

MS173D Benutzerhandbuch

C-863 Mercury

DC-Motor Controller

Version: 1.4.0 Datum: 04.11.2010



Dieses Dokument beschreibt folgendes Produkt:

- **C-863**
DC-Motor Controller, 1 Achse, vernetzbar



Physik Instrumente (PI) GmbH & Co. KG ist Inhaberin der nachfolgend aufgeführten Firmennamen/Marken:
PI®, PIC®, PICMA®, PLine®, PIFOC®, PiezoWalk®, NEXACT®, NEXLINE®, NanoCube®, NanoAutomation®

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

© 2010 Physik Instrumente (PI) GmbH & Co. KG, Karlsruhe, Germany.
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.

Erstdruck 04.11.2010
Dokumentnummer MS173D Benutzerhandbuch, CBo, Version 1.4.0
C-863_User_MS173D_140.doc
Änderungen vorbehalten. Dieses Handbuch verliert seine Gültigkeit mit Erscheinen einer neuen Revision. Die jeweils aktuelle Revision ist unter www.pi.ws zum Herunterladen verfügbar.

Konformitätserklärung

gemäß DIN EN ISO/IEC 17050:2005-01

Hersteller:	Physik Instrumente (PI) GmbH & Co. KG	
Herstelleradresse:	Auf der Römerstraße 1 D-76228 Karlsruhe	

Der Hersteller erklärt hiermit, dass das Produkt

Produktname: **1-Achsen-DC-Motor-Controller**

Modellnummer(n): **C-863**

Produktausführungen: **alle**

die folgenden europäischen Richtlinien erfüllt:

2006/95/EG, Niederspannungsrichtlinie

2004/108/EG, EMV-Richtlinie

Die zum Nachweis der Konformität zugrundegelegten Normen sind nachfolgend aufgelistet.

Elektromagnetische Emission: EN 61000-6-3, EN 55011

Elektromagnetische Störfestigkeit: EN 61000-6-1

Sicherheit (Niederspannungsrichtlinie): EN 61010-1

25. Juni 2009
Karlsruhe



Dr. Karl Spanner
Geschäftsführer

Über dieses Dokument

Zielgruppe dieses Handbuches

Dieses Handbuch soll den Leser bei der Installation und Bedienung des C-863 DC-Motor Controllern unterstützen. Grundsätzliches Wissen zu geregelten Systemen, zu Konzepten der Bewegungssteuerung und zu geeigneten Sicherheitsmaßnahmen wird vorausgesetzt. Das Handbuch beschreibt die physischen Spezifikationen und Abmessungen des C-863 DC-Motor Controllern sowie die Installationsverfahren für Software und Hardware, welche für die Inbetriebnahme des dazugehörigen Positioniersystems erforderlich sind. Dieses Dokument ist auf der Produkt-CD als PDF-Datei verfügbar. Aktualisierte Versionen stehen auf der PI-Website zum Herunterladen zur Verfügung oder sind per E-Mail erhältlich: Kontaktieren Sie Ihren PI-Vertriebsingenieur oder schreiben Sie an info@pi.ws.

Vereinbarungen

Die in diesem Handbuch verwendeten Hinweise und Symbole haben folgende Bedeutungen:

WARNUNG

Kennzeichnet Verfahrensschritte oder Vorkehrungen, die bei nicht ordnungsgemäßer Ausführung oder Befolgung zu Körperverletzung oder Tod führen können.



VORSICHT

Kennzeichnet Verfahrensschritte oder Vorkehrungen, die bei nicht ordnungsgemäßer Ausführung oder Befolgung zu Sachschäden führen können.



HINWEIS

Stellt zusätzliche Informationen oder Anwendungshinweise bereit.

