUDECODE_ILLEGAL_FORMAT	uudecode: illegal UUE format
209	PI_CNTR_CRC32_ERROR	CRC32 error
210	PI_CNTR_ILLEGAL_FILENAME	Illegal file name (must be 8-0 format)
211	PI_CNTR_FILE_NOT_FOUND	File not found on controller
212	PI_CNTR_FILE_WRITE_ERROR	Error writing file on controller
213	PI_CNTR_DTR_HINDERS_VELOCITY_CHANGE	VEL command not allowed in DTR Command Mode
214	PI_CNTR_POSITION_UNKNOWN	Position calculations failed
215	PI_CNTR_CONN_POSSIBLY_BROKEN	The connection between controller and stage may be broken
216	PI_CNTR_ON_LIMIT_SWITCH	The connected stage has driven into a limit switch, some controllers need CLR to resume operation
217	PI_CNTR_UNEXPECTED_STRUT_STOP	Strut test command failed because of an unexpected strut stop
218	PI_CNTR_POSITION_BASED_ON_ESTIMATION	While MOV! is running position can only be estimated!
219	PI_CNTR_POSITION_BASED_ON_INTERPOLATION	Position was calculated during MOV motion
220	PI_CNTR_INTERPOLATION_FIFO_UNDERRUN	FIFO buffer underrun during interpolation
221	PI_CNTR_INTERPOLATION_FIFO_OVERFLOW	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_WHEN_SERVO_ON	Open-loop motion attempted when servo ON
304	PI_CNTR_RECEIVING_BUFFER_OVERFLOW	Received command is too long
305	PI_CNTR_EEPROM_ERROR	Error while reading/writing EEPROM
306	PI_CNTR_I2C_ERROR	Error on I2C bus
307	PI_CNTR_RECEIVING_TIMEOUT	Timeout while receiving command
308	PI_CNTR_TIMEOUT	A lengthy operation has not finished in the expected time
309	PI_CNTR_MACRO_OUT_OF_SPACE	Insufficient space to store macro
310	PI_CNTR_EUI_OLDVERSION_CFGDATA	Configuration data has old version number
311	PI_CNTR_EUI_INVALID_CFGDATA	Invalid configuration data
333	PI_CNTR_HARDWARE_ERROR	Internal hardware error
400	PI_CNTR_WAV_INDEX_ERROR	Wave generator index error
401	PI_CNTR_WAV_NOT_DEFINED	Wave table not defined
402	PI_CNTR_WAV_TYPE_NOT_SUPPORTED	Wave type not supported
403	PI_CNTR_WAV_LENGTH_EXCEEDS_LIMIT	Wave length exceeds limit
404	PI_CNTR_WAV_PARAMETER_NR	Wave parameter number error
405	PI_CNTR_WAV_PARAMETER_OUT_OF_LIMIT	Wave parameter out of range
406	PI_CNTR_WGO_BIT_NOT_SUPPORTED	WGO command bit not supported
500	PI_CNTR_EMERGENCY_STOP_BUTTON_ACTIVATED	The \"red knob\" is still set and disables system
501	PI_CNTR_EMERGENCY_STOP_BUTTON_WAS_ACTIVATED	The \"red knob\" was activated and still disables system - reanimation required
502	PI_CNTR_REDUNDANCY_LIMIT_EXCEEDED	Position consistency check failed
503	PI_CNTR_COLLISION_SWITCH_ACTIVATED	Hardware collision sensor(s) are activated
504	PI_CNTR_FOLLOWING_ERROR	Strut following error occurred, e.g. caused by overload or encoder failure
505	PI_CNTR_SENSOR_SIGNAL_INVALID	One sensor signal is not valid
506	PI_CNTR_SERVO_LOOP_UNSTABLE	Servo loop was unstable due to wrong parameter setting and switched off to avoid damage.
507	PI_CNTR_LOST_SPI_SLAVE_CONNECTION	Digital connection to external SPI slave device is lost
508	PI_CNTR_MOVE_ATTEMPT_NOT_PERMITTED	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_EXIST	A command refers to a node that has no parent node
532	PI_CNTR_NODE_IN_USE	Attempt to delete a node that is in use
533	PI_CNTR_NODE_DEFINITION_IS_CYCLIC	Definition of a node is cyclic
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_INCONSISTENT	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_SUPPORTED	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_FILE	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_QUERIED	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_TRIGGERED_BY_I2T	Overcurrent protection triggered by I2T-module
656	PI_CNTR_OVER_CURR_PROTEC_TRIGGERED_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_MISSING	External mode driver missing
718	PI_CNTR_CONFIGURATION_FAILURE_EXTERNAL_MODE	Missing or incorrect configuration of external mode
719	PI_CNTR_EXTERNAL_MODE_CYCLETIME_INVALID	External mode cycletime invalid

720	PI_CNTR_BRAKE_ACTIVATED	Brake is activated
725	PI_CNTR_DRIVE_STATE_TRANSITION_ERROR	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_ARGUMENT	Variable or argument not defined
1008	PI_CNTR_RUNNING_MACRO	Controller is (already) running a macro
1009	PI_CNTR_MACRO_INVALID_OPERATOR	Invalid or missing operator for condition. Check necessary spaces around operator.
1010	PI_CNTR_MACRO_NO_ANSWER	No 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_REACHED	Maximum motor output reached
1028	PI_CNTR_UNKNOWN_CHANNEL_IDENTIFIER	Unknown channel identifier
1063	PI_CNTR_EXT_PROFILE_UNALLOWED_CMD	User Profile Mode: Command is not allowed, check for required preparatory commands
1064	PI_CNTR_EXT_PROFILE_EXPECTING_MOTION_ERROR	User Profile Mode: First target position in User Profile is too far from current position
1065	PI_CNTR_PROFILE_ACTIVE	Controller is (already) in User Profile Mode
1066	PI_CNTR_PROFILE_INDEX_OUT_OF_RANGE	User Profile Mode: Block or Data Set index out of allowed range
1071	PI_CNTR_PROFILE_OUT_OF_MEMORY	User Profile Mode: Out of memory
1072	PI_CNTR_PROFILE_WRONG_CLUSTER	User Profile Mode: Cluster is not assigned to this axis
1073	PI_CNTR_PROFILE_UNKNOWN_CLUSTER_IDENTIFIER	Unknown cluster identifier
1090	PI_CNTR_TOO_MANY_TCP_CONNECTIONS_OPEN	There are too many open tcpip connections
2000	PI_CNTR_ALREADY_HAS_SERIAL_NUMBER	Controller already has a serial number
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

Schnittstellenfehler

0	COM_NO_ERROR	No error occurred during function call
-1	COM_ERROR	Error during com operation (could not be specified)
-2	SEND_ERROR	Error while sending data
-3	REC_ERROR	Error while receiving data
-4	NOT_CONNECTED_ERROR	Not connected (no port with given ID open)
-5	COM_BUFFER_OVERFLOW	Buffer overflow
-6	CONNECTION_FAILED	Error while opening port
-7	COM_TIMEOUT	Timeout error
-8	COM_MULTILINE_RESPONSE	There are more lines waiting in buffer
-9	COM_INVALID_ID	There is no interface or DLL handle with the given ID
-10	COM_NOTIFY_EVENT_ERROR	Event/message for notification could not be opened
-11	COM_NOT_IMPLEMENTED	Function not supported by this interface type
-12	COM_ECHO_ERROR	Error while sending "echoed" data

-13	COM_GPIB_EDVR	IEEE488: System error
-14	COM_GPIB_ECIC	IEEE488: Function requires GPIB board to be CIC
-15	COM_GPIB_ENOL	IEEE488: Write function detected no listeners
-16	COM_GPIB_EADR	IEEE488: Interface board not addressed correctly
-17	COM_GPIB_EARG	IEEE488: Invalid argument to function call
-18	COM_GPIB_ESAC	IEEE488: Function requires GPIB board to be SAC
-19	COM_GPIB_EABO	IEEE488: I/O operation aborted
-20	COM_GPIB_ENEB	IEEE488: Interface board not found
-21	COM_GPIB_EDMA	IEEE488: Error performing DMA
-22	COM_GPIB_EOIP	IEEE488: I/O operation started before previous operation completed
-23	COM_GPIB_ECAP	IEEE488: No capability for intended operation
-24	COM_GPIB_EFSO	IEEE488: File system operation error
-25	COM_GPIB_EBUS	IEEE488: Command error during device call
-26	COM_GPIB_ESTB	IEEE488: Serial poll-status byte lost
-27	COM_GPIB_ESRQ	IEEE488: SRQ remains asserted
-28	COM_GPIB_ETAB	IEEE488: Return buffer full
-29	COM_GPIB_ELCK	IEEE488: Address or board locked
-30	COM_RS_INVALID_DATA_BITS	RS-232: 5 data bits with 2 stop bits is an invalid combination, as is 6, 7, or 8 data bits with 1.5 stop bits
-31	COM_ERROR_RS_SETTINGS	RS-232: Error configuring the COM port
-32	COM_INTERNAL_RESOURCES_ERROR	Error dealing with internal system resources (events, threads, ...)
-33	COM_DLL_FUNC_ERROR	A DLL or one of the required functions could not be loaded
-34	COM_FTDIUSB_INVALID_HANDLE	FTDIUSB: invalid handle
-35	COM_FTDIUSB_DEVICE_NOT_FOUND	FTDIUSB: device not found
-36	COM_FTDIUSB_DEVICE_NOT_OPENED	FTDIUSB: device not opened
-37	COM_FTDIUSB_IO_ERROR	FTDIUSB: IO error
-38	COM_FTDIUSB_INSUFFICIENT_RESOURCES	FTDIUSB: insufficient resources
-39	COM_FTDIUSB_INVALID_PARAMETER	FTDIUSB: invalid parameter
-40	COM_FTDIUSB_INVALID_BAUD_RATE	FTDIUSB: invalid baud rate
-41	COM_FTDIUSB_DEVICE_NOT_OPENED_FOR_ERASE	FTDIUSB: device not opened for erase
-42	COM_FTDIUSB_DEVICE_NOT_OPENED_FOR_WRITE	FTDIUSB: device not opened for write
-43	COM_FTDIUSB_FAILED_TO_WRITE_DEVICE	FTDIUSB: failed to write device
-44	COM_FTDIUSB_EEPROM_READ_FAILED	FTDIUSB: EEPROM read failed
-45	COM_FTDIUSB_EEPROM_WRITE_FAILED	FTDIUSB: EEPROM write failed
-46	COM_FTDIUSB_EEPROM_ERASE_FAILED	FTDIUSB: EEPROM erase failed
-47	COM_FTDIUSB_EEPROM_NOT_PRESENT	FTDIUSB: EEPROM not present

-48	COM_FTDIUSB_EEPROM_NOT_PROGRAMMED	FTDIUSB: EEPROM not programmed
-49	COM_FTDIUSB_INVALID_ARGS	FTDIUSB: invalid arguments
-50	COM_FTDIUSB_NOT_SUPPORTED	FTDIUSB: not supported
-51	COM_FTDIUSB_OTHER_ERROR	FTDIUSB: other error
-52	COM_PORT_ALREADY_OPEN	Error while opening the COM port: was already open
-53	COM_PORT_CHECKSUM_ERROR	Checksum error in received data from COM port
-54	COM_SOCKET_NOT_READY	Socket not ready, you should call the function again
-55	COM_SOCKET_PORT_IN_USE	Port is used by another socket
-56	COM_SOCKET_NOT_CONNECTED	Socket not connected (or not valid)
-57	COM_SOCKET_TERMINATED	Connection terminated (by peer)
-58	COM_SOCKET_NO_RESPONSE	Can't connect to peer
-59	COM_SOCKET_INTERRUPTED	Operation was interrupted by a nonblocked signal
-60	COM_PCI_INVALID_ID	No device with this ID is present
-61	COM_PCI_ACCESS_DENIED	Driver could not be opened (on Vista: run as administrator!)
-62	COM_SOCKET_HOST_NOT_FOUND	Host not found
-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-Fehler

-1001	PI_UNKNOWN_AXIS_IDENTIFIER	Unknown axis identifier
-1002	PI_NR_NAV_OUT_OF_RANGE	Number for NAV out of range--must be in [1,10000]
-1003	PI_INVALID_SGA	Invalid value for SGA--must be one of 1, 10, 100, 1000
-1004	PI_UNEXPECTED_RESPONSE	Controller sent unexpected response
-1005	PI_NO_MANUAL_PAD	No manual control pad installed, calls to SMA and related commands are not allowed
-1006	PI_INVALID_MANUAL_PAD_KNOB	Invalid number for manual control pad knob
-1007	PI_INVALID_MANUAL_PAD_AXIS	Axis not currently controlled by a manual control pad
-1008	PI_CONTROLLER_BUSY	Controller is busy with some lengthy operation (e.g., reference move, fast scan algorithm)
-1009	PI_THREAD_ERROR	Internal error--could not start thread
-1010	PI_IN_MACRO_MODE	Controller is (already) in macro mode--command not valid in macro mode

-1011	PI_NOT_IN_MACRO_MODE	Controller not in macro mode--command not valid unless macro mode active
-1012	PI_MACRO_FILE_ERROR	Could not open file to write or read macro
-1013	PI_NO_MACRO_OR_EMPTY	No macro with given name on controller, or macro is empty
-1014	PI_MACRO_EDITOR_ERROR	Internal error in macro editor
-1015	PI_INVALID_ARGUMENT	One or more arguments given to function is invalid (empty string, index out of range, ...)
-1016	PI_AXIS_ALREADY_EXISTS	Axis identifier is already in use by a connected stage
-1017	PI_INVALID_AXIS_IDENTIFIER	Invalid axis identifier
-1018	PI_COM_ARRAY_ERROR	Could not access array data in COM server
-1019	PI_COM_ARRAY_RANGE_ERROR	Range of array does not fit the number of parameters
-1020	PI_INVALID_SPA_CMD_ID	Invalid parameter ID given to SPA or SPA?
-1021	PI_NR_AVG_OUT_OF_RANGE	Number for AVG out of range--must be >0
-1022	PI_WAV_SAMPLES_OUT_OF_RANGE	Incorrect number of samples given to WAV
-1023	PI_WAV_FAILED	Generation of wave failed
-1024	PI_MOTION_ERROR	Motion error: position error too large, servo is switched off automatically
-1025	PI_RUNNING_MACRO	Controller is (already) running a macro
-1026	PI_PZT_CONFIG_FAILED	Configuration of PZT stage or amplifier failed
-1027	PI_PZT_CONFIG_INVALID_PARAMS	Current settings are not valid for desired configuration
-1028	PI_UNKNOWN_CHANNEL_IDENTIFIER	Unknown channel identifier
-1029	PI_WAVE_PARAM_FILE_ERROR	Error while reading/writing wave generator parameter file
-1030	PI_UNKNOWN_WAVE_SET	Could not find description of wave form. Maybe WG.INI is missing?
-1031	PI_WAVE_EDITOR_FUNC_NOT_LOADED	The WGWaveEditor DLL function was not found at startup
-1032	PI_USER_CANCELLED	The user cancelled a dialog
-1033	PI_C844_ERROR	Error from C-844 Controller
-1034	PI_DLL_NOT_LOADED	DLL necessary to call function not loaded, or function not found in DLL
-1035	PI_PARAMETER_FILE_PROTECTED	The open parameter file is protected and cannot be edited
-1036	PI_NO_PARAMETER_FILE_OPENED	There is no parameter file open
-1037	PI_STAGE_DOES_NOT_EXIST	Selected stage does not exist
-1038	PI_PARAMETER_FILE_ALREADY_OPENED	There is already a parameter file open. Close it before opening a new file
-1039	PI_PARAMETER_FILE_OPEN_ERROR	Could not open parameter file

-1040	PI_INVALID_CONTROLLER_VERSION	The version of the connected controller is invalid
-1041	PI_PARAM_SET_ERROR	Parameter could not be set with SPA--parameter not defined for this controller!
-1042	PI_NUMBER_OF_POSSIBLE_WAVES_EXCEEDED	The maximum number of wave definitions has been exceeded
-1043	PI_NUMBER_OF_POSSIBLE_GENERATORS_EXCEEDED	The maximum number of wave generators has been exceeded
-1044	PI_NO_WAVE_FOR_AXIS_DEFINED	No wave defined for specified axis
-1045	PI_CANT_STOP_OR_START_WAV	Wave output to axis already stopped/started
-1046	PI_REFERENCE_ERROR	Not all axes could be referenced
-1047	PI_REQUIRED_WAVE_NOT_FOUND	Could not find parameter set required by frequency relation
-1048	PI_INVALID_SPP_CMD_ID	Command ID given to SPP or SPP? is not valid
-1049	PI_STAGE_NAME_ISNT_UNIQUE	A stage name given to CST is not unique
-1050	PI_FILE_TRANSFER_BEGIN_MISSING	A uuencoded file transferred did not start with "begin" followed by the proper filename
-1051	PI_FILE_TRANSFER_ERROR_TEMP_FILE	Could not create/read file on host PC
-1052	PI_FILE_TRANSFER_CRC_ERROR	Checksum error when transferring a file to/from the controller
-1053	PI_COULDNT_FIND_PISTAGES_DAT	The PiStages.dat database could not be found. This file is required to connect a stage with the CST command
-1054	PI_NO_WAVE_RUNNING	No wave being output to specified axis
-1055	PI_INVALID_PASSWORD	Invalid password
-1056	PI_OPM_COM_ERROR	Error during communication with OPM (Optical Power Meter), maybe no OPM connected
-1057	PI_WAVE_EDITOR_WRONG_PARAMNUM	WaveEditor: Error during wave creation, incorrect number of parameters
-1058	PI_WAVE_EDITOR_FREQUENCY_OUT_OF_RANGE	WaveEditor: Frequency out of range
-1059	PI_WAVE_EDITOR_WRONG_IP_VALUE	WaveEditor: Error during wave creation, incorrect index for integer parameter
-1060	PI_WAVE_EDITOR_WRONG_DP_VALUE	WaveEditor: Error during wave creation, incorrect index for floating point parameter
-1061	PI_WAVE_EDITOR_WRONG_ITEM_VALUE	WaveEditor: Error during wave creation, could not calculate value
-1062	PI_WAVE_EDITOR_MISSING_GRAPH_COMPONENT	WaveEditor: Graph display component not installed
-1063	PI_EXT_PROFILE_UNALLOWED_CMD	User Profile Mode: Command is not allowed, check for required preparatory commands
-1064	PI_EXT_PROFILE_EXPECTING_MOTION_ERROR	User Profile Mode: First target position in User Profile is too far from current position
-1065	PI_EXT_PROFILE_ACTIVE	Controller is (already) in User Profile Mode