Zugehörige Dokumente

Software-Tools und alle Versteller, welche gegebenenfalls mit dem C-863 DC-Motor Controller ausgeliefert werden, sind in ihren eigenen Handbüchern beschrieben. Sämtliche Dokumente sind auf der Produkt-CD enthalten und können darüber hinaus in ihrer aktuellsten Version als PDF-Dateien von der PI-Website (www.pi.ws) heruntergeladen werden. Wenn Sie aktualisierte Versionen oder frühere Versionen benötigen, kontaktieren Sie Ihren PI-Vertriebsingenieur oder schreiben Sie an info@pi.ws.

MMCRun MS139E	Mercury-Betriebssoftware (native Befehle)
Mercury Native DLL & LabVIEW MS177E	Windows-DLL-Programmbibliothek und LabView-VIs (basierend auf nativen Befehlen)
Mercury Native Commands MS176E	Native Mercury-Befehle
Mercury GCSLabVIEW_MS149E	LabView-VIs, basierend auf PI-GCS-Befehlssatz
Mercury GCS DLL_MS154E	Windows-DLL-Programmbibliothek (GCS-Befehle)
PIMikroMove® User Manual SM148E	PIMikroMove® Betriebssoftware (GCS-basiert)
Mercury GCS Commands MS163E	Mercury-GCS-Befehle
PIStageEditor SM144E	Software für GCS-Versteller-Datenbank

1	Einleitung	3
1.1	Überblick.....	3
1.2	Bestimmungsgemäße Verwendung	4
1.3	Sicherheitsmaßnahmen.....	5
1.4	Auspacken.....	7
1.5	Optionales Zubehör.....	7
1.6	Software-Schnittstellen.....	8
1.6.1	Nativer Befehlssatz.....	8
1.6.2	GCS-Befehlssatz.....	8
1.6.3	Verfügbare Software	9
2	Erste Schritte	11
2.1	Installation der Software	11
2.1.1	GCS-basierte Software	11
2.1.2	Software für den nativen Befehlssatz.....	12
2.2	Inbetriebnahme.....	13
3	Angaben zum Betrieb	17
3.1	Bedienelemente.....	17
3.1.1	Bedienelemente an der Vorderwand.....	17
3.1.2	Einstellung der DIP-Schalter	18
3.1.3	Bedienelemente an der Rückwand	19
3.2	Anschluss des Netzteils.....	20
3.3	Standardeinstellungen.....	21
3.4	Verbindung zum PC	22
3.4.1	Anschluss über RS-232.....	22
3.4.2	Anschluss über USB.....	22
3.4.3	Netzwerkbetrieb.....	23
3.5	Ansteuerungsarten	24
3.5.1	Positionsregelung.....	24
3.5.2	Kraftregelung	25
3.5.3	Positions- und Kraftregelung	25
3.6	Notchfilter	26
3.7	Stellwegsbegrenzung	26
3.7.1	Endschalter.....	26
3.7.2	Softwareseitige Verfahrbereichsgrenzen	27
3.8	Referenzsignal.....	27
3.9	Ein- und Ausgangsleitungen.....	28
3.9.1	Überblick.....	28
3.9.2	Trigger-Ausgang.....	28

4	Updates	29
4.1	Aktualisierung der Firmware.....	29
4.2	Software-Aktualisierungen.....	33
5	Manuelle Bedienhilfen	34
5.1	Joystick-Steuerung.....	34
5.1.1	Zubehör zum Joystick.....	34
5.1.2	Joystick-Lookup-Tabellen.....	35
5.1.3	Kalibrierung der Z-Achse mit dem C-819.30-Joystick.....	35
5.2	Steuerung über die Pushbutton-Box.....	37
5.3	Trackball-Steuerung.....	38
6	Makros	39
7	Störungsbehebung	41
8	Technische Daten	45
8.1	Spezifikationen.....	45
8.2	Abmessungen.....	46
8.3	Pinbelegungen.....	47
8.3.1	Motoranschluss.....	47
8.3.2	I/O-Anschluss.....	48
8.3.3	Joystick-Anschluss.....	49
8.3.4	Y-Kabel für Joystick.....	49
8.3.5	RS-232-Anschlüsse.....	50
8.3.6	USB-Anschluss.....	51
8.3.7	Spannungsversorgungsanschluss und Erdungsschraube.....	51
9	Entsorgung	52

1 Einleitung

Der Mercury DC-Motor Controller ist die perfekte Lösung zum Aufbau preisgünstiger und flexibler Positioniersysteme, bei denen ein Präzisionsverstärker mit einem PC oder programmierbaren Controller angesteuert werden soll. Der C-863 ergänzt den erfolgreichen C-663 Mercury Step Schrittmotor-Controller. Die Controller der Mercury-Baureihe verwenden dieselben Befehlsätze und können miteinander vernetzt werden.

1.1 Überblick

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



Abb. 1: C-863.10 Mercury Controller

Mehrachsenbetrieb von DC- und Schrittmotoren

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

Der C-863 Mercury DC-Motor Controller hat eine gemeinsame Programmiersprache mit dem Mercury Step Schrittmotor-Controller. Bis zu 16 Mercury-Controller (für DC- und Schrittmotoren) können vernetzt und über dieselbe Computerschnittstelle betrieben werden.

Flexible Automatisierung

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

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

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

Benutzerfreundlich: Umfangreiche Software und zwei Schnittstellen

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

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

1.2 Bestimmungsgemäße Verwendung

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

Beachten Sie die Sicherheitshinweise in diesem Benutzerhandbuch.

Der C-863 kann im geregelten Betrieb mit inkrementellen Positionssensoren betrieben werden. Nähere Informationen zum Positionssensor finden Sie in den Produktspezifikationen der Mechanik, mit welcher der C-863 betrieben werden soll.

Der C-863 darf nur gemäß der jeweiligen Gerätespezifikation verwendet werden. Wird der C-863 nicht entsprechend diesem Benutzerhandbuch verwendet, so kann der vorgesehene Schutz beeinträchtigt sein.

Die genaue Prüfung der technischen Spezifikationen durch den Hersteller beinhaltet keine Gültigkeitsprüfung für komplette Anwendungen. Vielmehr ist der Betreiber verantwortlich für die Prüfung von Prozessen und die entsprechenden Freigaben.

Der C-863 ist ein Laborgerät im Sinne der DIN EN 61010. Er erfüllt die nachfolgenden Mindestspezifikationen für einen sicheren Betrieb. Jegliche

weitergehenden Spezifikationen in der Datentabelle (S. 45) werden selbstverständlich auch eingehalten.

- Verwendung nur in Innenräumen
- Höhe bis zu 2000 m
- Temperaturbereich von 5 °C bis 40 °C
- Höchste relative Luftfeuchte 80 % für Temperaturen bis 31 °C, linear abnehmend bis 50 % relativer Luftfeuchte bei 40 °C
- Netzspannungsschwankungen nicht größer als ± 10 % der Nennspannung
- Transiente Überspannungen wie sie üblicherweise im Versorgungsnetz auftreten.

Anmerkung: Der Nenn-Pegel der transienten Überspannung ist die Stehstoßspannung nach Überspannungskategorie II (IEC-60364-4-443).

- Verschmutzungsgrad: 2

1.3 Sicherheitsmaßnahmen

Installieren und betreiben Sie den C-863 DC-Motor Controller erst, nachdem Sie die Bedienungsanleitung gelesen haben. Bewahren Sie das Benutzerhandbuch sowie alle dazugehörigen Technical Notes in der Nähe des Gerätes an einem sicheren Ort griffbereit auf. Sollte die Anleitung verlorengehen oder unbrauchbar werden, fordern Sie vom Hersteller ein neues Exemplar an oder laden Sie es von der Website herunter (www.pi.ws). Lesen Sie alle Anleitungen für die mitgelieferte Software und Mechanik sorgfältig durch.