-1066	PI_EXT_PROFILE_INDEX_OUT_OF_RANGE	User Profile Mode: Block or Data Set index out of allowed range
-1067	PI_PROFILE_GENERATOR_NO_PROFILE	ProfileGenerator: No profile has been created yet
-1068	PI_PROFILE_GENERATOR_OUT_OF_LIMITS	ProfileGenerator: Generated profile exceeds limits of one or both axes
-1069	PI_PROFILE_GENERATOR_UNKNOWN_PARAMETER	ProfileGenerator: Unknown parameter ID in Set/Get Parameter command
-1070	PI_PROFILE_GENERATOR_PAR_OUT_OF_RANGE	ProfileGenerator: Parameter out of allowed range
-1071	PI_EXT_PROFILE_OUT_OF_MEMORY	User Profile Mode: Out of memory
-1072	PI_EXT_PROFILE_WRONG_CLUSTER	User Profile Mode: Cluster is not assigned to this axis
-1073	PI_UNKNOWN_CLUSTER_IDENTIFIER	Unknown cluster identifier
-1074	PI_INVALID_DEVICE_DRIVER_VERSION	The installed device driver doesn't match the required version. Please see the documentation to determine the required device driver version.
-1075	PI_INVALID_LIBRARY_VERSION	The library used doesn't match the required version. Please see the documentation to determine the required library version.
-1076	PI_INTERFACE_LOCKED	The interface is currently locked by another function. Please try again later.
-1077	PI_PARAM_DAT_FILE_INVALID_VERSION	Version of parameter DAT file does not match the required version. Current files are available at www.pi.ws.
-1078	PI_CANNOT_WRITE_TO_PARAM_DAT_FILE	Cannot write to parameter DAT file to store user defined stage type.
-1079	PI_CANNOT_CREATE_PARAM_DAT_FILE	Cannot create parameter DAT file to store user defined stage type.
-1080	PI_PARAM_DAT_FILE_INVALID_REVISION	Parameter DAT file does not have correct revision.
-1081	PI_USERSTAGES_DAT_FILE_INVALID_REVISION	User stages DAT file does not have correct revision.
-1082	PI_SOFTWARE_TIMEOUT	Timeout Error. Some lengthy operation did not finish within expected time.
-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_IN_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_ARGUMENTS	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_ENTRY_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_VERSION_DOES_NOT_EXIST_IN_PISTAGES_DATABASE	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_ERROR	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_AVAILABLE	PI stage database: Database does not contain the selected stage type for the connected controller.
-10002	PI_PARAMETER_DB_FAILED_TO_ESTABLISH_CONNECTION	PI stage database: Establishing the connection has failed.
-10003	PI_PARAMETER_DB_COMMUNICATION_ERROR	PI stage database: Communication was interrupted (e.g. because database was deleted).
-10004	PI_PARAMETER_DB_ERROR_WHILE_QUERYING_PARAMETERS	PI stage database: Querying data failed.
-10005	PI_PARAMETER_DB_SYSTEM_ALREADY_EXISTS	PI stage database: System already exists. Rename stage and try again.
-10006	PI_PARAMETER_DB_QHPA_CONTANS_UNKNOWN_PAM_IDS	PI stage database: Response to HPA? contains unknown parameter IDs.
-10007	PI_PARAMETER_DB_AND_QHPA_ARE_INCONSISTENT	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_PARAMETERS_MISMATCH	Controller does not support all stage parameters stored in PI stage database. No parameters were set.
-10011	PI_PARAMETER_DB_DATABASE_IS_OUTDATED	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_MISMATCH_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_MISMATCH_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_PARAMETERS_CORRECTLY	One or more parameters could not be set correctly on the controller.
-10015	PI_PARAMETER_DB_MISSING_PARAMETER_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

In diesem Kapitel

Überblick über die Einstellungen des C-887 317
Parameterwerte im C-887 ändern 317
Parameterwerte in Textdatei sichern 319
Parameterübersicht 320

9.1 Überblick über die Einstellungen des C-887

Für den C-887 können verschiedene Einstellungen als Standardwerte im permanenten Speicher hinterlegt werden, so dass sie beim Ausschalten oder Neustart des C-887 erhalten bleiben. Die Standardwerte für die Einstellungen stammen aus verschiedenen Quellen:

- Konfigurationsdateien: siehe "Firmware und Konfigurationsdateien aktualisieren" (S. 340)
- Positioniererdatenbanken: siehe "Betriebsparameter der Achsen A und B" (S. 32)
- Schnittstellenparameter: Anpassung im permanenten Speicher mit dem Befehl `IFS` (S. 207) möglich, weitere Informationen in "Kommunikation über TCP/IP-Schnittstelle herstellen" (S. 74) und "Kommunikation über RS-232-Schnittstelle herstellen" (S. 82)
- Befehle für die Arbeit mit benutzerdefinierten Koordinatensystemen: weitere Informationen in "Koordinatensysteme" (S. 41)
- Parameter, die mit dem Befehl `SPA` (S. 258) geändert und mit dem Befehl `WPA` (S. 293) gespeichert werden können. Diese 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 auf die Parameter wird durch Befehlsebenen (S. 162) festgelegt.

  Weitere Informationen in "Parameterwerte im C-887 ändern" (S. 317).

9.2 Parameterwerte im C-887 ändern

Jeder Parameter ist sowohl im flüchtigen als auch im permanenten Speicher des C-887 vorhanden. Die Werte im permanenten Speicher werden als Standardwerte beim Einschalten oder Neustart des C-887 in den flüchtigen Speicher geladen. Die Werte im flüchtigen Speicher bestimmen das aktuelle Verhalten des Systems.

Verfügbare Parameter ermitteln

- Senden Sie den Befehl `HPA?` (S. 204), um eine Liste aller verfügbaren Parameter des C-887 mit Kurzbeschreibung zu erhalten.

Parameterwerte im flüchtigen Speicher ändern

INFORMATION

Das Ändern von Parameterwerten kann zu unerwünschten Ergebnissen führen.

- Legen Sie vor dem Ändern der Parametereinstellungen des C-887 eine Sicherungskopie auf dem PC an, siehe "Parameterwerte in Textdatei sichern" (S. 319). Sie können dann jederzeit die Originaleinstellungen wiederherstellen.

- Senden Sie den Befehl `SPA?` (S. 259), um eine Liste der Parameterwerte im flüchtigen Speicher des C-887 zu erhalten.
- Ändern Sie die Parameterwerte im flüchtigen Speicher:
  a) Wenn für den Schreibzugriff auf die Parameterwerte notwendig, senden Sie den Befehl `CCL 1 advanced`, um auf die Befehlsebene 1 zu wechseln.
  b) Ändern Sie die Parameterwerte im flüchtigen Speicher des C-887 mit dem Befehl `SPA` (S. 258).

INFORMATION

Parameterwerte werden auch durch den Befehl `CST` (S. 163) geändert, der den Achsen A und B einen Positionierertyp zuweist. Dabei werden die entsprechenden Parameterwerte aus einer Positioniererdatenbank in den flüchtigen Speicher geladen. Weitere Informationen finden Sie in "Betriebsparameter der Achsen A und B" (S. 32).

Parameterwerte vom flüchtigen Speicher in den permanenten Speicher schreiben

INFORMATION

Für das Speichern von Parameterwerten im permanenten Speicher mit dem Befehl `WPA` ist die Eingabe eines Kennworts erforderlich. Verwendbare Kennwörter:

100	Speichert die aktuell gültigen Werte aller Parameter, die aktuell gültigen Einstellungen für Koordinatensysteme und die aktuelle Zuweisung von Positionierertypen zu den Achsen A und B
101	Speichert die aktuell gültigen Werte aller Parameter. Parameter können einzeln ausgewählt werden.
SKS	Speichert die aktuell gültigen Einstellungen für Koordinatensysteme
A12	Zuweisung von Positionierertypen zu den Achsen A und B

- Schreiben Sie die aktuellen Parameterwerte mit dem Befehl `WPA` (S. 293) in den permanenten Speicher des C-887.

Alternatives Vorgehen, wenn Sie mit PIMikroMove® arbeiten:

a) Wählen Sie im Hauptfenster von PIMikroMove® den Menüeintrag ***C-887 > Save parameters to non-volatile memory***. Der Dialog ***Save Parameters to Non-Volatile Memory*** öffnet sich.

b) Geben Sie in das Auswahlfeld im Dialog ***Save Parameters to Non-Volatile Memory*** entweder das passende Kennwort ein, oder wählen Sie den passenden Eintrag.

c) Wählen Sie ***OK***, um das Speichern auszuführen und den Dialog zu schließen.

9.3 Parameterwerte in Textdatei sichern

Voraussetzungen

✓ Sie haben die allgemeinen Hinweise zur Inbetriebnahme gelesen und verstanden (S. 69).

✓ Sie haben die Kommunikation zwischen dem C-887 und dem PC mit PIMikroMove® oder PITerminal über TCP/IP (S. 79) oder RS-232 (S. 82) hergestellt.

Parameterwerte in Textdatei sichern

1. Wenn Sie PIMikroMove® verwenden, öffnen Sie das Fenster zum Senden von Befehlen:

   – Wählen Sie im Hauptfenster den Menüeintrag ***Tools > Command entry*** oder drücken Sie die Taste F4 auf der Tastatur.

   In PITerminal ist nach dem Herstellen der Kommunikation automatisch das Hauptfenster geöffnet, aus dem Befehle gesendet werden können.

2. Fragen Sie mit dem Befehl `SPA?` die aktuellen Parameterwerte des C-887 ab.

3. Wählen Sie 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.

9.4 Parameterübersicht

INFORMATION

Der Schreibzugriff auf die Parameter des C-887 ist durch Befehlsebenen festgelegt. Nach dem Einschalten oder Neustart des Controllers ist die aktive Befehlsebene immer 0. Für bestimmte Parameter ist der Schreibzugriff nur auf der Befehlsebene 1 zugelassen. Auf Befehlsebenen > 1 besteht Schreibzugriff nur für Servicepersonal von PI.

- Wenn notwendig, senden Sie den Befehl `CCL 1 advanced` oder geben Sie das Kennwort `advanced` ein, um auf die Befehlsebene 1 zu wechseln.
- Wenn Sie Probleme mit Parametern der Befehlsebene 2 oder höher haben, wenden Sie sich an den Kundendienst (S. 367).

Parameter-ID (hexa-dezimal)	Befehls-ebene für den Schreib-zugriff	Betroffener Elementtyp	Daten-typ	Parameter-name (Einheit)	Beschreibung
0x1	0	Hexapod-Beine 1 bis 6, Achsen A und B	INT	P-Term	Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x2	0	Hexapod-Beine 1 bis 6, Achsen A und B	INT	I-Term	Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x3	0	Hexapod-Beine 1 bis 6, Achsen A und B	INT	D-Term	Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x4	0	Hexapod-Beine 1 bis 6, Achsen A und B	INT	I-Limit	Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x5	0	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Velocity Feed Forward	Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.

Parameter-ID (hexa-dezimal)	Befehls-ebene für den Schreib-zugriff	Betroffener Elementtyp	Daten-typ	Parameter-name (Einheit)	Beschreibung
0x8	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Maximum Position Error (mm)	Maximaler Positionsfehler Wird für die Erkennung von Bewegungsfehlern verwendet. Falls die Positionsabweichung (d. h. der absolute Wert der Differenz zwischen der aktuellen Position und der kommandierten Position) im geregelten Betrieb den vorgegebenen Maximalwert überschreitet, setzt der C-887 den Fehlercode +1024 ("Motion error"). Details zur Fehlerbehandlung siehe "Automatisches Ausschalten des Servomodus / Stoppen der Bewegung" (S. 91). Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x9	0	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Maximum Motor Output	Maximal zulässiger Betrag des Stellwerts des Motortreibers (dimensionslos) 0 bis 32767 Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0xA	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Maximum Allowed Velocity	Maximale Geschwindigkeit Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0xB	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Current Acceleration	Beschleunigung Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0xC	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Current Deceleration	Verzögerung Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.

Parameter-ID (hexa-dezimal)	Befehls-ebene für den Schreib-zugriff	Betroffener Elementtyp	Daten-typ	Parameter-name (Einheit)	Beschreibung
0xE	0	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Numerator Of The Counts-Per-Physical-Unit Factor	Zähler und Nenner des Faktors für Impulse pro physikalische Längeneinheit 1 bis 1.000.000 für jeden Parameter. Der Faktor für Impulse pro physikalische Längeneinheit bestimmt die Einheit für Bewegungsbefehle. Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0xF	0	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Denominator Of The Counts-Per-Physical-Unit Factor	
0x10	1	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Output Mode	1 = PWM Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x13	0	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Is Rotary Stage?	Handelt es sich um einen Rotationstisch? 0 = Kein Rotationstisch 1 = Rotationstisch Keine Auswertung durch den C-887, sondern nur durch die PC-Software: PIMikroMove® entscheidet anhand dieses Wertes, welche Bewegungen zulässig sind. Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x14	0	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Stage has reference switch	Ist ein Referenzschalter vorhanden? 0 = Kein Referenzschalter eingebaut 1 = Referenzschalter vorhanden Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x15	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Maximum Travel In Positive Direction (Phys. Unit)	Verfahrbereichsgrenze in positiver Richtung Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x16	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Value at reference switch position	Positionswert am Referenzschalter Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.

Parameter-ID (hexa-dezimal)	Befehls-ebene für den Schreib-zugriff	Betroffener Elementtyp	Daten-typ	Parameter-name (Einheit)	Beschreibung
0x17	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Distance between Negative Limit To Reference Position (Phys. Unit)	Abstand zwischen Referenzschalter und negativem Endschalter Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x18	0	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Limit Mode	Signallogik der Endschalter Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x1A	0	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Has brake?	Ist eine Bremse vorhanden? 0 = Keine Bremse vorhanden 1 = Bremse vorhanden Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x2F	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Distance between Reference Position To Positive Limit (Phys. Unit)	Abstand zwischen Referenzschalter und positivem Endschalter Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x30	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Maximum Travel In Negative Direction (Phys. Unit)	Verfahrbereichsgrenze in negativer Richtung Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x31	0	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Invert Reference Switch Signal	Soll das Referenzsignal invertiert werden? Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x32	0	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Has No Limit Switches	Sind Endschalter vorhanden? 0 = Endschalterauswertung aktiviert. Nur wirksam, wenn die Mechanik tatsächlich Endschaltersignale an den C-887 liefert. 1 = Endschalterauswertung deaktiviert. Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.

Parameter-ID (hexa-dezimal)	Befehls-ebene für den Schreib-zugriff	Betroffener Elementtyp	Daten-typ	Parameter-name (Einheit)	Beschreibung
0x33	1	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Offset for the positive direction of motion	Offset für die positive Bewegungsrichtung Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x34	1	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Offset for the negative direction of motion	Offset für die negative Bewegungsrichtung Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x36	0	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Settling Window (encoder counts)	Einschwingfenster um die Zielposition Wird verwendet für die Ermittlung des Bewegungsstatus (S. 46). Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x38	0	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Settle Time	Verzögerungszeit Angabe in Servozyklen Wird verwendet für die Ermittlung des Bewegungsstatus (S. 46).
0x3B	0	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Acceleration feed forward	
0x3C	0	Hexapod-Beine 1 bis 6, Achsen A und B	CHAR	Stage Name	Positionierername Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x3F	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Settling Time (s)	Verzögerungszeit für das Setzen des On-Target-Status. Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x47	3	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Default Direction for the reference move	Standardrichtung für die Referenzierungsfahrt 0 = Automatic Detection Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x48	1	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Velocity-dependent offset	Geschwindigkeitsabhängiger Offset Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.

Parameter-ID (hexa-dezimal)	Befehls-ebene für den Schreib-zugriff	Betroffener Elementtyp	Daten-typ	Parameter-name (Einheit)	Beschreibung
0x49	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Velocity in closed-loop operation (Phys. Unit/s)	Geschwindigkeit im geregelten Betrieb Wird begrenzt durch Parameter 0xA Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x4A	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Maximum allowed acceleration	Maximale Beschleunigung Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x4B	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Maximum allowed deceleration	Maximale Verzögerung Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x50	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Velocity for Reference Moves (Phys. Unit/s)	Höchstgeschwindigkeit für Referenzierungsfahrt Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x5A	0	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Numerator of the Servo-Loop Input Factor	Zähler und Nenner des Eingangsfaktors des Regelkreises Der Eingangsfaktor des Regelkreises entkoppelt die Regelparameter von der Encoderauflösung. 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. Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x5B	0	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Denumerator of the Servo-Loop Input Factor	
0x63		Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Distance between internal limit switch and hard stop (Phys. Unit)	Abstand zwischen eingebautem Endschalter und mechanischem Anschlag Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.

Parameter-ID (hexa-dezimal)	Befehls-ebene für den Schreib-zugriff	Betroffener Elementtyp	Daten-typ	Parameter-name (Einheit)	Beschreibung
0x70	0	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Reference Signal Type	Art des Referenzsignals 0 = Richtungserkennender Referenzschalter. 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 des Referenzschalters erfolgt über die negative Grenze des Stellwegs. 3 = Indexpuls. Das Anfahren des Referenzschalters erfolgt über die positive Grenze des Stellwegs. 4 = Kein Referenzsignal 5 = Das Referenzsignal wird am negativen Endschalter ausgegeben. 6 = Das Referenzsignal wird am positiven Endschalter ausgegeben. Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x71	0	Hexapod-Beine 1 bis 6, Achsen A und B	INT	D-Term-Delay (No. of Servo Cycles)	D-Term-Verzögerung 1 bis 8 Servozyklen Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x72	0	System	INT	Ignore Macro Error?	Makrofehler ignorieren? Legt fest, ob das Controllermakro gestoppt wird, wenn bei dessen Ausführung ein Fehler auftritt. 0 = Bei Fehler Makro anhalten (Standard) 1 = Makrofehler ignorieren
0x77	0	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Use Limit Switches only for Reference Move	Sollen die Endschalter nur für Referenzierungsfahrten verwendet werden? Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.

Parameter-ID (hexa-dezimal)	Befehls-ebene für den Schreib-zugriff	Betroffener Elementtyp	Daten-typ	Parameter-name (Einheit)	Beschreibung
0x78	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Distance between limit switch and the starting position (Phys. Unit)	Abstand zwischen Endschalter oder mechanischem Anschlag und der Startposition für die Referenzierungsfahrt zum Indexpuls Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x79	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Maximum distance for the motion to the index pulse (Phys. Unit)	Maximale Strecke für die Referenzierungsfahrt zum Indexpuls Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x94	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Notch Filter Frequency 1 (Hz)	Frequenz des ersten Kerbfilters Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0x95	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Notch Filter Edge 1	Anstieg der Flanke des ersten Kerbfilters Wird für die Achsen A und B bei Zuweisung eines Positionierertyps im flüchtigen Speicher geändert.
0xAC	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Lowpass Filter Frequency 1 (Hz)	Frequenz des ersten Tiefpassfilters
0x03000100	2	Eingangs-signalkanal	FLOAT	Polynomial 0 order of Range X (X = Index)	Koeffizienten des Polynoms zur Linearisierung der analogen Eingänge
0x03000200	2	Eingangs-signalkanal	FLOAT	Polynomial 1st order of Range X (X = Index)	
0x03000300	2	Eingangs-signalkanal	FLOAT	Polynomial 2nd order of Range X (X = Index)	
0x03000400	2	Eingangs-signalkanal	FLOAT	Polynomial 3rd order of Range X (X = Index)	

Parameter-ID (hexa-dezimal)	Befehls-ebene für den Schreib-zugriff	Betroffener Elementtyp	Daten-typ	Parameter-name (Einheit)	Beschreibung
0x03000500	2	Eingangs-signalkanal	FLOAT	Polynomial 4th order of Range X (X = Index)	
0x03000600	2	Eingangs-signalkanal	FLOAT	Polynomial 5th order of Range X (X = Index)	
0x03003320	1	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Type of Sensor	Vom Positionssensor ausgegebener Signaltyp 1 = inkrementeller Sensor, AB-Quadratursignale (Standardeinstellung) 3 = inkrementeller Sensor, BiSS (32 Bit) 4 = absolut messender Sensor, BiSS (32 Bit) 7 = inkrementeller Sensor, BiSS (24 Bit) 8 = absolut messender Sensor, BiSS (24 Bit)
0x03003330	1	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Sensor Position Offset	Offset für die Korrektur der Signale eines absolut messenden Sensors
0x03003340	1	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Sensor Number of CRC Errors	Anzahl Prüfsummenfehler, wenn ein Sensorsignaltyp mit BiSS-Protokoll verwendet wird
0x03003360	3	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Sensor Home Position	Interne Skalierung/Rundung des Sensorwertes
0x04000300	3	Eingangs-signalkanal	FLOAT	ADC Range Minimum	Einstellungen für die analogen Eingänge des C-887: ▪ Wertebereich des A/D-Wandlers ▪ Verstärkungsfaktor ▪ Bit-Auflösung ▪ Einheit für Anzeige
0x04000400	3	Eingangs-signalkanal	FLOAT	ADC Range Maximum	
0x04000700	3	Eingangs-signalkanal	FLOAT	Hardware Gain	
0x04000B01	3	Eingangs-signalkanal	INT	ADC Input Resolution	
0x04000E00	3	Eingangs-signalkanal	CHAR	Display Unit	