Nichtbeachten der Sicherheitshinweise in diesem Benutzerhandbuch kann zu Unfällen mit Personenschäden, zu Sachschäden und zum Verlust der Gerätegarantie führen. Beachten Sie, dass der C-863 keine vom Benutzer zu wartenden Teile umfasst.



WARNUNG

Die Bewegungen der angeschlossenen Motoren und Versteller sind softwaregesteuert, und Software kann versagen. Beachten Sie, dass motorisierte Versteller große Kräfte erzeugen, die bei fehlerhafter Bedienung zu Verletzungen oder Sachschäden führen können.

VORSICHT

Falls Sie verschiedene Mercury-Controller verwenden, achten Sie auf die korrekte Zuordnung der Netzteile für 12 V, 15 V oder 24 V Versorgungsspannung!



VORSICHT

Verbinden Sie niemals einen Schrittmotor mit einem C-863 DC-Motor Controller. Dies kann irreparable Schäden verursachen!



VORSICHT

Aktivieren Sie den Joystick-Eingang in der Software nur dann, wenn tatsächlich ein Joystick an den Controller angeschlossen ist. Andernfalls können unkontrollierte Bewegungen der angeschlossenen Achse auftreten, die die Anwendung gefährden können.



VORSICHT

Die Ausgangsspannung an den Anschlüssen "Motor +" und "Motor -" kann so hoch wie die Versorgungsspannung sein. Falls Sie kein Standardnetzteil für den Mercury-Controller verwenden und ein Motor an diese Leitungen angeschlossen ist (d. h. Anschluss ohne separaten PWM-Verstärker), dann stellen Sie sicher, dass die Betriebsspannung des Motors nicht überschritten wird.



VORSICHT

Verbinden Sie niemals gleichzeitig den USB- und den RS-232-Anschluss eines Controllers mit Ihrem PC. Dies kann Schäden verursachen.



VORSICHT

Bei fehlendem Schutzleiter können gefährliche Berührungsspannungen entstehen, und es besteht Stromschlagsgefahr. Im Falle eines Fehlers oder Defekts kann das Berühren des Controllers zu schweren Verletzungen oder zum Tode führen.

Stellen Sie sicher, dass das Metallgehäuse über die Erdungsschraube unten links an der Geräterückwand geerdet bzw. an einen Schutzleiter angeschlossen ist.



1.4 Auspacken

Packen Sie den C-863 Mercury DC-Motor Controller vorsichtig aus. Vergleichen Sie den Inhalt mit dem Inhalt laut Bestellung und mit der Packliste.

Im Lieferumfang des C-863.10 ist Folgendes enthalten:

- Mercury-Controller (C-863.10)
- Weitbereichsnetzteil für 15 V (C-890.PS)
- Netzkabel
- RS-232-Nullmodemkabel zur Verbindung mit dem PC (C-815.34, 3 m)
- RS-232-Kabel für Daisy-Chain-Netzwerk (C-862.CN, 28 cm)
- USB-Kabel (Typ A auf Mini-B) zur Verbindung mit dem PC (000014651)
- Mercury-Produkt-CD mit der gesamten Software und allen Benutzerhandbüchern für Mercury-Produkte
- *MS173D* Benutzerhandbuch in gedruckter Form (dieses Dokument)
- *MS163E Mercury GCS Commands* Benutzerhandbuch in gedruckter Form
- *MS176E Mercury Native Commands* Benutzerhandbuch in gedruckter Form

Bei Anzeichen von Beschädigungen oder fehlenden Teilen wenden Sie sich sofort an PI.

Bewahren Sie das komplette Verpackungsmaterial auf für den Fall, dass das Produkt erneut versendet werden muss.