Parameter-ID (hexa-dezimal)	Befehls-ebene für den Schreib-zugriff	Betroffener Elementtyp	Daten-typ	Parameter-name (Einheit)	Beschreibung
0x07000000	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Range Limit Min	Grenzen des zulässigen Bereichs für die Bewegung (in physikalischen Einheiten)
0x07000001	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Range Limit Max	
0x07000601	0		CHAR	Axis Unit	Einheitenzeichen
0x07030401	0	Hexapod-Achse	INT	Behaviour After Reference Move	Verhalten der Bewegungsplattform des Hexapods nach der Referenzierungsfahrt Details siehe "Bewegungen des Hexapods" (S. 34).
0x07030402	0	Hexapod-Achse	FLOAT	Target For Motion After Reference Move	
0x0D000200	2	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Hardware Name	
0x0D000700	2	System	CHAR	Device Name	Produktbezeichnung des C-887
0x0D001000	1	System	CHAR	Customer Device Name	Frei wählbarer Namenszusatz für den C-887 Kann durch den Anwender angepasst werden, um mehrere C-887 im selben Netzwerk oder am selben PC unterscheiden zu können. Wird z. B. beim Herstellen der Kommunikation über TCP/IP in der Liste der gefundenen Controller angezeigt.
0x0E000200	3	System	FLOAT	Servo Update Time	Servozykluszeit in Sekunden
0x0E000900	0	System	INT	Pulse Length Factor	Wird mit der Zyklusdauer multipliziert (S. 376) Produkt entspricht der Pulsweite des Impulses Standard: 5

Parameter-ID (hexa-dezimal)	Befehls-ebene für den Schreib-zugriff	Betroffener Elementtyp	Daten-typ	Parameter-name (Einheit)	Beschreibung
0x0E000B00	3	System	INT	Number Of Input Channels	Anzahl der installierten analogen Eingangskanäle Details zu den Kennungen der Eingangskanäle siehe "Kommandierbare Elemente" (S. 27). Entspricht der Antwort auf `TAC?` (S. 269)
0x0E001600	0	System	INT	HID Device Button Mode	Verhalten der Drucktasten einer manuellen Bedieneinheit (C-887.MC2 oder C-887.MC) 0 = Drucktasten lösen Aktionen aus (Stop, Referenzierungsfahrt; Standard) 1 = Drucktasten lösen keine Aktionen aus Weitere Informationen siehe "Bewegungen des Hexapods" (S. 34).
0x13000004	3	System	INT	Max Wave Points	Gesamtanzahl der für Kurvenformen verfügbaren Punkte Details siehe "Funktionsgenerator" (S. 102).
0x1300010A	3	System	INT	Number Of Wave Tables	Anzahl der Kurventabellen zum Speichern von Kurvenformen Details siehe "Funktionsgenerator" (S. 102).
0x16000000	0	System	INT	Data Recorder Table Rate	Aufzeichnungsrate des Datenrekorders 1 bis 10000 Standard: 10 Siehe auch `RTR` (S. 254).
0x16000100	3	System	INT	Max. Number of Data Recorder Channels	Maximale Anzahl der Datenrekordertabellen
0x16000200	3	System	INT	Data Recorder Max. Points	Maximale Anzahl aller Punkte der Datenrekordertabellen
0x16000201	0	System	INT	Data Recorder Points Per Table	Anzahl der Punkte pro Datenrekordertabelle 1 bis 10240 Standard: 1024

Parameter-ID (hexa-dezimal)	Befehls-ebene für den Schreib-zugriff	Betroffener Elementtyp	Daten-typ	Parameter-name (Einheit)	Beschreibung
0x16000300	3	System	INT	Channel Number	Anzahl der Datenrekordertabellen
0x19000951	3	System	INT	Status	Lizenzstatus der kostenpflichtigen PIVeriMove Hexapod Software zur Kollisionsprüfung (C-887.VM1, siehe "Optionales Zubehör" (S. 24)) -2 = Lizenz ungültig -1 = Konfigurationsdatei ungültig, z. B. falsch formatiert 0 = nicht aktiviert 1 = aktiviert
0x19000952	0	System	FLOAT	Security Distance (mm)	Sicherheitsabstand für die Verwendung in der kostenpflichtigen PIVeriMove Hexapod Software zur Kollisionsprüfung Parameterwert muss ≥ 0 sein
0x19001500	3	System	FLOAT	Maximum System Velocity (mm/s)	Maximale Systemgeschwindigkeit Siehe Spezifikationen im Benutzerhandbuch des Hexapods und "Bewegungen des Hexapods" (S. 34).
0x19001501	3	System	FLOAT	Minimum System Velocity (mm/s)	Minimale Systemgeschwindigkeit Details siehe "Bewegungen des Hexapods" (S. 34).
0x19001502	3	System	FLOAT	Maximum System Acceleration (mm/s^2)	Maximale Systembeschleunigung Details siehe "Bewegungen des Hexapods" (S. 34).
0x19001504	3	System	FLOAT	Path Control Step Size (mm)	Schrittweite für die Berechnung des Dynamikprofils der Plattformbewegung Details siehe "Bewegungen des Hexapods" (S. 34).
0x19001510	0	System	FLOAT	Trajectory Velocity (Phys. Unit/s)	Geschwindigkeit, Beschleunigung und Ruck für die Bewegungsplattform des Hexapods Details siehe "Profilgenerator für Punkt-zu-Punkt-Bewegungen" (S. 37).
0x19001511	0	System	FLOAT	Trajectory Acceleration (Phys. Unit/s)	
0x19001512	0	System	FLOAT	Trajectory Jerk (Phys. Unit/s)	

Parameter-ID (hexa-dezimal)	Befehls-ebene für den Schreib-zugriff	Betroffener Elementtyp	Daten-typ	Parameter-name (Einheit)	Beschreibung
0x19001800	0	System	INT	Coordination Mode	Koordinationsmodus 0 = Mehrachsmodus 1 = Hexapod-Modus (Standard) Der Parameterwert darf nicht geändert werden.
0x19001900	0	System	INT	Trajectory Source	Quelle des Dynamikprofils für MOV-Befehle 0 = Dynamikprofil wird durch Profilgenerator festgelegt (Standard) 1 = Dynamikprofil wird durch aufeinander folgende `MOV`-Befehle festgelegt Details siehe "Bewegungen des Hexapods" (S. 34).
0x19001901	0	System	INT	Trajectory Execution	Ausführung des Dynamikprofils 0 = Dynamikprofil wird sofort ausgeführt (Standard) 1 = Dynamikprofil wird vor Ausführung in einem Zwischenspeicher abgelegt Details siehe "Zyklische Übertragung von Zielpositionen" (S. 39).
0x19001902	3	System	INT	Maximum Number of Trajectory Points	Maximale Anzahl der Dynamikprofilpunkte Details siehe "Zyklische Übertragung von Zielpositionen" (S. 39).
0x19001903	0	System	INT	Threshold for Trajectory Execution	Schwellenwert für Ausführung des Dynamikprofils Details siehe "Zyklische Übertragung von Zielpositionen" (S. 39).
0x19001904	3	System	INT	Current Number of Trajectory Points	Zeigt die aktuelle Anzahl der Dynamikprofilpunkte im Zwischenspeicher an. Details siehe "Zyklische Übertragung von Zielpositionen" (S. 39).
0x19002000	0	System	INT	Configure Command Mode	Auswahl der Steuerungsquelle für die Achsen der Bewegungsplattform des Hexapods (X, Y, Z, U, V, W) 0 = GCS, siehe "Bewegungen des Hexapods" (S. 34) 1 = External: EtherCAT-Master. Nur möglich, wenn der C-887 eine EtherCAT-Feldbus-Schnittstelle hat.

Parameter-ID (hexa-dezimal)	Befehls-ebene für den Schreib-zugriff	Betroffener Elementtyp	Daten-typ	Parameter-name (Einheit)	Beschreibung
0x19002001	0	System	INT	Start All Hexapod Wave Generators	Startverhalten der Funktionsgeneratoren für die Achsen der Bewegungsplattform des Hexapods (X, Y, Z, U, V, W). Details siehe "Funktionsgenerator" (S. 102).
0x19002010	3	System	FLOAT	Cycletime For Interpolation In External Mode	Aktuell gültige Zykluszeit des C-887 während der Steuerung durch einen EtherCAT-Master.
0x19003000	1	Hexapod-Beine 1 bis 6	FLOAT	Hexapod Position Vector A0_X	Positions- und Richtungsvektoren für die Beine und Gelenke des Hexapods
0x19003001	2	Hexapod-Beine 1 bis 6	FLOAT	Hexapod Position Vector A0_Y	
0x19003002	1	Hexapod-Beine 1 bis 6	FLOAT	Hexapod Position Vector A0_Z	
0x19003010	1	Hexapod-Beine 1 bis 6	FLOAT	Hexapod Joint Vector A0_X	
0x19003011	1	Hexapod-Beine 1 bis 6	FLOAT	Hexapod Joint Vector A0_Y	
0x19003012	1	Hexapod-Beine 1 bis 6	FLOAT	Hexapod Joint Vector A0_Z	
0x19003100	1	Hexapod-Beine 1 bis 6	FLOAT	Hexapod Position Vector B0_X	
0x19003101	1	Hexapod-Beine 1 bis 6	FLOAT	Hexapod Position Vector B0_Y	
0x19003102	1	Hexapod-Beine 1 bis 6	FLOAT	Hexapod Position Vector B0_Z	
0x19003110	1	Hexapod-Beine 1 bis 6	FLOAT	Hexapod Joint Vector B0_X	
0x19003111	1	Hexapod-Beine 1 bis 6	FLOAT	Hexapod Joint Vector B0_Y	

Parameter-ID (hexa-dezimal)	Befehls-ebene für den Schreib-zugriff	Betroffener Elementtyp	Daten-typ	Parameter-name (Einheit)	Beschreibung
0x19003112	1	Hexapod-Beine 1 bis 6	FLOAT	Hexapod Joint Vector B0_Z	
0x19003200	2	Hexapod-Beine 1 bis 6	FLOAT	Hexapod Joint GLA	
0x19003201	2	Hexapod-Beine 1 bis 6	FLOAT	Hexapod Joint GLB	
0x19003202	2	Hexapod-Beine 1 bis 6	FLOAT	Hexapod Joint Type	
0x19003203	1	Hexapod-Beine 1 bis 6	FLOAT	Hexapod Spindle Pitch	
0x19004000	1	System	INT	Check PowerGood Signal	Prüfung des Power-Good-Signals 0 = Power-Good-Signal wird nicht geprüft 1 = Power-Good-Signal wird geprüft Details siehe "Automatisches Ausschalten des Servomodus / Stoppen der Bewegung" (S. 91).
0x19004001	1	System	INT	Axis Handling on Motion Error	Auswahl der im Fehlerfall zu behandelnden Achsen 0 = Behandlung nur für die betroffene Achse 1 = Behandlung für alle Achsen (Standard) Details siehe "Automatisches Ausschalten des Servomodus / Stoppen der Bewegung" (S. 91).
0x19004002	1	System	INT	Type of Axis Handling on Motion Error	Art der Behandlung im Fehlerfall 0 = Stoppen der Bewegung durch STP (Fehlercode 10 wird gesetzt), Servomodus bleibt eingeschaltet 1 = Stoppen der Bewegung durch Ausschalten des Servomodus (Standard) Details siehe "Automatisches Ausschalten des Servomodus / Stoppen der Bewegung" (S. 91).

Parameter-ID (hexa-dezimal)	Befehls-ebene für den Schreib-zugriff	Betroffener Elementtyp	Daten-typ	Parameter-name (Einheit)	Beschreibung
0x19005000	1	System	INT	Stiffness Compen-sation	Steifigkeitskompensation. Die Steifigkeitskompensation korrigiert bei Bewegungsbefehlen die aktuelle Position der Bewegungsplattform des Hexapods. Die Korrektur erfolgt in Abhängigkeit von der definierten Last und Steifigkeit und basiert auf einem theoretischen Modell. Für die Verwendung der Steifigkeitskompensation sind neben der Anpassung der Parameterwerte zusätzliche Konfigurationsdateien erforderlich. Wenn Sie die Steifigkeitskompensation verwenden möchten, wenden Sie sich an den Kundendienst (S. 367).
0x19005001	1	System	FLOAT	Mass Of Load	
0x19005002	1	System	FLOAT	Center Of Gravity	
0x19005003	1	System	FLOAT	Tensor Matrix	
0x19005004	1	System	FLOAT	Direction To GeoCenter	
0x19006000	0	System	FLOAT	Reduction Factor for TRA? Response	Faktor für die Begrenzung der Antwort auf TRA? 0 bis 1 Der Parameter beeinflusst **nicht** die tatsächlichen Verfahrbereichsgrenzen, sondern begrenzt nur die durch TRA? (S. 273) angezeigten Positionswerte. Details siehe "Stellweg und Verfahrbereichsgrenzen" (S. 34).
0x19006001	2	System	INT	Enable Gradient Search	Verfügbarkeit von Fast-Alignment-Routinen zur Gradientensuche (Befehl FDG) 0 = Gradientensuche nicht verfügbar 1 = Gradientensuche verfügbar Siehe auch die Option R-FMP-GSM in "Optionales Zubehör" (S. 24).
0x19006002	0	System	INT	Check Calculated Trajectory Before Motion	Aktivierungszustand der Prüfung des Dynamikprofils 0 = Prüfung deaktiviert 1 = Prüfung aktiviert Details siehe "Profilgenerator für Punkt-zu-Punkt-Bewegungen" (S. 37) und "Störungsbehebung" (S. 357).

Parameter-ID (hexa-dezimal)	Befehls-ebene für den Schreib-zugriff	Betroffener Elementtyp	Daten-typ	Parameter-name (Einheit)	Beschreibung
0x1A002000	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Max Enc Vel	Sicherheitsabschaltung im Fehlerfall Details siehe "Sicherheitsabschaltung konfigurieren" (S. 97).
0x1A002100	0	Hexapod-Beine 1 bis 6, Achsen A und B	INT	Activate Motor Stuck Check	
0x1A002200	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Minimal MotorOut to Move Axis	
0x1A002300	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Velocity Threshold under which Axis is considered not moving	
0x1A002400	0	Hexapod-Beine 1 bis 6, Achsen A und B	FLOAT	Time Period for which Axis is not yet considered not moving	
0x20001C00	1	System	INT	Fast Alignment Axis Signal Type	Typ der Positionssignale für Fast-Alignment-Routinen Fast-Alignment-Routinen führen Berechnungen auf Basis der Position der beteiligten Achsen aus. Mögliche Typen: 0 = aktuelle Position, basierend auf Encodersignalen 1 = Dynamikprofil, erzeugt vom Profilgenerator des C-887 (Standard) Die Verwendung der aktuellen Position kann zum Beispiel sinnvoll sein, wenn am Ende einer Area-Scan-Routine das gefundene Signalmaximum angefahren werden soll. Die Abweichung beim Anfahren des Signalmaximums kann in diesem Fall geringfügig reduziert sein im Vergleich zur Abweichung bei Verwendung des Dynamikprofils. Details siehe E712T0016 "Schnelle, mehrkanalige Ausrichtung in der Photonik".

10 Wartung

In diesem Kapitel

C-887 reinigen 339
Firmware und Konfigurationsdateien aktualisieren 340
Hexapod warten und prüfen 349

10.1 C-887 reinigen

HINWEIS

Kurzschlüsse oder Überschläge!
Der C-887 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-887 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-887 mit einem Tuch, das leicht mit einem milden Reinigungs- oder Desinfektionsmittel angefeuchtet wurde.

10.2 Firmware und Konfigurationsdateien aktualisieren

10.2.1 Allgemeine Hinweise zur Aktualisierung von Firmware und Konfigurationsdateien

HINWEIS

Schäden durch ungewollte Positionsänderungen!

Das Simulationsprogramm PIVirtualMove berechnet die zulässigen Posen auf Basis der maximalen Nutzlast des Hexapods (= Grenzwert bei eingeschaltetem Servomodus). Die maximale Haltekraft bei ausgeschaltetem Servomodus basiert auf der Selbsthemmung der Aktoren in den Hexapod-Beinen und fällt geringer aus als die maximale Nutzlast.

Wenn die tatsächliche Belastung des Hexapods die auf der Selbsthemmung der Aktoren basierende maximale Haltekraft überschreitet, können in folgenden Fällen ungewollte Positionsänderungen des Hexapods auftreten, obwohl PIVirtualMove die aktuelle Pose als zulässig eingestuft hat:

- Ausschalten des C-887
- Neustart des C-887
- Ausschalten des Servomodus für die Achsen der Bewegungsplattform des Hexapods, z. B. durch Verwendung der Buchse **E-Stop** (S. 93)

Dadurch sind Kollisionen zwischen Hexapod, zu bewegender Last und Umgebung möglich. Kollisionen können den Hexapod, die zu bewegende Last oder die Umgebung beschädigen.

➢ Stellen Sie sicher, dass die tatsächliche Belastung der Bewegungsplattform des Hexapods die auf der Selbsthemmung der Aktoren basierende maximale Haltekraft nicht überschreitet, bevor Sie den Servomodus ausschalten, den C-887 neu starten oder ausschalten.

HINWEIS

Fehlfunktion nach Ausschalten während Aktualisierung der Firmware!

Wenn Sie den C-887 während der Aktualisierung der Firmware ausschalten, werden Firmware und Konfigurationsdateien unvollständig oder fehlerhaft auf den C-887 übertragen. Unvollständige oder fehlerhafte Firmware und Konfiguration kann das Booten des C-887 verhindern.

- Schalten Sie den C-887 während der Aktualisierung der Firmware **nicht** aus.
- Notieren Sie sich beim Aufbau der Verbindung mit dem PIFirmwareManager die IP-Adresse des C-887.
- Wenn der C-887 während der Aktualisierung der Firmware ausgeschaltet wurde und anschließend noch bootet, wiederholen Sie die Aktualisierung der Firmware.
- Wenn der C-887 während der Aktualisierung der Firmware ausgeschaltet wurde und anschließend nicht mehr bootet:
  - Schalten Sie den C-887 **nicht** aus.
  - Kontaktieren Sie unseren Kundendienst (S. 367).

INFORMATION

Die Firmware des C-887 besteht aus mehreren Komponenten, die mit dem Programm PIFirmwareManager separat aktualisiert werden können. Die Kombination aus den Versionen der einzelnen Komponenten muss von PI freigegeben sein.

- ➢ Bevor Sie die Firmware des C-887 aktualisieren, prüfen Sie durch Abfrage mit VER? (S. 278) die Versionen der Firmwarekomponenten.
- ➢ Wenden Sie sich für Informationen zu freigegebenen Versions-Kombinationen an unseren Kundendienst (S. 367).
- ➢ Aktualisieren Sie die einzelnen Firmwarekomponenten in separaten Schritten. Wenn unser Kundendienst eine bestimmte Reihenfolge für die Aktualisierung der Komponenten vorgegeben hat, halten Sie diese Reihenfolge ein.

INFORMATION

Für die Aktualisierung der Firmware müssen C-887 und PC über die TCP/IP-Schnittstelle kommunizieren.

10.2.2 Aktuelle Firmware und Konfigurationsdateien beziehen

Aktuelle Firmware und Konfigurationsdateien des C-887 beziehen

1. Halten Sie folgende Informationen bereit:
   - Lesen Sie mit dem Befehl `*IDN?` die Ident-Bezeichnung des C-887 aus.
   - Lesen Sie mit dem Befehl `VER?` die Versionen aller Firmwarekomponenten aus.
   - Lesen Sie mit dem Befehl `CST?` den Hexapod-Typ aus, auf den der Controller abgestimmt ist.