1.5 Optionales Zubehör

Die folgenden Komponenten können zusätzlich bestellt werden:

Bestellnummer

C-815.38	Motorkabel, 3m, D-Sub, 15-pol., m/f
C-862.CN2	Netzwerkkabel für Mercury-Controller, 180 cm
C-819.20	Analoger Joystick, 2 Achsen
C-819.20Y	Y-Kabel für 2 Mercury-Controller an Joystick
C-819.30	Analoger Joystick für Mercury-Controller, 3 Achsen
C-170.PB	Pushbutton-Box mit 4 Tasten und 4 LEDs
C-170.IO	I/O-Anschlusskabel, offenes Ende (S. 48)
C-663.PS	24-V-Netzteil, vorrangig für Systeme mit 24-Volt-DC-Motoren ohne PWM-Verstärker; nicht für 12-V-Motoren verwenden.

1.6 Software-Schnittstellen

Viele Funktionen und Befehle des C-863 Mercury DC-Motor Controllers sind schon von anderen PI-Controllern der Mercury-Produktreihe bekannt. Zur Ansteuerung der Mercury-Controller können sowohl der native Befehlssatz als auch der GCS-Befehlssatz (PI General Command Set) verwendet werden. Die Befehle dienen der Einstellung des Betriebsmodus, lösen Bewegungen aus oder fragen System- und Bewegungswerte ab.

1.6.1 Nativer Befehlssatz

Der native Befehlssatz spricht den Controller direkt an. Er kann mit jeder beliebigen Terminal-Software und mit *MMCRun* benutzt werden. Die meisten nativen Mercury-Befehle beginnen mit einer Kombination aus zwei Buchstaben. Die Befehle und die entsprechende Syntax werden im "Mercury Class Native Commands Software Manual" (MS176E) detailliert beschrieben.

1.6.2 GCS-Befehlssatz

Der GCS (PI General Command Set) als der Standardbefehlssatz von PI garantiert die Kompatibilität verschiedener Controller. Mit der aktuellen Version der Mercury-Firmware wird die GCS-Unterstützung durch eine Windows-DLL gewährt, welche die GCS-Befehle in die nativen Befehle übersetzt (Einzelheiten finden Sie in den Handbüchern "Mercury GCS DLL Manual" (MS154E) sowie "Mercury GCS Commands Manual" (MS163E)). Die meisten GCS-Befehle und die entsprechenden DLL-Funktionsaufrufe sind durch eine Kombination aus drei Buchstaben gekennzeichnet.

HINWEISE

Die GCS-DLL bietet auch die Möglichkeit, native Befehle direkt an den Controller zu senden. Dennoch ist es nicht empfehlenswert, GCS-Befehle und native Befehle zu kombinieren. Beispielsweise können Bewegungsbefehle in GCS nicht korrekt ausgeführt werden, wenn die Position vorher mit einem nativen Befehl geändert wurde.

GCS-Befehle können im Fenster "Command entry" der GCS-kompatiblen Benutzersoftware wie z. B. PIMikroMove® eingegeben werden (vgl. PIMikroMove® Handbuch auf der Produkt-CD).

1.6.3 Verfügbare Software

Beim C-863 DC-Motor Controller sind alle Bewegungen der angeschlossenen Motorachse softwaregesteuert. Für größtmögliche Flexibilität bietet PI Software für verschiedene Integrationsstufen mit der entsprechenden Dokumentation an. Die meisten Programme und Treiberbibliotheken sind in eigenen Handbüchern beschrieben. Aktualisierte Versionen sind unter www.pi.ws oder per E-Mail erhältlich: Kontaktieren Sie Ihren PI Vertriebsingenieur oder schreiben Sie an info@pi.ws.

- *PITerminal* ist ein Windows-Programm, das mit fast allen Controllern von PI als einfaches Terminal verwendet werden kann. PITerminal ist unabhängig vom verwendeten Befehlssatz und unterstützt die Verbindung zu Controllern der Mercury-Reihe sowohl über RS-232 als auch über USB. Die USB-Verbindung wird dabei von der Host-Software wie ein COM-Port behandelt. Wenn mit GCS-Befehlen gearbeitet werden soll, muss mit der derzeitigen Firmware die Verbindung zum PC über die Schaltfläche "GCS DLL..." aufgebaut werden. Wird stattdessen die Schaltfläche "Connect..." verwendet, dann erwartet die aktuelle Mercury-Firmware Befehle aus dem nativen Befehlssatz.