2. Wenden Sie sich an unseren Kundendienst (S. 367), um aktuelle Versionen der Firmwarekomponenten und Konfigurationsdateien sowie Informationen zu freigegebenen Versions-Kombinationen zu erhalten.

10.2.3 Firmware aktualisieren

Werkzeug und Zubehör

- PC mit Windows-Betriebssystem, der wie folgt vorbereitet wurde:
  - Das Programm PIFirmwareManager ist installiert. Weitere Informationen siehe "PC-Software installieren" (S. 58).
  - Die aktuellen Firmware- und/oder Konfigurationsdateien, die Sie von unserem Kundendienst erhalten haben (S. 341), liegen in einem Verzeichnis auf dem PC.

Voraussetzungen

✓ Sie haben die allgemeinen Hinweise für das Aktualisieren von Firmware und Konfigurationsdateien gelesen und verstanden (S. 340).

✓ Sie haben alle erforderlichen Vorbereitungen für die Kommunikation über die TCP/IP-Schnittstelle getroffen, siehe "Kommunikation über TCP/IP in der PC-Software herstellen" (S. 79).

✓ Der C-887 ist eingeschaltet, und der Startvorgang des C-887 ist beendet (S. 74).

✓ Der PC ist eingeschaltet.

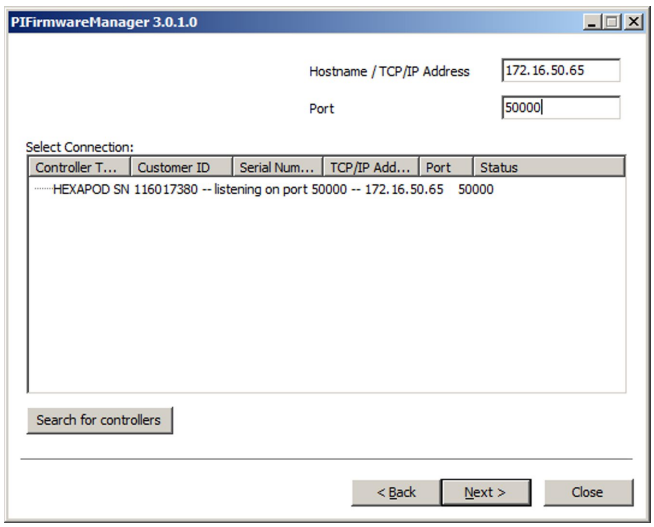
Firmware des C-887 aktualisieren

1. Starten Sie das Programm PIFirmwareManager am PC über den Startmenü-Eintrag ***Alle Programme > PI > PIFirmwareManager.***

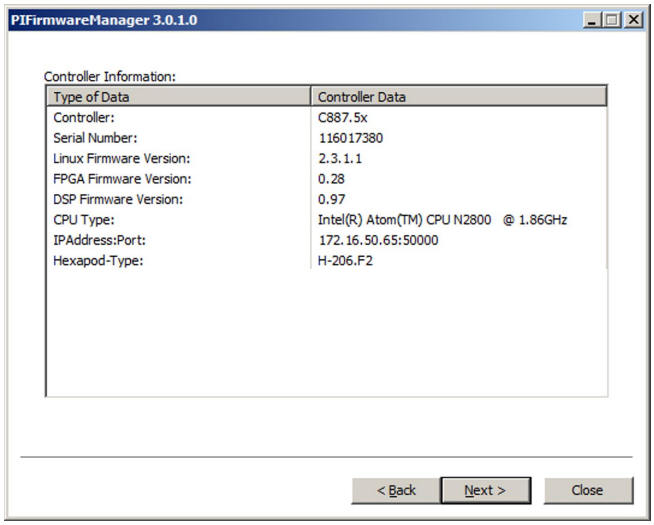
   Das Fenster ***PIFirmwareManager*** öffnet sich.

   - Wählen Sie die Schaltfläche ***Next >***.

2. Stellen Sie die Kommunikation zwischen dem C-887 und dem PC her.
   a) Wählen Sie die Schaltfläche ***Search for controllers***.
   b) Wenn Ihr C-887 im Feld ***Select Connection:*** angezeigt wird, wählen Sie die entsprechende Zeile in der Liste.
   c) Wenn Ihr C-887 nicht im Feld ***Select Connection:*** angezeigt wird, aber seine aktuelle IP-Adresse bekannt ist, geben Sie die IP-Adresse im Feld ***Hostname / TCP/IP Address*** ein, und den Port *50000* im Feld ***Port***.
   d) Wählen Sie die Schaltfläche ***Next >***.

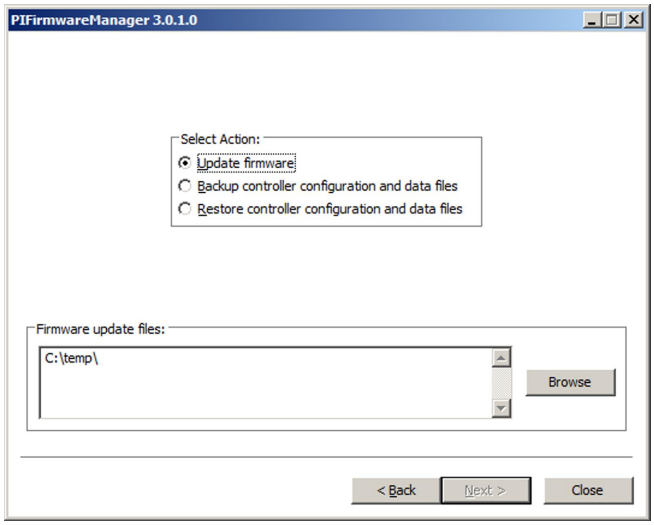


3. Prüfen Sie, ob die angezeigten Informationen zu den Informationen passen, die Sie zusammen mit der neuen Firmware von unserem Kundendienst erhalten haben.

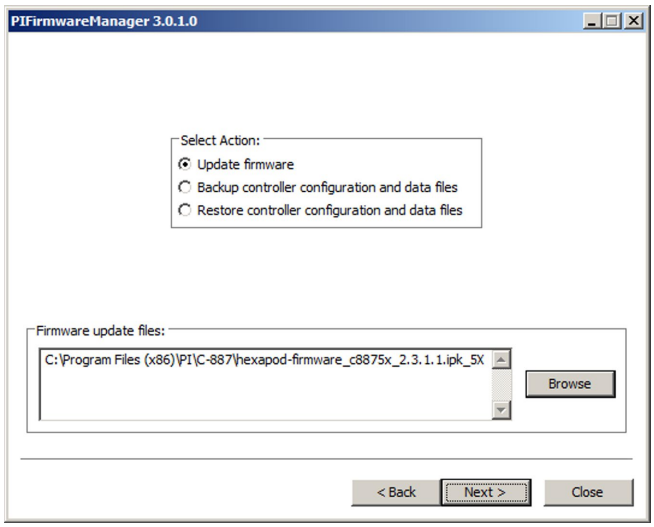


   – Wählen Sie die Schaltfläche ***Next >***.

4. Wählen Sie im Feld ***Select Action:*** die Option ***Update firmware***.



5. Wählen Sie die Datei für die zu aktualisierende Firmwarekomponente aus.
   a) Wählen Sie die Schaltfläche ***Browse***. Ein Dateiauswahldialog öffnet sich.
   b) Wechseln Sie im Dateiauswahldialog in das Verzeichnis, in dem die Dateien liegen, die Sie vom Kundendienst erhalten haben.
   c) Wählen Sie im Dateiauswahldialog die Datei für die zu aktualisierende Firmwarekomponente aus, z. B. hexapod-firmware_c8875x_x.x.x.x.ipk_5X.
   d) Wählen Sie im Dateiauswahldialog die Schaltfläche ***Öffnen***, um die Auswahl zu bestätigen. Die ausgewählte Datei wird im PIFirmwareManager im Feld ***Firmware update files:*** angezeigt.
6. Starten Sie die Übertragung der Firmware auf den C-887.

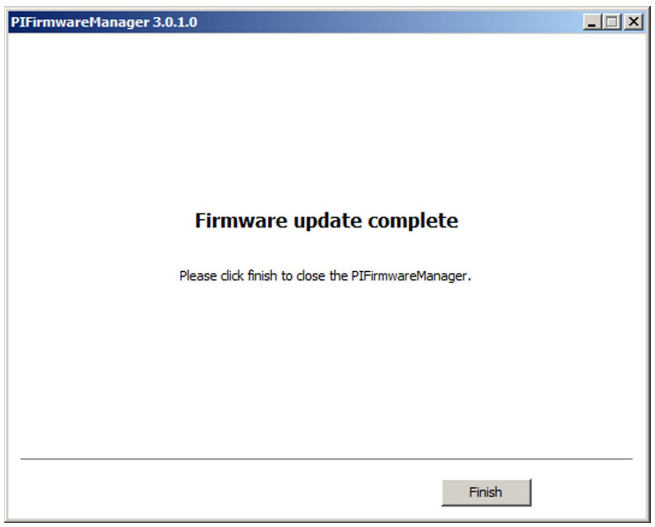


a) Wählen Sie die Schaltfläche ***Next >***. Ein Dialogfenster öffnet sich.

b) Wählen Sie im Dialogfenster ***Ja***, um die Übertragung der Firmware auf den C-887 zu starten.

Der Fortschritt der Aktualisierung wird angezeigt. Die Aktualisierung ist beendet, wenn der C-887 einen Neustart durchgeführt hat.

7. Wählen Sie die Schaltfläche ***Finish***.



Der PIFirmwareManager schließt sich.

8. Wenn Sie eine weitere Firmwarekomponente des C-887 aktualisieren wollen, wiederholen Sie die Schritte 1 bis 7.

10.2.4 Konfigurationsdateien aktualisieren

HINWEIS

Fehlfunktion nach fehlerhafter Änderung von Konfigurationsdateien!
Wenn Sie ein falsches oder unvollständiges tar-Archiv auf den C-887 übertragen, können Sie den C-887 funktionsunfähig machen.

- Notieren Sie sich beim Aufbau der Verbindung mit dem PIFirmwareManager die IP-Adresse des C-887.
- Wenn Sie den Inhalt des tar-Archivs auf dem PC bearbeiten, ändern Sie nicht die Verzeichnisstruktur und die Namen von Verzeichnissen und vorhandenen Dateien.
- Ändern und ergänzen Sie Konfigurationsdateien im tar-Archiv auf dem PC nur nach Rücksprache mit PI.
- Schalten Sie den C-887 während der Rückübertragung des tar-Archivs nicht aus.
- Wenn fehlerhafte Konfigurationsdateien auf den C-887 übertragen wurden:
  - Schalten Sie den C-887 **nicht** aus.
  - Kontaktieren Sie unseren Kundendiens (S. 367)t.

Sie können verschiedene Konfigurationsdateien des C-887 aktualisieren und zusätzliche Konfigurationsdateien auf den C-887 übertragen.

Der direkte Zugriff auf die Konfigurationsdateien des C-887 ist nicht möglich. Sie müssen zunächst die Konfigurationsdateien des C-887 als tar-Archiv auf den PC kopieren. Im tar-Archiv auf dem PC aktualisieren und ergänzen Sie die Konfigurationsdateien. Anschließend übertragen Sie das geänderte tar-Archiv vom PC zurück auf den C-887.

INFORMATION

Anwendungstipp:
Sie können einen C-887 "duplizieren", indem Sie seine Konfigurationsdateien als tar-Archiv auf den PC kopieren und anschließend ohne Änderung vom PC auf einen anderen C-887 übertragen. Die Seriennummer des C-887, auf den die kopierten Daten übertragen werden, bleibt dabei unverändert.

Konfigurationsdateien des C-887 aktualisieren

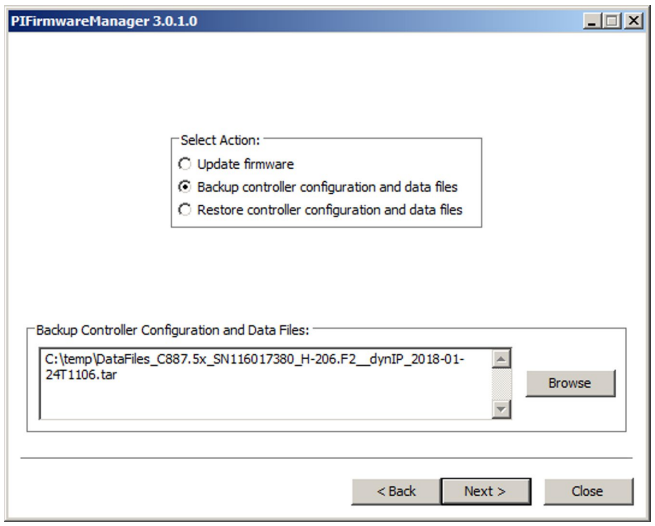
1. Um das tar-Archiv mit den Konfigurationsdateien des C-887 auf den PC zu kopieren, starten Sie den PIFirmwareManager. Führen Sie dazu die Schritte 1 bis 3 der Anleitung "Firmware aktualisieren" (S. 342) aus.
2. Wählen Sie im Feld ***Select Action:*** die Option ***Backup controller configuration and data files***.

Im Feld ***Backup Controller Configuration and Data Files*** wird ein Vorschlag für den Verzeichnispfad auf dem PC und den Namen des tar-Archivs angezeigt.

– Prüfen Sie, ob das vorgeschlagene Verzeichnis auf dem PC existiert.

– Wenn Sie das Verzeichnis auf dem PC und/oder den Namen des tar-Archivs ändern wollen: Wählen Sie die Schaltfläche ***Browse*** und wählen Sie ein neues Verzeichnis und/oder einen neuen Namen.

– Ändern Sie **nicht** die Dateiendung .tar.



3. Starten Sie die Übertragung des tar-Archivs auf den PC, indem Sie die Schaltfläche ***Next >*** wählen.

   Der Fortschritt der Übertragung wird angezeigt. Die Übertragung auf den PC ist beendet, wenn im PIFirmwareManager *Backup of controller data files complete* angezeigt wird.

4. Wählen Sie ***Finish***, um den PIFirmwareManager zu schließen.

5. Aktualisieren und ergänzen Sie die Konfigurationsdateien im tar-Archiv auf dem PC.

   a) Öffnen Sie auf dem PC das Verzeichnis, in dem der PIFirmwareManager das tar-Archiv mit den Kopien der Konfigurationsdateien abgelegt hat.

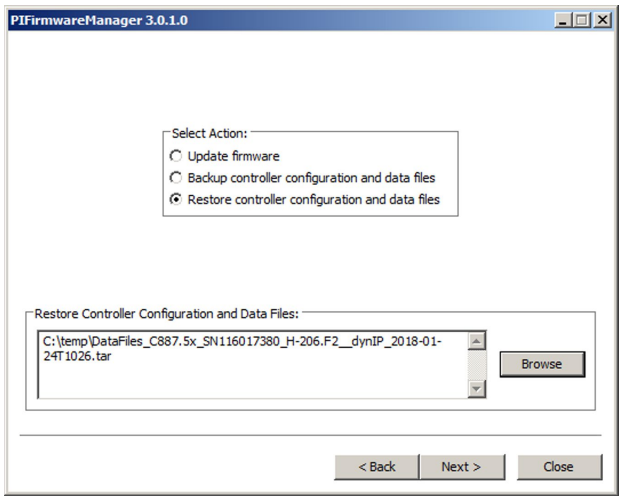
   b) Aktualisieren und ergänzen Sie die Konfigurationsdateien im tar-Archiv gemäß den Anweisungen, die Sie zusammen mit den neuen Konfigurationsdateien von unserem Kundendienst erhalten haben.

6. Um die Rückübertragung des geänderten tar-Archivs vom PC auf den C-887 vorzunehmen, starten Sie erneut den PIFirmwareManager. Führen Sie dazu die Schritte 1 bis 3 der Anleitung "Firmware aktualisieren" (S. 342) aus.

7. Wählen Sie im Feld ***Select Action:*** die Option ***Restore controller configuration and data files***.

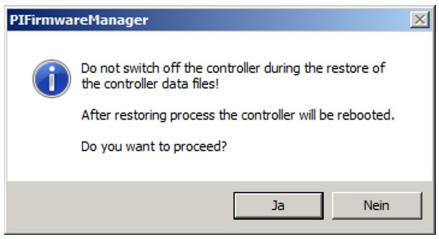
8. Wählen Sie auf dem PC das zu übertragende tar-Archiv aus.

– Wählen Sie die Schaltfläche ***Browse***.



9. Starten Sie die Rückübertragung des tar-Archivs auf den C-887.

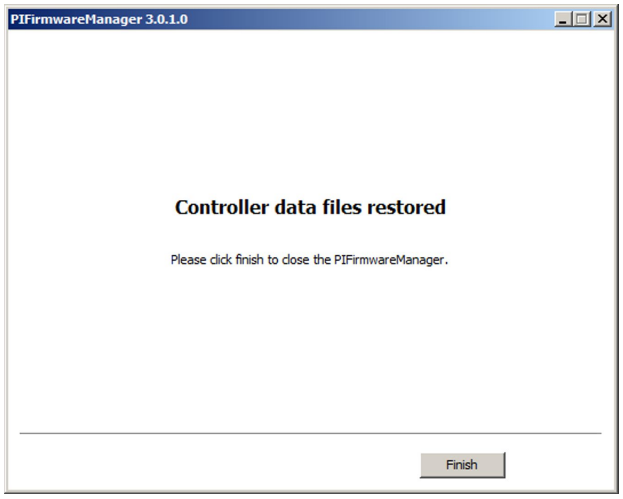
   a) Wählen Sie die Schaltfläche ***Next >***. Ein Dialogfenster öffnet sich.



   b) Wählen Sie im Dialogfenster ***Ja***, um die Übertragung des tar-Archivs auf den C-887 zu starten.

   Der Fortschritt der Rückübertragung wird angezeigt. Die Rückübertragung ist beendet, wenn der C-887 einen Neustart durchgeführt hat.

10. Wählen Sie die Schaltfläche ***Finish***.



Der PIFirmwareManager schließt sich.

10.3 Hexapod warten und prüfen

10.3.1 Wartungsfahrt durchführen

VORSICHT

Quetschgefahr durch bewegte Teile!

Zwischen den bewegten Teilen des Hexapods und einem feststehenden Teil oder Hindernis besteht die Gefahr von leichten Verletzungen durch Quetschung.

- Halten Sie Ihre Finger von Bereichen fern, in denen sie von bewegten Teilen erfasst werden können.

HINWEIS

Schäden durch Kollisionen!

Kollisionen können den Hexapod, die zu bewegende Last und die Umgebung beschädigen.

- Stellen Sie sicher, dass im Arbeitsraum des Hexapods keine Kollisionen zwischen Hexapod, zu bewegender Last und Umgebung möglich sind.
- Platzieren Sie keine Gegenstände in Bereichen, in denen sie von bewegten Teilen erfasst werden können.
- Kommandieren Sie nur zulässige Posen. Für die Definition einer zulässigen Pose siehe "Zulässige Posen ermitteln" (S. 62).
- Halten Sie bei einer Fehlfunktion des Controllers die Bewegung sofort an.

Häufige Bewegungen über einen eingeschränkten Stellweg können dazu führen, dass das Schmiermittel auf den Spindeln der Hexapod-Beine ungleichmäßig verteilt ist. Daraus kann ein erhöhter Verschleiß resultieren. Eine Wartungsfahrt über den gesamten Stellweg der Hexapod-Beine verteilt das Schmiermittel wieder gleichmäßig.

INFORMATION

Je öfter Bewegungen über einen eingeschränkten Stellweg durchgeführt werden, desto kürzer muss der zeitliche Abstand der Wartungsfahrten sein.