Software für den nativen Befehlssatz

- *MMCRun* (für Windows™ 95/98/2000/XP und NT) ist ein Anwenderprogramm für C-863 und C-862 Mercury Controller und für C-663 Mercury Step Controller. *MMCRun* erlaubt die einfache Kommandierung und Makroprogrammierung und spricht die Controller der Mercury-Reihe direkt an. Eine ausführliche Beschreibung finden Sie im *MMCRun*-Handbuch MS139E.
- Die Windows-DLL "MMC410.DLL" unterstützt die Programmierer individueller Anwendungen bei vielen Aufgaben, z. B. im Hinblick auf Schnittstellen und Datenkonvertierung. Diese DLL beruht auf dem nativen Befehlssatz (Einzelheiten siehe "Mercury Native DLL and LabView Manual", MS177).

- LabVIEW-VIs erleichtern die Einbindung von Controllern der Mercury-Reihe in eine LabVIEW-Umgebung. Verwendet die o. g. MMC410.DLL, die auf dem nativen Befehlssatz beruht (Einzelheiten siehe "Mercury Native DLL and LabView Manual", MS177).
- *MMC_Z-Joy* ist ein Software-Tool, das für die Kalibrierung der Z-Achse eines analogen, dreiachsigen C-819.30-Joysticks erforderlich ist. Es errechnet eine spezielle Lookup-Tabelle und speichert sie im entsprechenden Controller.

GCS-basierte Software*

- PIMikroMove® für Microsoft Windows ist eine Anwendersoftware, die für diesen und viele andere PI-Controller verwendet wird. PIMikroMove® ermöglicht Ihnen, Ihr Positioniersystem – PC, Controller und Verstärker – sofort zu starten, ohne kundenspezifische Software erstellen zu müssen. Dank der Anzeigen und Funktionen von PIMikroMove® kann häufig darauf verzichtet werden, sich mit ASCII-Befehlen auseinanderzusetzen. PIMikroMove® unterstützt auch die Eingabe von Befehlen, so dass Sie die verschiedenen GCS-Befehle leicht ausprobieren können. Die Kommandierung eines einzelnen Controllers oder des Controller-Netzwerks erfolgt mit PIMikroMove® über die unten beschriebene GCS-DLL.
- GCS-LabVIEW-Treiber ermöglichen über die Mercury-GCS-DLL die Kommunikation mit dem C-863-Controller aus einer vorhandenen LabVIEW-Umgebung von National Instruments heraus (Einzelheiten siehe Handbuch MS149E).
- GCS-DLL (Windows Dynamic Link Library): Die Mercury General Command Set Dynamic Link Library (Mercury-GCS-DLL) ermöglicht Windows-Programmen den einfachen Zugriff auf den Controller. Handhabung und Funktionalität der DLL sind in einem separaten Handbuch beschrieben (MS154E). Die meisten DLL-Funktionen entsprechen direkt einem Befehl aus dem PI General Command Set (PI-GCS-Befehlssatz).

HINWEIS

Die GCS-LabVIEW-Treiber und die GCS-Programmbibliothek sind auch für Linux-Betriebssysteme (Kernel 2.6, GTK 2.0, Glibc 2.4) erhältlich.

* Diese Software basiert auf dem standardisierten GCS-Befehlssatz (PI General Command Set), wobei die GCS-Befehle mit Hilfe der Mercury-GCS-DLL übersetzt werden. Die so übersetzten Befehle können dann von der Mercury-Firmware, die auf dem nativen Befehlssatz beruht, verarbeitet werden.

2 Erste Schritte

2.1 Installation der Software

Auch wenn der Betrieb mit kundeneigener Software erfolgen soll, ist die Software von PI doch sehr komfortabel, um Systemtests durchzuführen. Sie können *PIMikroMove®*, *MMCRun* oder *PITerminal* verwenden (auf der Produkt-CD). Siehe Abschnitt 1.6, S. 8 für mehr Informationen zu diesen Programmen.