- Führen Sie die Wartungsfahrt entsprechend den Anforderungen Ihrer Anwendung in regelmäßigen Abständen durch.

Voraussetzungen

- ✓ Sie haben die allgemeinen Hinweise zur Inbetriebnahme gelesen und verstanden (S. 69).
- ✓ Sie haben das Hexapod-System korrekt installiert (S. 57).
- ✓ Sie haben das Benutzerhandbuch des Hexapods gelesen und verstanden.
- ✓ Wenn Sie die Wartungsfahrt mit PIMikroMove® durchführen wollen:
  - PIMikroMove® ist auf dem PC installiert (S. 58).
  - Sie haben das PIMikroMove®-Handbuch gelesen und verstanden. Links auf die Softwarehandbücher finden Sie in der Datei A000T0081 auf dem Datenträger der PI Software Suite.
  - Sie haben die Kommunikation zwischen dem C-887 und dem PC mit PIMikroMove® über die TCP/IP-Schnittstelle (S. 79) oder die RS-232-Schnittstelle (S. 83) hergestellt.
  - Sie haben den Abschnitt "Bewegungen starten" (S. 84) gelesen und verstanden.

Wartungsfahrt durchführen

1. Stellen Sie die Geschwindigkeit für die Bewegungsplattform des Hexapods mit dem Befehl `VLS` (S. 279) auf circa 30 % des nach dem Einschalten oder Neustart des C-887 gültigen Standardwerts ein.
2. Kommandieren Sie eine Bewegung des Hexapods zu folgender Zielposition:

   Z = größte kommandierbare Position (Abfrage mit dem Befehl `TMX?` (S. 272) möglich)

   X = Y = U = V = W = 0
3. Kommandieren Sie eine Bewegung des Hexapods zu folgender Zielposition:

   Z = kleinste kommandierbare Position (Abfrage mit dem Befehl `TMN?` (S. 271) möglich)

   X = Y = U = V = W = 0
4. Wiederholen Sie die Schritte 2 und 3 jeweils dreimal.
5. Kommandieren Sie eine Bewegung des Hexapods zur Zielposition
   X = Y = Z = U = V = W = 0.

10.3.2 Beintest durchführen

VORSICHT

Quetschgefahr durch bewegte Teile!

Zwischen den bewegten Teilen des Hexapods und einem feststehenden Teil oder Hindernis besteht die Gefahr von leichten Verletzungen durch Quetschung.

➢ Halten Sie Ihre Finger von Bereichen fern, in denen sie von bewegten Teilen erfasst werden können.

HINWEIS

Schäden durch Kollisionen während des Beintests!

Während eines Beintests bewegt sich der Hexapod auf unvorhersehbare Weise. Es findet **keine** Kollisionsprüfung und -vermeidung statt, selbst wenn eine Konfiguration zur Kollisionsvermeidung auf dem C-887 abgelegt wurde (z. B. mit der kostenpflichtigen PIVeriMove Hexapod Software zur Kollisionsprüfung). Verfahrbereichsgrenzen, die mit den Befehlen NLM (S. 248) und PLM (S. 251) für die Bewegungsplattform des Hexapods gesetzt wurden, werden während eines Beintests ignoriert.

Dadurch sind Kollisionen zwischen Hexapod, zu bewegender Last und Umgebung möglich. Kollisionen können den Hexapod, die zu bewegende Last und die Umgebung beschädigen.

- Stellen Sie sicher, dass während eines Beintests des Hexapods keine Kollisionen zwischen Hexapod, zu bewegender Last und Umgebung möglich sind.
- Platzieren Sie keine Gegenstände in Bereichen, in denen sie während eines Beintests von bewegten Teilen erfasst werden können.
- Beaufsichtigen Sie den Hexapod während eines Beintests, um bei Störungen schnell eingreifen zu können.

HINWEIS

Schäden durch ungewollte Positionsänderungen während des Beintests!

Während eines Beintests kann das Hexapod-Bein auf einen Endschalter fahren. Dadurch wird automatisch der Servomodus für die Achsen der Bewegungsplattform des Hexapods ausgeschaltet.

Wenn die tatsächliche Belastung des Hexapods die auf der Selbsthemmung der Aktoren basierende maximale Haltekraft überschreitet, kann das Ausschalten des Servomodus für die Achsen der Bewegungsplattform des Hexapods ungewollte Positionsänderungen des Hexapods verursachen.

Dadurch sind Kollisionen zwischen Hexapod, zu bewegender Last und Umgebung möglich. Kollisionen können den Hexapod, die zu bewegende Last und die Umgebung beschädigen.

- Stellen Sie sicher, dass die tatsächliche Belastung der Bewegungsplattform des Hexapods die auf der Selbsthemmung der Aktoren basierende maximale Haltekraft nicht überschreitet, bevor Sie einen Beintest starten.

INFORMATION

Empfehlungen für die Ausführung von Beintests:

- Wenn möglich, kommandieren Sie den Hexapod vor einem Beintest in die Referenzposition, um für jedes Hexapod-Bein einen möglichst großen Stellweg in die positive und negative Bewegungsrichtung zur Verfügung zu haben.
- Stellen Sie die Geschwindigkeit vor einem Beintest mit dem Befehl `VLS` (S. 279) passend ein.

Wenn Störungen des Hexapod-Systems auftreten, z. B. Bewegungsfehler (Fehlercode 1024), können folgende Tests der Hexapod-Beine die Fehlerdiagnose erleichtern:

- Messung der Impulsantwort des Hexapod-Beins. Anzeichen für Störungen sind:
  - Starkes Überschwingen der aktuellen Position
  - Schwingungen der aktuellen Position
  - Deutliche Abweichung zwischen den Kurven der aktuellen Position und der Zielposition
- Messung der Motoransteuerung während einer Spindelumdrehung des Hexapod-Beins (dimensionsloser Wert von -32767 bis 32768; das Vorzeichen entspricht der Bewegungsrichtung); ermöglicht unter anderem die Beurteilung der Regelreserve. Anzeichen für Störungen sind:
  - Motoransteuerung dauerhaft >70 % des Maximalwerts
  - Ungewöhnlich hoher Wert der Motoransteuerung an einem bestimmten Punkt des Stellwegs

Folgendes Hilfsmittel kann für die Tests genutzt werden:

- Fenster ***Hexapod Service Tools*** in PIMikroMove®

INFORMATION

Alternativ zur Impulsantwort können Sie auch eine Sprungantwort des Hexapod-Beins aufzeichnen und auswerten. Nach der Ausführung des Sprungs fährt das Hexapod-Bein jedoch **nicht** wieder zurück zur Ausgangsposition. Bei mehrfacher Ausführung eines Sprungs kann das Hexapod-Bein auf den Endschalter fahren. Beim Erreichen des Endschalters wird der Servomodus für die Achsen der Bewegungsplattform des Hexapods ausgeschaltet.

Voraussetzungen

- ✓ Sie haben die allgemeinen Hinweise zur Inbetriebnahme gelesen und verstanden (S. 69).
- ✓ Sie haben das Hexapod-System korrekt installiert (S. 57).
- ✓ Sie haben das Benutzerhandbuch des Hexapods gelesen und verstanden.
- ✓ PIMikroMove® ist auf dem PC installiert (S. 58).
- ✓ Sie haben das PIMikroMove®-Handbuch gelesen und verstanden. Links auf die Softwarehandbücher finden Sie in der Datei A000T0081 auf dem Datenträger der PI Software Suite.
- ✓ Sie haben die Kommunikation zwischen dem C-887 und dem PC mit PIMikroMove® über die TCP/IP-Schnittstelle (S. 79) oder die RS-232-Schnittstelle (S. 83) hergestellt.

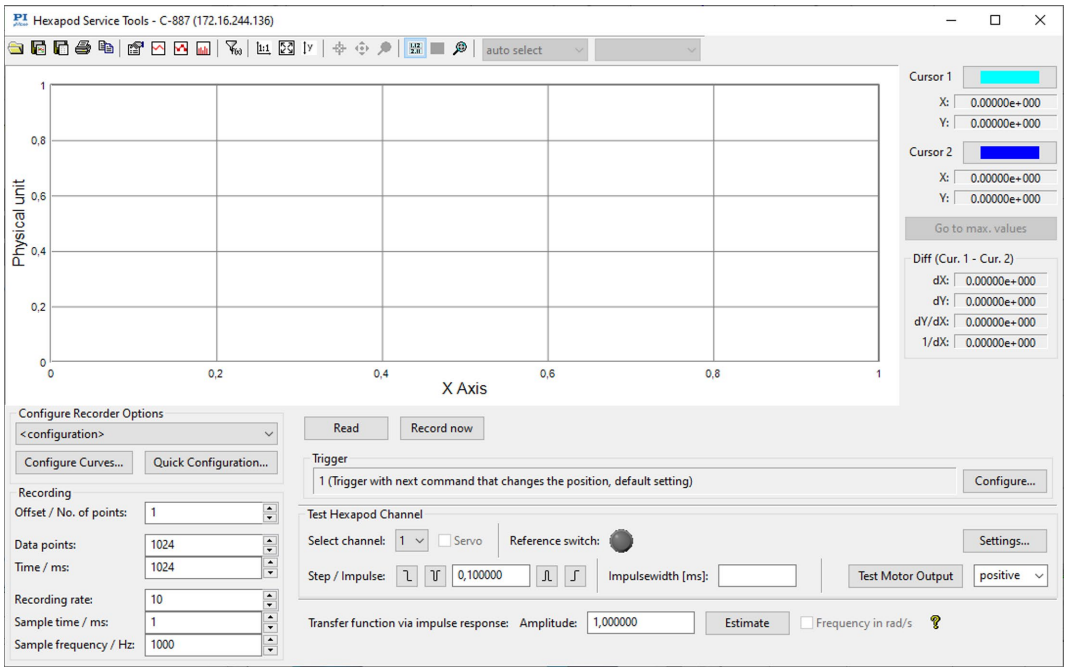
Beintest im Fenster *Hexapod Service Tools* von PIMikroMove® durchführen

1. Wählen Sie im Hauptfenster von PIMikroMove® den Menüeintrag ***C-887 > Show hexapod service tools...*** .

   Ein Fenster mit dem Hinweis auf mögliche Sachschäden bei Verwendung des Fensters ***Hexapod Service Tools*** öffnet sich.

2. Wählen Sie im Hinweisfenster ***Show service tools***.

   Das Fenster ***Hexapod Service Tools*** öffnet sich.



3. Zeichnen Sie im Fenster ***Hexapod Service Tools*** eine Impulsantwort für ein Hexapod-Bein auf:

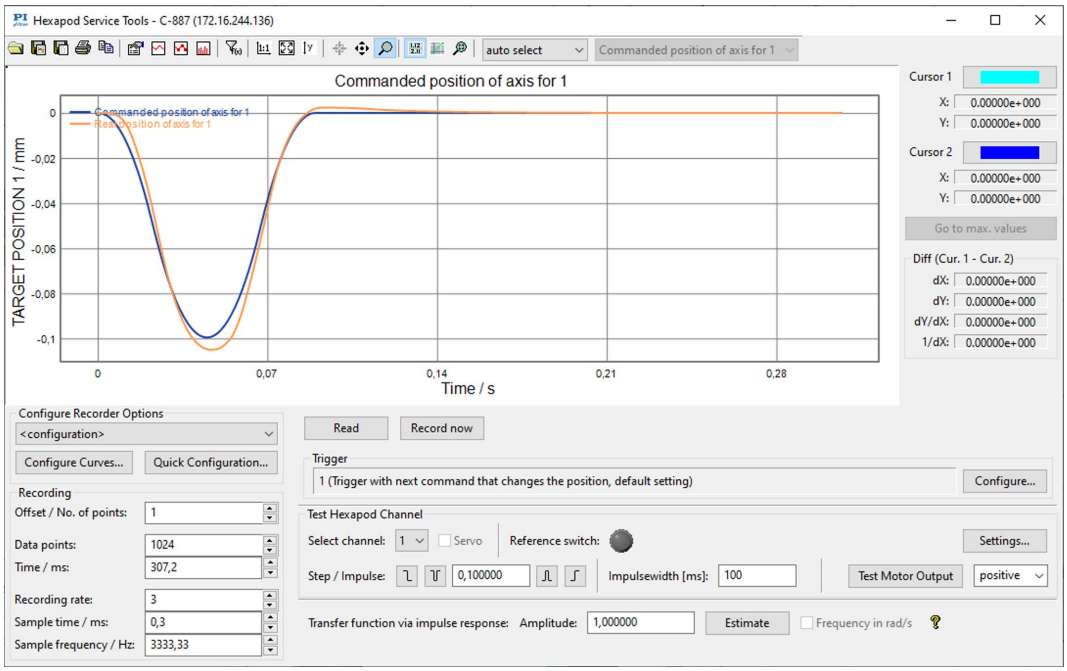
   a) Wählen Sie im Feld ***Select channel*** das Bein aus, für das die Impulsantwort aufgezeichnet werden soll.

   b) Geben Sie im Feld ***Impulsewidth [ms]*** die Pulsweite des Impulses in Millisekunden ein.

   c) Starten Sie den Impuls in positiver oder negativer Richtung mit der Schaltfläche oder .
   Das Starten des Impulses startet auch die Aufzeichnung der aktuellen Position und der Zielposition des Hexapod-Beins. Das Hexapod-Bein bewegt sich entsprechend der eingestellten Pulsweite des Impulses und fährt anschließend zurück zur Ausgangsposition.

   d) Werten Sie die Impulsantwort anhand der Kurven im Grafikfeld des Fenster ***Hexapod Service Tools*** aus (siehe nachfolgende Abbildung).

Eine Sprungantwort mit entsprechender Aufzeichnung kann mit der Schaltfläche oder gestartet werden. Die Schrittweite des Sprungs wird durch den Eintrag im Feld zwischen den Schaltflächen zum Starten von Impulsen vorgegeben.



4. Zeichnen Sie im Fenster ***Hexapod Service Tools*** die Motoransteuerung während einer Spindelumdrehung eines Hexapod-Beins auf:

   a) Wählen Sie im Feld ***Select channel*** das Bein aus, für das die Motoransteuerung aufgezeichnet werden soll.

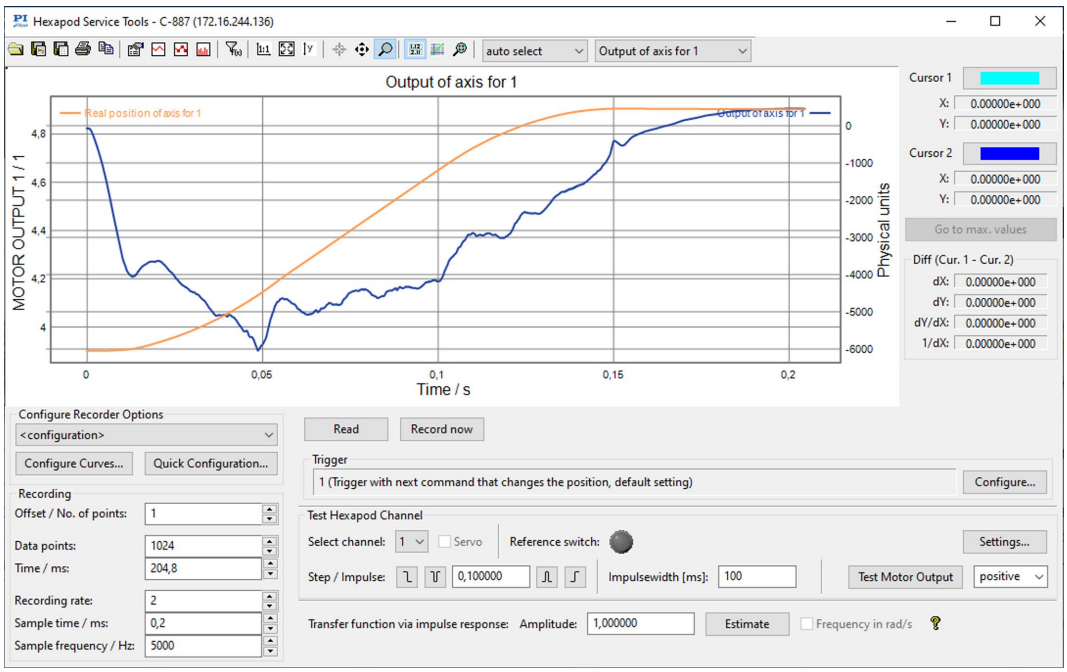
   b) Wählen Sie im Feld rechts neben der Schaltfläche ***Test Motor Output*** die Richtung (positiv oder negativ), in die die Spindelumdrehung ausgeführt werden soll.

   c) Starten Sie eine Spindelumdrehung des Hexapod-Beins in die ausgewählte Richtung, indem Sie die Schaltfläche ***Test Motor Output*** wählen.
   Das Starten der Spindelumdrehung startet auch die Aufzeichnung der aktuellen Motoransteuerung und der aktuellen Position des Hexapod-Beins. Das Hexapod-Bein fährt nach der Spindelumdrehung **nicht** zurück zur Ausgangsposition.

   d) Werten Sie die aufgezeichnete Motoransteuerung anhand der Kurve im Grafikfeld des Fenster ***Hexapod Service Tools*** aus (siehe nachfolgende Abbildung).

Bei mehrfacher Ausführung einer Spindelumdrehung in die gleiche Richtung kann das Hexapod-Bein auf den Endschalter fahren. Beim Erreichen des Endschalters wird der Servomodus für die Achsen der Bewegungsplattform des Hexapods ausgeschaltet.

Beachten Sie, dass die Motoransteuerung in der nachfolgenden Abbildung auf einer zweiten Y-Achse dargestellt ist, da Motoransteuerung und aktuelle Position unterschiedliche Einheiten haben. Die Zuweisung zur zweiten Y-Achse können Sie im Fenster ***Display Properties*** vornehmen, siehe das PIMikroMove®-Handbuch (SM148E).



5. Wenn die Testergebnisse auf Störungen hinweisen:
   a) Betreiben Sie das Hexapod-System **nicht** weiter.
   b) Übermitteln Sie die Testergebnisse für die Fehlerdiagnose an unseren Kundendienst (S. 367).

11 Störungsbehebung

Störung	Mögliche Ursachen	Behebung
C-887 bootet nicht	Stromversorgung nicht korrekt angeschlossen	➢ Prüfen Sie, ob der C-887 korrekt an die Stromversorgung angeschlossen ist (S. 61).
	Firmware und/oder Konfigurations-dateien unvoll-ständig oder fehlerhaft	1. Schalten Sie den C-887 nicht aus. 2. Versuchen Sie die zuletzt gültige IP-Adresse des C-887 zu ermitteln. 3. Kontaktieren Sie unseren Kundendienst (S. 367).

Störung	Mögliche Ursachen	Behebung
Der Hexapod bewegt sich nicht.	Kabel defekt oder nicht korrekt angeschlossen	➢ Prüfen Sie die Kabelanschlüsse.
Der Hexapod bewegt sich nicht.	Stromversorgung des Hexapods ist unterbrochen	➢ Überprüfen Sie das Stromversorgungskabel. ➢ Überprüfen Sie, falls vorhanden, das Netzteil des Hexapods. ➢ Überprüfen Sie das Power-Good-Signal des Hexapods. Optionen: – Senden Sie den Befehl `DIA?` (S. 165) – Öffnen Sie in PIMikroMove® das Fenster ***Diagnostic Information*** über den Menüeintrag ***C-887 > Show diagnostic information...***
Der Hexapod bewegt sich nicht.	Nur Modelle C-887.522, .523, .532, .533: Beschaltung der Buchse **E-Stop** verhindert Auslösen von Bewegungen	➢ Überprüfen Sie das Power-Good-Signal und den Aktivierungszustand des 24-V-Ausgangs für den Hexapod (**24 V Out 7 A**). Optionen: – Senden Sie den Befehl `DIA?` (S. 165) – Öffnen Sie in PIMikroMove® das Fenster ***Diagnostic Information*** über den Menüeintrag ***C-887 > Show diagnostic information...*** ➢ Verbinden Sie die Buchse **E-Stop** gemäß den Anforderungen Ihrer Anwendung mit externer Hardware, Details siehe "Buchse E-Stop verwenden" (S. 93).

Störung	Mögliche Ursachen	Behebung

Der Hexapod bewegt sich nicht.	Der Hexapod ist nicht mit einem Power-Good-Signal ausgestattet, aber die Prüfung des Power-Good-Signals ist aktiviert.	1. Senden Sie den Befehl `ERR?` und prüfen Sie den zurückgemeldeten Fehlercode.
		2. Wenn die Antwort auf `ERR?` den Fehlercode 500 oder 66 enthält:
		a) Überprüfen Sie, ob Ihr Hexapod mit einem Power-Good-Signal ausgestattet ist (Pin 59 der Buchse **Hexapod** (S. 383)). Das Signal fehlt in der Regel bei älteren Modellen (z. B. H-840.D1), die über einen Adapter von MDR68 auf HD D-Sub 78 angeschlossen sind. Diese Information finden Sie zum Beispiel im Kapitel "Pinbelegung" im Benutzerhandbuch des Hexapods. Zusätzlich können Sie den Befehl `DIA?` (S. 165) senden.
		b) Fragen Sie den Aktivierungszustand der Prüfung des Power-Good-Signals ab, indem Sie folgenden Befehl senden:
		`SPA? 1 0x19004000`
		Antwort ist 0: Power-Good-Signal wird nicht geprüft
		Antwort ist 1: Power-Good-Signal wird geprüft
		3. Wenn Ihr Hexapod nicht mit einem Power-Good-Signal ausgestattet ist, aber die Prüfung des Power-Good-Signals im C-887 aktiviert ist, senden Sie folgenden Befehl, um die Prüfung zu deaktivieren:
		`SPA 1 0x19004000 0`
		Zur Prüfung des Power-Good-Signals siehe auch "Automatisches Ausschalten des Servomodus / Stoppen der Bewegung" (S. 91).

Der Hexapod bewegt sich nicht.	Falsche Konfiguration des C-887: ID-Chip des Hexapods nicht ausgelesen	➢ Schließen Sie den Hexapod nur an den ausgeschalteten Controller an. ➢ Wenn das Booten der Firmware abgeschlossen ist, senden Sie den Befehl `CST?` (S. 164), um zu prüfen, ob die installierte Konfiguration durch einen Neustart des Controllers aktiviert werden muss. Ein Neustart ist erforderlich, wenn die Antwort für die Achsen X, Y, Z, U, V und W "NOSTAGE" lautet. Der Controller kann mit dem Befehl `RBT` (S. 253) neu gestartet werden. ➢ Senden Sie den Befehl `ERR?` (S. 176). Wenn die Antwort auf `ERR?` den Fehlercode 233 oder 211 enthält, fehlt im Controller die Konfigurationsdatei für den Hexapod. Kontaktieren Sie unseren Kundendienst (S. 367), um eine geeignete Konfigurationsdatei zu erhalten. Für die Installation der neuen Konfigurationsdatei siehe "Firmware und Konfigurationsdateien aktualisieren" (S. 340).

Störung	Mögliche Ursachen	Behebung
		➢ Senden Sie den Befehl `VER?` (S. 278), um die auf dem ID-Chip gespeicherten Angaben für Hexapod-Typ, Seriennummer und Herstelldatum zu prüfen. Beispiel für die Antwort: `IDChip: H-811.F-2 SN123456789 20/1/2016`
Der Hexapod bewegt sich nicht.	Falsche Konfiguration des C-887: Hexapod hat keinen ID-Chip	1. Wenn das Booten der Firmware abgeschlossen ist, senden Sie den Befehl `CST?` (S. 164), um zu prüfen, auf welchen Hexapod-Typ der Controller abgestimmt ist. 2. Wenn die Antwort auf `CST?` nicht zum angeschlossenen Hexapod passt, laden Sie manuell die passende Konfiguration, indem Sie folgenden Befehl senden: `DBG? choosehexapod` *Type* wobei *Type* die genaue Produktbezeichnung des angeschlossenen Hexapods ist. Der Controller antwortet mit "1" (Laden der Konfiguration war erfolgreich) oder "0" (Laden war nicht erfolgreich). Abhängig von der Antwort des Controllers sind weitere Schritte erforderlich. Wenn der Controller "1" zurückgemeldet hat: 1. Senden Sie den Befehl `RBT` (S. 253), um die neue Konfiguration durch einen Neustart des Controllers zu aktivieren. 2. Wenn das Booten der Firmware abgeschlossen ist, senden Sie den Befehl `CST?`, um zu prüfen, ob der Controller nun auf den korrekten Hexapod-Typ abgestimmt ist. 3. Wenn die Antwort auf `CST?` den korrekten Hexapod-Typ anzeigt, können Sie mit der Inbetriebnahme Ihres Systems fortfahren. Andernfalls kontaktieren Sie unseren Kundendienst (S. 367). Wenn der Controller "0" zurückgemeldet hat: 1. Senden Sie den Befehl `ERR?` (S. 176). Wenn die Antwort auf `ERR?` den Fehlercode 233 oder 211 enthält, fehlt im Controller die Konfigurationsdatei für den Hexapod. 2. Kontaktieren Sie unseren Kundendienst (S. 367), um eine geeignete Konfigurationsdatei zu erhalten. Für die Installation der neuen Konfigurationsdatei siehe "Firmware und Konfigurationsdateien aktualisieren" (S. 340). 3. Wiederholen Sie das Laden der Konfiguration (siehe oben).

Störung	Mögliche Ursachen	Behebung
Der Hexapod bewegt sich nicht.	Falscher Befehl oder falsche Syntax	➢ Senden Sie den Befehl `ERR?` (S. 176) und prüfen Sie den zurückgemeldeten Fehlercode.
Der Hexapod bewegt sich nicht.	Kommandierte Position außerhalb der aktuellen Stellwegsgrenzen	Die Antwort auf TRA? zeigt rundungsbedingt möglicherweise eine Position an, die nicht erreicht werden kann. Das Kommandieren einer solchen Position schlägt fehl und erzeugt den Fehlercode 7 ("Position out of limits"). Sie können deshalb die Antwort auf TRA? mit einem Faktor so begrenzen, dass nur Positionen angezeigt werden, die auch tatsächlich kommandiert werden können: ➢ Setzen Sie mit SPA den Parameter ***Reduction Factor for TRA? Response*** (0x19006000) auf einen geeigneten Wert zwischen 0 und 1.
Der Hexapod bewegt sich nicht.	Die Bewegungs-plattform befindet sich an einer Position außerhalb der Stellwegs-grenzen. Das Kommandieren einer zulässigen Zielposition erzeugt den Fehlercode 7 („Position out of limits“).	1. Senden Sie den Befehl `ERR?` (S. 176) und prüfen Sie den zurückgemeldeten Fehlercode. 2. Senden Sie den Befehl `POS?`, um die aktuelle Position der Bewegungsplattform zu prüfen. Wenn der Fehlercode 7 zurückgemeldet wird und die aktuelle Position für mindestens eine Achse außerhalb der Stellwegsgrenzen liegt, sind abhängig vom Sensortyp des Hexapods (inkrementell oder absolut messend) die folgenden Schritte erforderlich. **Wenn der Hexapod mit inkrementellen Sensoren ausgestattet ist:** 1. Überprüfen Sie Ihr System und stellen Sie sicher, dass alle Achsen gefahrlos bewegt werden können. 2. Starten Sie eine Referenzierungsfahrt für den Hexapod. **Wenn der Hexapod mit absolut messenden Sensoren ausgestattet ist:** 1. Überprüfen Sie Ihr System und stellen Sie sicher, dass alle Achsen gefahrlos bewegt werden können. 2. Deaktivieren Sie über den Parameter ***Check Calculated Trajectory Before Motion*** temporär die automatische Prüfung des Dynamikprofils (S. 37). **Hinweis:** Mit dem Deaktivieren der automatischen Prüfung des Dynamikprofils schalten Sie eine Sicherheitsfunktion des C-887 temporär aus. Wenn die automatische Prüfung deaktiviert ist, tragen Sie die Verantwortung dafür, die Bewegungsplattform in die richtige Richtung zu kommandieren und Kollisionen zu vermeiden. Senden Sie `SPA 1 0x19006002 0`, um die automatische Prüfung zu deaktivieren.

Störung	Mögliche Ursachen	Behebung
		3. Schalten Sie mit dem Befehl `SVO` (S. 268) den Servomodus für alle Achsen ein. 4. Kommandieren Sie die Bewegungsplattform an eine Position, die im zulässigen Arbeitsraum liegt und ca. 1 bis 5 µm von den Stellwegsgrenzen entfernt ist. Sie können dazu einen der folgenden Befehle verwenden: `MOV`, `MVR`, `MRW`, `MRT`. Nach dem erfolgreichen Anfahren der kommandierten Position wird die Prüfung des Dynamikprofils automatisch wieder aktiviert. Anmerkungen: Wenn das Anfahren der kommandierten Position fehlschlägt, bleibt die Prüfung des Dynamikprofils **deaktiviert.** Die Prüfung des Dynamikprofils wird auch durch die Befehle `HLT`, `FRF`, `STE`, `IMP`, `VMO?` wieder aktiviert. 5. Überprüfen Sie Ihr System erneut: a) Senden Sie den Befehl `ERR?` und prüfen Sie den zurückgemeldeten Fehlercode. b) Senden Sie den Befehl `POS?`, um die aktuelle Position der Bewegungsplattform zu prüfen. c) Senden Sie den Befehl `SPA? 1 0x19006002`, um den Aktivierungszustand der Prüfung des Dynamikprofils zu prüfen. Wenn der Fehlercode 0 zurückgemeldet wird, die aktuelle Position innerhalb der Stellwegsgrenzen liegt und die Prüfung des Dynamikprofils aktiviert ist, ist Ihr System wieder betriebsbereit. Wenn eine dieser Voraussetzungen nicht erfüllt ist: 1. Wiederholen Sie die oben angegebenen Schritte. 2. Wenn die Prozedur dauerhaft erfolglos bleibt, kontaktieren Sie unseren Kundendienst (S. 367).
Der Hexapod bewegt sich nicht.	Servomodus wurde automatisch ausgeschaltet	Bewegungen der Achsen sind nur bei eingeschaltetem Servomodus möglich. Das Ausschalten des Servomodus stoppt die Bewegung. 1. Senden Sie den Befehl `SVO?` (S. 269), um den Aktivierungszustand des Servomodus zu prüfen. 2. Senden Sie den Befehl `ERR?` (S. 176) und prüfen Sie den zurückgemeldeten Fehlercode. Details zu möglichen Fehlercodes und deren Ursachen siehe "Schutzfunktionen des C-887" (S. 90).

Störung	Mögliche Ursachen	Behebung
Der Hexapod bewegt sich nicht.	Funktionsgenera-torausgabe stoppt abrupt	Während der Funktionsgeneratorausgabe prüft der C-887 ständig, ob die Bewegung noch möglich ist. In folgenden Fällen stoppt der C-887 die Bewegung abrupt und setzt einen Fehlercode: ▪ Die auszugebenden Zielpositionen können nicht erreicht werden. ▪ Die erforderliche Geschwindigkeit kann nicht erreicht werden. ▪ Die Bewegung würde eine Kollision verursachen. ➢ Definieren Sie geeignete Kurvenformen, Details siehe "Kurvenform in Kurventabelle erstellen" (S. 106).
Der Hexapod bewegt sich nicht.	▪ Fremdkörper ist in die Antriebs-spindel geraten ▪ Motor defekt ▪ Sensor defekt ▪ Gelenk gebrochen oder blockiert ▪ Last zu groß	Die Referenzierungsfahrt wird nicht erfolgreich ausgeführt, oder der Servomodus wird während einer Bewegung automatisch ausgeschaltet (siehe "Automatisches Ausschalten des Servomodus / Stoppen der Bewegung" (S. 91)). Wenn möglich: ➢ Führen Sie einen Beintest durch (S. 350) und übermitteln Sie die Testergebnisse an unseren Kundendienst (S. 367). ➢ Führen Sie entweder eine Bewegung in der Z-Achse durch oder die Bewegung, bei der der Fehler auftritt, und zeichnen Sie dabei mit dem Datenrekorder für die Hexapod-Beine die aktuelle Position (RecOption 2), die Zielposition (RecOption 1) und den Motorstellwert (RecOption 73) auf. Übermitteln Sie die Testergebnisse an unseren Kundendienst (S. 367).
Verringerte Wiederhol-genauigkeit des Hexapods	▪ Verspannte Grundplatte ▪ Verspannte Bewegungs-plattform	➢ Montieren Sie den Hexapod auf ebener Grundfläche (siehe Benutzerhandbuch des Hexapods). ➢ Befestigen Sie auf der Bewegungsplattform nur Lasten mit ebener Grundfläche.
Verringerte Wiederhol-genauigkeit des Hexapods	Mangelschmierung aufgrund kleiner Bewegungen über einen langen Zeitraum	➢ Führen Sie in regelmäßigen Abständen eine Wartungsfahrt durch (S. 349).
Verringerte Wiederhol-genauigkeit des Hexapods	Äußere Störein-flüsse	➢ Stellen Sie sicher, dass keine Schwingungen auf das System übertragen werden. ➢ Stellen Sie sicher, dass die Bewegung der Bewegungsplattform nicht durch Kräfte beeinträchtigt wird, die zum Beispiel aus mitgeschleppten Kabeln resultieren.

Störung	Mögliche Ursachen	Behebung
		➢ Stellen Sie sicher, dass sich das System in einem thermischen Gleichgewicht befindet.
Schlechte Ablaufgenauigkeit	▪ Ungeeignete Parameter-einstellungen für Regelung ▪ Geschwindigkeit, Beschleunigung zu hoch oder zu niedrig ▪ Ungeeignetes Dynamikprofil ▪ Verändertes Systemverhalten durch zunehmende Leichtgängigkeit	➢ Halten Sie bei der Vorgabe von Zielposition, Geschwindigkeit und Beschleunigung die Spezifikationen des Hexapods ein. Häufig hilft es, eine möglichst niedrige Geschwindigkeit mit einer möglichst hohen Beschleunigung zu kombinieren. ➢ Stellen Sie die Parameter für die Regelung (z. B. P-, I-, D-Term) passend ein. Typischerweise erfolgt die Optimierung empirisch, d. h., Sie beobachten das Verhalten des Hexapods bei verschiedenen Werten. ➢ Wenn Sie das Dynamikprofil durch aufeinander folgende MOV-Befehle oder über die EtherCAT-Schnittstelle vorgeben, beachten Sie die Hinweise in "Zyklische Übertragung von Zielpositionen" (S. 39). ➢ Wenn Sie den Funktionsgenerator verwenden: Definieren Sie geeignete Kurvenformen, Details siehe "Kurvenform in Kurventabelle erstellen" (S. 106).

Störung	Mögliche Ursachen	Behebung
Kommunikation mit dem C-887 funktioniert nicht	Falsches Kommunikationskabel wird verwendet oder es ist defekt	➢ Prüfen Sie das Kabel. – Verwenden Sie für die RS-232-Verbindung ein Nullmodemkabel. – Verwenden Sie für die TCP/IP-Verbindung über einen Hub oder einen Router (mit DHCP-Server) das Straight-Through-Netzwerkkabel. – Verwenden Sie für die direkte Verbindung mit der Ethernet-Anschlussbuchse des PC das Crossover-Netzwerkkabel. ➢ Prüfen Sie gegebenenfalls, ob das Kabel an einem fehlerfreien System funktioniert.
	Kommunikationsschnittstelle ist nicht richtig konfiguriert	**Bei Verwendung der RS-232 Schnittstelle:** ➢ Prüfen Sie die Port- Einstellungen, die Baudrate und die Handshake-Einstellung des PC. **Bei Verwendung der TCP/IP-Verbindung:** ➢ Schließen Sie den Controller an das Netzwerk an, **bevor** Sie ihn einschalten. Andernfalls müssen Sie den Controller aus- und erneut einschalten. ➢ Prüfen Sie die Netzwerkeinstellungen (S. 74). ➢ Prüfen Sie in den Einstellungen der Firewall, dass der UDP-Broadcast erlaubt ist. Wenden Sie sich

Störung	Mögliche Ursachen	Behebung
		gegebenenfalls an Ihren Netzwerkadministrator. Geben Sie alternativ die IP-Adresse des Controllers manuell ein. ➢ Stellen Sie sicher, dass das Netzwerk nicht für unbekannte Geräte gesperrt ist. ➢ Stellen Sie sicher, dass der Netzwerkverkehr zum C-887 nicht durch eine Firewall blockiert wird. ➢ Stellen Sie sicher, dass nicht mehrere PC-Softwareanwendungen gleichzeitig auf den C-887 zugreifen können. ➢ Stellen Sie sicher, dass Sie beim Herstellen der Kommunikation den richtigen C-887 ausgewählt haben. ➢ Wenn Sie die Probleme nicht beheben können, wenden Sie sich gegebenenfalls an Ihren Netzwerkadministrator.
	Startvorgang des C-887 ist noch nicht beendet	1. Warten Sie nach dem Einschalten oder nach einem Neustart des C-887 circa 40 Sekunden (zweiter Signalton wurde ausgegeben und die LEDs **PWR** und **STA** leuchten). 2. Versuchen Sie erneut, die Kommunikation aufzubauen oder Befehle zu senden. Wenn die IP-Adressen der Netzwerkteilnehmer mit AutoIP konfiguriert werden, dauert es nach dem Ende des Startvorgangs des C-887 (S. 74) bis zu 2 Minuten, bis die Kommunikation über TCP/IP möglich ist.
	Ein anderes Programm greift auf die Schnitt-stelle zu	➢ Schließen Sie das andere Programm.
	Probleme mit spezieller PC-Software	➢ Prüfen Sie, ob das System mit einer anderen PC-Software, wie z. B. einem Terminalprogramm oder einer Entwicklungsumgebung, funktioniert. Sie können die Kommunikation testen, indem Sie ein Terminalprogramm (z. B. PITerminal) starten und `*IDN?` oder `HLP?` eingeben. ➢ Achten Sie darauf, dass Sie Befehle mit einem LF (line feed) abschließen. Ein Befehl wird erst ausgeführt, wenn der LF empfangen wurde.

Störung	Mögliche Ursachen	Behebung
Kundensoftware läuft nicht mit den Treibern von PI	Falsche Kombination der Treiberroutinen/VIs	➢ Prüfen Sie, ob das System mit einem Terminalprogramm läuft. Wenn ja: ➢ Lesen Sie die Angaben im Handbuch der zugehörigen PC-Software und vergleichen Sie den Beispielcode auf dem Datenträger der PI Software Suite mit Ihrem Programmcode.

Wenn die Störung Ihres Systems nicht in der Tabelle angeführt ist oder wenn sie nicht wie beschrieben behoben werden kann, kontaktieren Sie unseren Kundendienst (S. 367).

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. 9) bereit.

13 Technische Daten

Änderungen vorbehalten. Die aktuellen Produktspezifikationen finden Sie auf der Seite des Produkts unter www.pi.de.

In diesem Kapitel

Spezifikationen 369
Abmessungen 378
Spezifikationen Kabel 379
Pinbelegung 380
Statusregister 386

13.1 Spezifikationen

13.1.1 Datentabelle

Grundlegendes	C-887.52	C-887.521	C-887.522	C-887.523
Achsen/Kanäle	6	6	6	6
Zusatzachsen	2 Einzelachsen	2 Einzelachsen	2 Einzelachsen	2 Einzelachsen
Prozessor	Intel Atom Dual Core (1,8 GHz)	Intel Atom Dual Core (1,8 GHz)	Intel Atom Dual Core (1,8 GHz)	Intel Atom Dual Core (1,8 GHz)
Anwendungs-bezogene Funktionen	Controller-Makros GCS | Controller-Makros PIPython | Datenrekorder | Startup-Makro	Controller-Makros GCS | Controller-Makros PIPython | Datenrekorder | Fast Alignment | Startup-Makro	Controller-Makros GCS | Controller-Makros PIPython | Datenrekorder | Startup-Makro	Controller-Makros GCS | Controller-Makros PIPython | Datenrekorder | Fast Alignment | Startup-Makro
Schutzfunktionen	Ausschalten des Servomodus im Fehlerfall	Ausschalten des Servomodus im Fehlerfall	Achsabschaltung über externen Schalter | Ausschalten des Servomodus im Fehlerfall	Achsabschaltung über externen Schalter | Ausschalten des Servomodus im Fehlerfall
Konfigurations-Management	Auslesen des ID-Chips | manuelle Parametereingabe	Auslesen des ID-Chips | manuelle Parametereingabe	Auslesen des ID-Chips | manuelle Parametereingabe	Auslesen des ID-Chips | manuelle Parametereingabe
Unterstützter ID-Chip	ID-Chip 2.0	ID-Chip 2.0	ID-Chip 2.0	ID-Chip 2.0

Bewegung und Regler	C-887.52	C-887.521	C-887.522	C-887.523
Unterstütztes Sensorsignal	A/B-Quadratur, RS-422 | BiSS-C	A/B-Quadratur, RS-422 | BiSS-C	A/B-Quadratur, RS-422 | BiSS-C	A/B-Quadratur, RS-422 | BiSS-C
Geregelte Größen	Position	Position	Position	Position
Maximale Regelfrequenz (Servozyklus)	10000 Hz	10000 Hz	10000 Hz	10000 Hz
Bewegungstypen	Punkt-zu-Punkt-Bewegung mit Profilgenerator | Zyklisch synchrone Zielwertvorgabe für Position | Area Scan Routinen | Gradient Search Routinen | Funktionsgenerator	Punkt-zu-Punkt-Bewegung mit Profilgenerator | Zyklisch synchrone Zielwertvorgabe für Position | Area Scan Routinen | Gradient Search Routinen | Funktionsgenerator	Punkt-zu-Punkt-Bewegung mit Profilgenerator | Zyklisch synchrone Zielwertvorgabe für Position | Area Scan Routinen | Gradient Search Routinen | Funktionsgenerator	Punkt-zu-Punkt-Bewegung mit Profilgenerator | Zyklisch synchrone Zielwertvorgabe für Position | Area Scan Routinen | Gradient Search Routinen | Funktionsgenerator
Koordinierung der Bewegung	Koordinierte Mehrachsbewegung | Benutzerdefinierte Koordinatensysteme | Work- und Tool-Koordinatensysteme	Koordinierte Mehrachsbewegung | Benutzerdefinierte Koordinatensysteme | Work- und Tool-Koordinatensysteme	Koordinierte Mehrachsbewegung | Benutzerdefinierte Koordinatensysteme | Work- und Tool-Koordinatensysteme	Koordinierte Mehrachsbewegung | Benutzerdefinierte Koordinatensysteme | Work- und Tool-Koordinatensysteme
Referenzschaltereingang	TTL	TTL	TTL	TTL
Endschaltereingang	TTL	TTL	TTL	TTL

Schnittstellen und Bedienung	C-887.52	C-887.521	C-887.522	C-887.523
Kommunikationsschnittstellen	RS-232 | TCP/IP | USB (nur für manuelle Bedieneinheiten)	RS-232 | TCP/IP | USB (nur für manuelle Bedieneinheiten)	RS-232 | TCP/IP | USB (nur für manuelle Bedieneinheiten)	RS-232 | TCP/IP | USB (nur für manuelle Bedieneinheiten)
An/Aus-Schalter	Hardware-Schalter An/Aus	Hardware-Schalter An/Aus	Hardware-Schalter An/Aus	Hardware-Schalter An/Aus
Display und Anzeigen	Status-LED | Error-LED | Power-LED | Macro-LED	Status-LED | Error-LED | Power-LED | Macro-LED	Status-LED | Error-LED | Power-LED | Macro-LED	Status-LED | Error-LED | Power-LED | Macro-LED
Manuelle Bedienhilfe(n)	Manuelle Bedieneinheit mit USB-Anschluss	Manuelle Bedieneinheit mit USB-Anschluss	Manuelle Bedieneinheit mit USB-Anschluss	Manuelle Bedieneinheit mit USB-Anschluss
Befehlssatz	GCS 2.0	GCS 2.0	GCS 2.0	GCS 2.0
Bedienersoftware	PIMikroMove	PIMikroMove	PIMikroMove	PIMikroMove

Schnittstellen und Bedienung	C-887.52	C-887.521	C-887.522	C-887.523
Software-APIs	C, C++, C# | MATLAB | NI LabView | Python	C, C++, C# | MATLAB | NI LabView | Python	C, C++, C# | MATLAB | NI LabView | Python	C, C++, C# | MATLAB | NI LabView | Python
Analoge Eingänge	4	6	4	6
Analoges Eingangssignal	4 x -10 bis 10 V, 12 Bit	2 x -5 bis 5 V, 16 Bit, 5 kHz Bandbreite 4 x -10 bis 10 V, 12 Bit	4 x -10 bis 10 V, 12 Bit	2 x -5 bis 5 V, 16 Bit, 5 kHz Bandbreite 4 x -10 bis 10 V, 12 Bit
Digitale Eingänge	4	4	4	4
Digitales Eingangssignal	TTL	TTL	TTL	TTL
Digitale Ausgänge	4	4	4	4
Digitales Ausgangssignal	TTL	TTL	TTL	TTL

Elektrische Eigenschaften	C-887.52	C-887.521	C-887.522	C-887.523
Ausgangsspannung	24 V	24 V	24 V	24 V
Spitzenausgangs-strom	6000 mA	6000 mA	6000 mA	6000 mA

Anschlüsse und Umgebung	C-887.52	C-887.521	C-887.522	C-887.523
Hexapodanschluss Stromversorgung	M12 4-polig (w)	M12 4-polig (w)	M12 4-polig (w)	M12 4-polig (w)
Hexapodanschluss Datenübertragung	HD D-Sub 78 (w)	HD D-Sub 78 (w)	HD D-Sub 78 (w)	HD D-Sub 78 (w)
Anschluss Zusatzachsen	D-Sub 15 (w)	D-Sub 15 (w)	D-Sub 15 (w)	D-Sub 15 (w)
Anschluss Motion-Stop-Button	-	-	M12 8-polig (w)	M12 8-polig (w)
Anschluss analoger Eingang	HD D-Sub 26 (w)	BNC | HD D-Sub 26 (w)	HD D-Sub 26 (w)	BNC | HD D-Sub 26 (w)
Anschluss digitaler Eingang	HD D-Sub 26 (w)	HD D-Sub 26 (w)	HD D-Sub 26 (w)	HD D-Sub 26 (w)
Anschluss digitaler Ausgang	HD D-Sub 26 (w)	HD D-Sub 26 (w)	HD D-Sub 26 (w)	HD D-Sub 26 (w)
Anschluss TCP/IP	RJ45 Buchse, 8P8C	RJ45 Buchse, 8P8C	RJ45 Buchse, 8P8C	RJ45 Buchse, 8P8C
Anschluss RS-232	D-Sub 9 (m)	D-Sub 9 (m)	D-Sub 9 (m)	D-Sub 9 (m)

Anschlüsse und Umgebung	C-887.52	C-887.521	C-887.522	C-887.523
Anschluss Versorgungsspannung	M12 4-polig (m)	M12 4-polig (m)	M12 4-polig (m)	M12 4-polig (m)
Betriebsspannung	24 V	24 V	24 V	24 V
Maximale Stromaufnahme	8 A	8 A	8 A	8 A
Netzteil	Im Lieferumfang | Netzteil 24 V DC	Im Lieferumfang | Netzteil 24 V DC	Im Lieferumfang | Netzteil 24 V DC	Im Lieferumfang | Netzteil 24 V DC
Betriebstemperaturbereich	5 bis 40 °C	5 bis 40 °C	5 bis 40 °C	5 bis 40 °C
Gesamtmasse	2800 g	2800 g	2800 g	2800 g

Grundlegendes	C-887.53	C-887.531	C-887.532	C-887.533
Achsen/Kanäle	6	6	6	6
Zusatzachsen	2 Einzelachsen	2 Einzelachsen	2 Einzelachsen	2 Einzelachsen
Prozessor	Intel Atom Dual Core (1,8 GHz)	Intel Atom Dual Core (1,8 GHz)	Intel Atom Dual Core (1,8 GHz)	Intel Atom Dual Core (1,8 GHz)
Anwendungsbezogene Funktionen	Controller-Makros GCS | Controller-Makros PIPython | Datenrekorder | Startup-Makro	Controller-Makros GCS | Controller-Makros PIPython | Datenrekorder | Fast Alignment | Startup-Makro	Controller-Makros GCS | Controller-Makros PIPython | Datenrekorder | Startup-Makro	Controller-Makros GCS | Controller-Makros PIPython | Datenrekorder | Fast Alignment | Startup-Makro
Schutzfunktionen	Ausschalten des Servomodus im Fehlerfall	Ausschalten des Servomodus im Fehlerfall	Achsabschaltung über externen Schalter | Ausschalten des Servomodus im Fehlerfall	Achsabschaltung über externen Schalter | Ausschalten des Servomodus im Fehlerfall
Konfigurations-Management	Auslesen des ID-Chips | manuelle Parametereingabe	Auslesen des ID-Chips | manuelle Parametereingabe	Auslesen des ID-Chips | manuelle Parametereingabe	Auslesen des ID-Chips | manuelle Parametereingabe
Unterstützter ID-Chip	ID-Chip 2.0	ID-Chip 2.0	ID-Chip 2.0	ID-Chip 2.0

Bewegung und Regler	C-887.53	C-887.531	C-887.532	C-887.533
Unterstütztes Sensorsignal	A/B-Quadratur, RS-422 | BiSS-C	A/B-Quadratur, RS-422 | BiSS-C	A/B-Quadratur, RS-422 | BiSS-C	A/B-Quadratur, RS-422 | BiSS-C
Geregelte Größen	Position	Position	Position	Position
Maximale Regelfrequenz (Servozyklus)	10000 Hz	10000 Hz	10000 Hz	10000 Hz

Bewegung und Regler	C-887.53	C-887.531	C-887.532	C-887.533
Bewegungstypen	Punkt-zu-Punkt-Bewegung mit Profilgenerator | Zyklisch synchrone Zielwertvorgabe für Position | Area Scan Routinen | Gradient Search Routinen | Funktions-generator	Punkt-zu-Punkt-Bewegung mit Profilgenerator | Zyklisch synchrone Zielwertvorgabe für Position | Area Scan Routinen | Gradient Search Routinen | Funktions-generator	Punkt-zu-Punkt-Bewegung mit Profilgenerator | Zyklisch synchrone Zielwertvorgabe für Position | Area Scan Routinen | Gradient Search Routinen | Funktions-generator	Punkt-zu-Punkt-Bewegung mit Profilgenerator | Zyklisch synchrone Zielwertvorgabe für Position | Area Scan Routinen | Gradient Search Routinen | Funktions-generator
Koordinierung der Bewegung	Koordinierte Mehr-achsbewegung | Benutzerdefinierte Koordinaten-systeme | Work- und Tool-Koordina-tensysteme	Koordinierte Mehr-achsbewegung | Benutzerdefinierte Koordinaten-systeme | Work- und Tool-Koordina-tensysteme	Koordinierte Mehr-achsbewegung | Benutzerdefinierte Koordinaten-systeme | Work- und Tool-Koordina-tensysteme	Koordinierte Mehr-achsbewegung | Benutzerdefinierte Koordinaten-systeme | Work- und Tool-Koordina-tensysteme
Referenzschalter-eingang	TTL	TTL	TTL	TTL
Endschaltereingang	TTL	TTL	TTL	TTL

Schnittstellen und Bedienung	C-887.53	C-887.531	C-887.532	C-887.533
Kommunikations-schnittstellen	EtherCAT-Slave | RS-232 | TCP/IP | USB (nur für manuelle Bedieneinheiten)	EtherCAT-Slave | RS-232 | TCP/IP | USB (nur für manuelle Bedieneinheiten)	EtherCAT-Slave | RS-232 | TCP/IP | USB (nur für manuelle Bedieneinheiten)	EtherCAT-Slave | RS-232 | TCP/IP | USB (nur für manuelle Bedieneinheiten)
An/Aus-Schalter	Hardware-Schalter An/Aus	Hardware-Schalter An/Aus	Hardware-Schalter An/Aus	Hardware-Schalter An/Aus
Display und Anzeigen	Status-LED | Error-LED | Power-LED | Macro-LED	Status-LED | Error-LED | Power-LED | Macro-LED	Status-LED | Error-LED | Power-LED | Macro-LED	Status-LED | Error-LED | Power-LED | Macro-LED
Manuelle Bedienhilfe(n)	Manuelle Bedieneinheit mit USB-Anschluss	Manuelle Bedieneinheit mit USB-Anschluss	Manuelle Bedieneinheit mit USB-Anschluss	Manuelle Bedieneinheit mit USB-Anschluss
Befehlssatz	GCS 2.0	GCS 2.0	GCS 2.0	GCS 2.0
Bedienersoftware	PIMikroMove	PIMikroMove	PIMikroMove	PIMikroMove
Software-APIs	C, C++, C# | MATLAB | NI LabView | Python	C, C++, C# | MATLAB | NI LabView | Python	C, C++, C# | MATLAB | NI LabView | Python	C, C++, C# | MATLAB | NI LabView | Python
Analoge Eingänge	4	6	4	6

Schnittstellen und Bedienung	C-887.53	C-887.531	C-887.532	C-887.533
Analoges Eingangssignal	4 x -10 bis 10 V, 12 Bit	2 x -5 bis 5 V, 16 Bit, 5 kHz Bandbreite 4 x -10 bis 10 V, 12 Bit	4 x -10 bis 10 V, 12 Bit	2 x -5 bis 5 V, 16 Bit, 5 kHz Bandbreite 4 x -10 bis 10 V, 12 Bit
Digitale Eingänge	4	4	4	4
Digitales Eingangssignal	TTL	TTL	TTL	TTL
Digitale Ausgänge	4	4	4	4
Digitales Ausgangssignal	TTL	TTL	TTL	TTL
Industrial Ethernet-Protokoll	EtherCAT	EtherCAT	EtherCAT	EtherCAT
EtherCAT-Geräteklasse	EtherCAT-Slave	EtherCAT-Slave	EtherCAT-Slave	EtherCAT-Slave
EtherCAT-Kommunikationsprofil	CAN application protocol over EtherCAT (CoE)	CAN application protocol over EtherCAT (CoE)	CAN application protocol over EtherCAT (CoE)	CAN application protocol over EtherCAT (CoE)
Antriebsprofil implementiert für EtherCAT	CiA402 Drive Profile (IEC 61800-7-201)	CiA402 Drive Profile (IEC 61800-7-201)	CiA402 Drive Profile (IEC 61800-7-201)	CiA402 Drive Profile (IEC 61800-7-201)
Unterstützte Betriebsmodi gemäß CiA402	Referenzierungsfahrt (homing mode) | Positioniermodus mit zyklischer Positionsvorgabe durch die SPS (cyclic synchronous position mode) | Sicherer Grundzustand zum Aktivieren von Koordinatensystemen (No mode changes / no mode selected)	Referenzierungsfahrt (homing mode) | Positioniermodus mit zyklischer Positionsvorgabe durch die SPS (cyclic synchronous position mode) | Sicherer Grundzustand zum Aktivieren von Koordinatensystemen (No mode changes / no mode selected)	Referenzierungsfahrt (homing mode) | Positioniermodus mit zyklischer Positionsvorgabe durch die SPS (cyclic synchronous position mode) | Sicherer Grundzustand zum Aktivieren von Koordinatensystemen (No mode changes / no mode selected)	Referenzierungsfahrt (homing mode) | Positioniermodus mit zyklischer Positionsvorgabe durch die SPS (cyclic synchronous position mode) | Sicherer Grundzustand zum Aktivieren von Koordinatensystemen (No mode changes / no mode selected)
EtherCAT-Synchronisierungsmodi	Distributed Clocks (DC) | Synchron mit SYNC0 Event	Distributed Clocks (DC) | Synchron mit SYNC0 Event	Distributed Clocks (DC) | Synchron mit SYNC0 Event	Distributed Clocks (DC) | Synchron mit SYNC0 Event
EtherCAT-Zykluszeit	≥1 ms	≥1 ms	≥1 ms	≥1 ms

Elektrische Eigenschaften	C-887.53	C-887.531	C-887.532	C-887.533
Ausgangsspannung	24 V	24 V	24 V	24 V
Spitzenausgangs-strom	6000 mA	6000 mA	6000 mA	6000 mA

Anschlüsse und Umgebung	C-887.53	C-887.531	C-887.532	C-887.533
Hexapodanschluss Stromversorgung	M12 4-polig (w)	M12 4-polig (w)	M12 4-polig (w)	M12 4-polig (w)
Hexapodanschluss Datenübertragung	HD D-Sub 78 (w)	HD D-Sub 78 (w)	HD D-Sub 78 (w)	HD D-Sub 78 (w)
Anschluss Zusatzachsen	D-Sub 15 (w)	D-Sub 15 (w)	D-Sub 15 (w)	D-Sub 15 (w)
Anschluss Motion-Stop-Button	-	-	M12 8-polig (w)	M12 8-polig (w)
Anschluss analoger Eingang	HD D-Sub 26 (w)	BNC | HD D-Sub 26 (w)	HD D-Sub 26 (w)	BNC | HD D-Sub 26 (w)
Anschluss digitaler Eingang	HD D-Sub 26 (w)	HD D-Sub 26 (w)	HD D-Sub 26 (w)	HD D-Sub 26 (w)
Anschluss digitaler Ausgang	HD D-Sub 26 (w)	HD D-Sub 26 (w)	HD D-Sub 26 (w)	HD D-Sub 26 (w)
Anschluss TCP/IP	RJ45 Buchse, 8P8C	RJ45 Buchse, 8P8C	RJ45 Buchse, 8P8C	RJ45 Buchse, 8P8C
Anschluss RS-232	D-Sub 9 (m)	D-Sub 9 (m)	D-Sub 9 (m)	D-Sub 9 (m)
Anschluss EtherCAT	RJ45 Buchse, 8P8C	RJ45 Buchse, 8P8C	RJ45 Buchse, 8P8C	RJ45 Buchse, 8P8C
Anschluss Versor-gungsspannung	M12 4-polig (m)	M12 4-polig (m)	M12 4-polig (m)	M12 4-polig (m)
Betriebsspannung	24 V	24 V	24 V	24 V
Maximale Stromaufnahme	8 A	8 A	8 A	8 A
Netzteil	Im Lieferumfang | Netzteil 24 V DC	Im Lieferumfang | Netzteil 24 V DC	Im Lieferumfang | Netzteil 24 V DC	Im Lieferumfang | Netzteil 24 V DC
Betriebstempera-turbereich	5 bis 40 °C	5 bis 40 °C	5 bis 40 °C	5 bis 40 °C
Gesamtmasse	2800 g	2800 g	2800 g	2800 g

13.1.2 Spezifikationen der analogen Eingänge

Anschluss	**Buchse** I/O	**BNC-Buchsen Analog In 5 und Analog In 6**
Vorhandensein	alle Modelle C-887.5xx	nur Modelle C-887.521, .523, .531, .533
Eingangsspannungsbereich	-10 bis 10 V	-5 bis 5 V
Max. zulässige Überspannung	12,5 V	25 V
Auflösung ADC	12 Bit	16 Bit
Absolute Genauigkeit	10 Bit	13 Bit
Abtastrate	10 kHz	10 kHz
Bandbreite	100 Hz	5 kHz
Eingangswiderstand	15 kΩ	15 kΩ
Kanalkennungen, die in Befehlen zu verwenden sind	1 bis 4 (Kennungen entsprechen den Nummern in den Signalnamen, siehe "I/O-Anschluss" (S. 381))	**Analog In 5**: 5 **Analog In 6**: 6

13.1.3 Zykluszeiten

Achse	**X, Y, Z, U, V, W**	**A, B, Hexapod-Beine 1 bis 6**
Zykluszeit	1 ms	0,1 ms

Die Unterschiede in den Zykluszeiten resultieren aus den Antriebskonzepten:

- Achsen X, Y, Z, U, V und W der Bewegungsplattform des Hexapods: Bewegung resultiert aus den Bewegungen der Hexapod-Beine 1 bis 6
- Hexapod-Beine 1 bis 6 und Achsen A und B: geregelte Antriebe mit PWM-Verstärkern

13.1.4 Bemessungsdaten

Der C-887 ist für folgende Betriebsgrößen ausgelegt:

Eingang an:	**Maximale Betriebs-spannung**	⚠	**Maximale Betriebs-frequenz**	⚠	**Maximale Stromauf-nahme**	⚠
Einbaustecker M12 4-polig (m)	24 V		⎓		8 A	

Ausgang an:	Maximale Ausgangs-spannung	⚠	Maximale Ausgabe-frequenz	⚠	Maximaler Ausgangs-strom	⚠
Buchse M12 4-polig (w)	24 V		⎓		7 A	

13.1.5 Umgebungsbedingungen und Klassifizierungen

Folgende Umgebungsbedingungen und Klassifizierungen sind für den C-887 zu beachten:

Einsatzbereich	Nur zur Verwendung in Innenräumen
Maximale Höhe	2000 m
Luftdruck	1100 hPa bis 0,1 hPa
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
Schutzart gemäß IEC 60529	IP20

13.2 Abmessungen

Abmessungen in mm

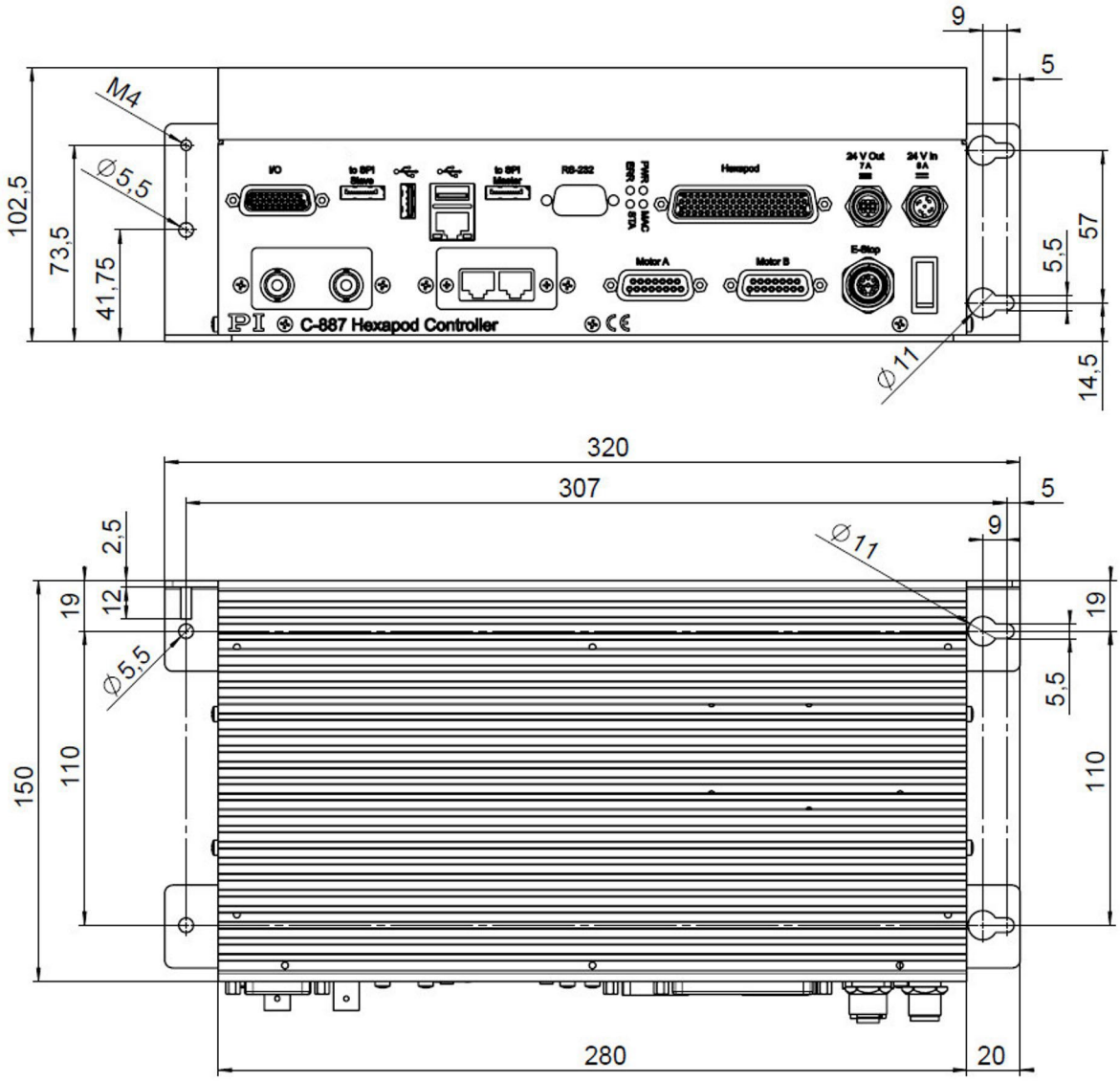


Abbildung 18: Abmessungen gültig für alle Modelle C-887.5xx

13.3 Spezifikationen Kabel

13.3.1 Datenübertragungs- und Stromversorgungskabel

Datenübertra-gungskabel	Stromversorgungskabel, einseitig abgewinkelter Stecker	Stromversorgungskabel, gerade Stecker
Alle Hexapodtypen	H-820, H-824, H-825, H-840, H-850	H-810, H-811, H-206
C-815.82D02	C-815.82P02A	C-815.82P02E
C-815.82D03	C-815.82P03A	C-815.82P03E
C-815.82D05	C-815.82P05A	C-815.82P05E
C-815.82D07	C-815.82P07A	C-815.82P07E
C-815.82D10	C-815.82P10A	C-815.82P10E
C-815.82D20	C-815.82P20A	C-815.82P20E

Die Modelle unterscheiden sich bezüglich folgender Merkmale:

1. Kabeltyp
2. Länge
3. Steckertyp (nur Stromkabel)

In der Produktnummer sind diese Merkmale durch die Stellen nach dem C-815.82 wie folgt verschlüsselt:

Stelle nach dem C-815.82	Bedeutung	Mögliche Werte
Erste Stelle	Kabeltyp	D – Datenübertragungskabel P – Stromversorgungskabel
Zweite Stelle	Länge	02 – 2 m 03 – 3 m 05 – 5 m 07 – 7,5 m 10 – 10 m 20 – 20 m
Dritte Stelle	Steckertyp (nur Stromversorgungskabel)	A – abgewinkelter Steckverbinder E – gerader Steckverbinder

Allgemein		Einheit
Kabellänge L	2 / 3 / 5 / 7,5 / 10 / 20	m
Maximale Geschwindigkeit	3	m/s
Maximale Beschleunigung	5	m/s^2
Maximale Anzahl Biegezyklen	1 Mio.	
Betriebstemperaturbereich	-10 bis +70	°C

Stromversorgungskabel, gerade Stecker		Einheit
Minimaler Biegeradius in einer Schleppkette	49	mm
Minimaler Biegeradius bei der Festinstallation	24,5	mm
Außendurchmesser	4,9	mm
Stecker	M12 m/w	

Stromversorgungskabel, abgewinkelter Stecker			Einheit
Kabellänge L	3	2 / 5 / 7,5 / 10 / 20	m
Minimaler Biegeradius in einer Schleppkette	72	94	mm
Minimaler Biegeradius bei der Festinstallation	36	57	mm
Außendurchmesser	7,2	7,5	mm
Stecker	M12 m/w		

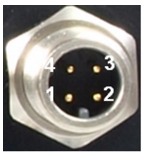
Datenübertragungskabel		Einheit
Minimaler Biegeradius in einer Schleppkette	107	mm
Minimaler Biegeradius bei der Festinstallation	81	mm
Außendurchmesser	10,7	mm
Stecker	HD D-Sub78 m/w	

Kabellängen über 20 m erfordern zusätzliche Leistungstreiber.

13.4 Pinbelegung

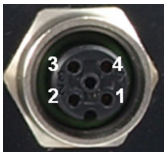
13.4.1 Anschluss zur Stromversorgung

Stromversorgung über den Einbaustecker **24 V In 8 A**, M12 4-polig (m)

Pin	Signal	
1	GND	
2	GND	
3	24 V DC	
4	24 V DC	

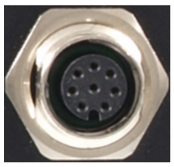
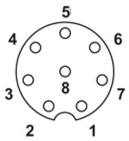
13.4.2 Versorgungsspannung für Hexapod

Ausgabe der Versorgungsspannung für die Antriebe des Hexapods über die Buchse **24 V Out 7 A**, M12 4-polig (w)

Pin	Signal	
1	GND	
2	GND	
3	24 V DC	
4	24 V DC	

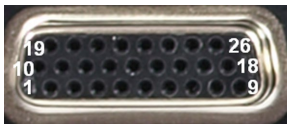
13.4.3 E-Stop

Buchse M12 8-polig (w)

Pin	Signal	
1	Break contact 1	 
2	Break contact 2	
3	reserviert	
4	reserviert	
5	Make contact 1	
6	Make contact 2	
7	reserviert	
8	reserviert	

13.4.4 IO-Anschluss

Buchse HD D-Sub 26 (w)

Pin	Pin	Pin	Signal	
	10		Analoger Eingang 1	
1			Analoger Eingang 2	
		19	Analoger Eingang 3	
	11		Analoger Eingang 4	
2			GND (analog)	
		20	GND	
	12		reserviert	
3			reserviert	
		21	reserviert	

Pin	Pin	Pin	Signal	
	13		reserviert	
4			reserviert	
		22	GND	
	14		reserviert	
5			reserviert	
		23	reserviert	
	15		reserviert	
6			Vcc (+5 V, max. 500 mA)	
		24	GND	
	16		Digitaler Eingang 4 (TTL)	
7			Digitaler Eingang 3 (TTL)	
		25	Digitaler Eingang 2 (TTL)	
	17		Digitaler Eingang 1 (TTL)	
8			Digitaler Ausgang 4 (TTL)	
		26	Digitaler Ausgang 3 (TTL)	
	18		Digitaler Ausgang 2 (TTL)	
9			Digitaler Ausgang 1 (TTL)	

Analoge Eingänge: -10 V bis 10 V, 12 Bit; 15 kΩ Eingangswiderstand

Digitale Ausgänge:

Anstiegszeit und Abfallzeit = max. 500 ns

Ausgangsstrom = max. 10 mA per Pin

Digitale Eingänge:

Eingangswiderstand = 10 kΩ

Eingangsspannung = 0 bis 5,5 V

Schmitt-Triggereingang

	min	max
V_{T+} (Schaltschwelle bei ansteigender Eingangsspannung)	1,3 V	2,2 V
V_{T-} (Schaltschwelle bei fallender Eingangsspannung)	0,6 V	1,3 V
ΔV_T (Hysterese; V_{T+} - V_{T-})	0,4 V	1,1 V

13.4.5 Hexapod

Datenübertragung zwischen Hexapod und Controller, Stromversorgung für die Sensoren des Hexapods

Buchse HD D-Sub 78 (w)

Signaltyp	Buchse
Alle Signale: TTL	

Pinbelegung

Pin	Pin	Signal
1		CH1 Sign OUT / MO-
	21	CH1 Ref IN
2		nc
	22	CH1 A+ IN / MA+
3		CH1 A- IN / MA-
	23	GND
4		CH2 Sign OUT / MO-
	24	CH2 Ref IN
5		nc
	25	CH2 A+ IN / MA+
6		CH2 A- IN / MA-
	26	GND
7		CH3 Sign OUT / MO-
	27	CH3 Ref IN
8		nc
	28	CH3 A+ IN / MA+
9		CH3 A- IN / MA-
	29	GND
10		CH4 Sign OUT / MO-
	30	CH4 Ref IN
11		nc

Pin	Pin	Signal
40		CH1 MAGN OUT / MO+
	60	CH1 LimP IN
41		CH1 LimN IN
	61	CH1 B+ IN / SL+
42		CH1 B- IN / SL-
	62	GND
43		CH2 MAGN OUT / MO+
	63	CH2 LimP IN
44		CH2 LimN IN
	64	CH2 B+ IN / SL+
45		CH2 B- IN / SL-
	65	GND
46		CH3 MAGN OUT / MO+
	66	CH3 LimP IN
47		CH3 LimN IN
	67	CH3 B+ IN /SL+
48		CH3 B- IN / SL-
	68	GND
49		CH4 MAGN OUT / MO+
	69	CH4 LimP IN
50		CH4 LimN IN

Pin	Pin	Signal
	31	CH4 A+ IN / MA+
12		CH4 A- IN / MA-
	32	GND
13		CH5 Sign OUT / MO-
	33	CH5 Ref IN
14		nc
	34	CH5 A+ IN / MA+
15		CH5 A- IN / MA-
	35	GND
16		CH6 Sign OUT / MO-
	36	CH6 Ref IN
17		nc
	37	CH6 A+ IN / MA+
18		CH6 A- IN / MA-
	38	GND
19		ID Chip
	39	GND
20		24 V Ausgang

Pin	Pin	Signal
	70	CH4 B+ IN / SL+
51		CH4 B- IN / SL-
	71	GND
52		CH5 MAGN OUT / MO+
	72	CH5 LimP IN
53		CH5 LimN IN
	73	CH5 B+ IN / SL+
54		CH5 B- IN / SL-
	74	GND
55		CH6 MAGN OUT / MO+
	75	CH6 LimP IN
56		CH6 LimN IN
	76	CH6 B+ IN / SL+
57		CH6 B- IN / SL-
	77	GND
58		Brake/Enable drive
	78	GND
59		Power Good Eingang*

* Rückgabe der aktuellen Betriebsspannung des Hexapods. Die Antriebe des Hexapods werden ordnungsgemäß mit Strom versorgt, wenn das Power-Good-Signal der erforderlichen Betriebsspannung des Hexapods entspricht (Toleranzbereich ±10 %). Das Power-Good-Signal kann mit dem Befehl `DIA?` (S. 165) geprüft werden. Weitere Informationen siehe "Schutzfunktionen des C-887" (S. 90).

Der C-887 unterstützt ab der Seriennummer 121017873 die Signale MA, MO und SL, mit denen die Datenübertragung zwischen Hexapod und Controller über das BiSS-Protokoll erfolgt.

13.4.6 Motor A, Motor B

Buchse D-Sub 15 (w)

Pin	Pin	Signal	
1		Ausgang: +5 V für Motorbremse	
	9	Reserviert	
2		Reserviert	
	10	PWM GND	
3		Ausgang: MAGN (Motor-PWM, TTL-Signal)	
	11	Ausgang: SIGN (Drehrichtung des Motors, TTL-Signal)	
4		Ausgang: +5 V, für Encoder	
	12	Eingang: negativer Endschalter	
5		Eingang: positiver Endschalter	
	13	Eingang: REFS (Referenzschalter, TTL-Signal)	
6		Reserviert	
	14	Eingang: Encoder: A (+) / ENCA	
7		Eingang: Encoder: A (-)	
	15	Eingang: Encoder: B (+) / ENCB	
8		Eingang: Encoder: B (-)	

13.4.7 RS-232

Einbaustecker D-Sub 9 (m)

Pin	Funktion	
1	Nicht verbunden	
2	RxD, Dateneingang	
3	TxD, Datenausgang	
4	Nicht verbunden	
5	DGND (Masse digital)	
6	Nicht verbunden	
7	RTS, Hardware-Handshake Ausgang	
8	CTS, Hardware-Handshake Eingang	
9	Nicht verbunden	

13.5 Statusregister

13.5.1 Statusregister für Hexapod-Beine und Achsen A und B

Für jedes der Beine 1 bis 6 des Hexapods und für die Achsen A und B hat der C-887 jeweils ein Statusregister. Sie können die Bits dieser Register mit dem Befehl SRG? (S. 262) abfragen und mit dem Datenrekorder des C-887 (S. 99) aufzeichnen, Aufzeichnungsoption 80 (Status register of axis).

Bit	15	14	13	12	11	10	9	8
Beschreibung	Zielposition erreicht ("on target")	Referenzierungsfahrt wird ausgeführt	In Bewegung	Servomodus Ein	-	-	-	Fehlerflag

Bit	7	6	5	4	3	2	1	0
Beschreibung	-	-	-	-	-	Pos. Endschalter (wenn Auswertung aktiviert*)	Referenzschalter	Neg. Endschalter (wenn Auswertung aktiviert*)

* Die Endschalterauswertung wird mit dem Parameter ***Has No Limit Switches*** (ID 0x32) aktiviert. Die Aktivierung ist nur wirksam, wenn die Mechanik tatsächlich Endschaltersignale an den C-887 liefert.

Unbelegte Bits haben den Wert 0.

Der Zustand der Bits 13 und 15 basiert auf den Kriterien für die Ermittlung des Bewegungsstatus (S. 46).

13.5.2 Systemstatusregister

Sie können die Bits des folgenden Registers mit den Befehlen STA? (S. 266) und #4 (S. 152) abfragen:

Bit:	23	22	21	20	19	18	17	16
	-	-	-	-	Referenzierungsfahrt wird ausgeführt	Referenzierungsfahrt Achse B erfolgreich	Referenzierungsfahrt Achse A erfolgreich	Referenzierungsfahrt Hexapod erfolgreich
Bit:	15	14	13	12	11	10	9	8
	Achse B in Bewegung	Achse A in Bewegung	Bein 6 in Bewegung	Bein 5 in Bewegung	Bein 4 in Bewegung	Bein 3 in Bewegung	Bein 2 in Bewegung	Bein 1 in Bewegung
Bit:	7	6	5	4	3	2	1	0
	Bewegungsfehler Achse B	Bewegungsfehler Achse A	Bewegungsfehler Bein 6	Bewegungsfehler Bein 5	Bewegungsfehler Bein 4	Bewegungsfehler Bein 3	Bewegungsfehler Bein 2	Bewegungsfehler Bein 1

Unbelegte Bits haben den Wert 0.

Der Zustand der Bits 8 bis 15 basiert auf den Kriterien für die Ermittlung des Bewegungsstatus (S. 46).

Beispiel:

Senden: STA?

Empfangen: 0x71804

Die Antwort wird im Hexadezimalformat angegeben. Sie besagt: Für Bein 3 wurde ein Bewegungsfehler gemeldet, die Beine 4 und 5 sind in Bewegung. Die Referenzierungsfahrt des Hexapods und der Achsen A und B wurde erfolgreich abgeschlossen.

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) GmbH & 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) GmbH & Co. KG

Auf der Römerstr. 1

D-76228 Karlsruhe

15 Europäische Konformitätserklärungen

Für den C-887 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