Dieser Abschnitt beschreibt die zur Installation der verschiedenen Programme erforderlichen Maßnahmen.

HINWEIS

Die PI-Software wird ständig weiterentwickelt. Daher sollten Sie auf der PI-Website (www.pi.ws) regelmäßig nachschauen, ob Software-Updates zum Herunterladen verfügbar sind. Die Updates enthalten Informationen (Readme-Dateien), so dass Sie selbst entscheiden können, ob die Aktualisierung im Hinblick auf Ihre Anwendung sinnvoll ist. Sie benötigen ein Passwort, damit Sie Updates suchen und herunterladen können. Dieses Passwort finden Sie auf der Mercury-CD in der PDF-Datei mit den Mercury-Release-News im Verzeichnis \Manuals. Einzelheiten zum Herunterladen siehe "Software-Aktualisierungen" (S. 33).

2.1.1 GCS-basierte Software

Um die GCS-basierte Software zu verwenden, gehen Sie wie folgt vor:

Windows-Betriebssysteme:

- 1 Legen Sie die Mercury-CD in Ihren PC ein.
- 2 Falls das Installationsprogramm nicht automatisch startet, öffnen Sie die Datei *setup.exe* im Hauptverzeichnis der CD.
- 3 Folgen Sie den Bildschirmanweisungen und wählen Sie die typische Installation ("Typical") aus. Hierin enthaltene Komponenten sind LabVIEW-Treiber, GCS-DLL und PIMikroMove®.

Linux-Betriebssysteme:

- 1 Legen Sie die Mercury-CD in Ihren PC ein.
- 2 Öffnen Sie ein Konsolen-Fenster und gehen Sie in das Verzeichnis `/linux` auf der Mercury-CD.
- 3 Loggen Sie sich als superuser (root) ein.

- 4 Starten Sie das Installationsscript mit `./INSTALL`. Achten Sie auf die Groß- und Kleinschreibung in Linux, wenn Sie den Befehl eingeben.
- 5 Folgen Sie den Bildschirmanweisungen. Sie können die zu installierenden Komponenten einzeln auswählen.

Sollte die Installation fehlschlagen, dann stellen Sie sicher, dass die Kernel-Header-Dateien für Ihren Kernel installiert sind.

Die Verstellerdatenbankdatei "PIStages2.dat", die von der GCS-basierten Software benötigt wird, ist im Verzeichnis `...\PI\GcsTranslator` installiert. In diesem Verzeichnis befindet sich auch die Datenbank "MercuryUserStages2.dat", die automatisch angelegt wird, wenn Sie die Verbindung zwischen Versteller und Software erstmals herstellen (d. h. wenn die Funktionen VST? oder CST der GCS-Programmibibliothek erstmals verwendet werden).

Das PI-Verzeichnis wird bei der Installation angegeben und befindet sich normalerweise unter `C:\Dokumente und Einstellungen\All Users\Anwendungsdaten (Windows XP)` oder `C:\ProgramData (Windows Vista und Windows 7)`. Existiert dieses Verzeichnis nicht, so wird die EXE-Datei, welche die Verstellerdatenbanken benötigt, in ihrem eigenen Verzeichnis nachschauen.

2.1.2 Software für den nativen Befehlssatz

Die Installation von Software, die auf dem nativen Befehlssatz beruht, kann ebenfalls über die Windows-Installationsroutine erfolgen. Das Installationsprogramm legt die Software in Unterverzeichnissen des Installationsverzeichnisses ab und fügt die Programme für *MMCRun* und das Mercury-Firmware-Update zur Programmgruppe *PI* unter *Start→Alle Programme→PI→Mercury* hinzu.

Sofern Sie *ausschließlich* Software für den nativen Befehlssatz verwenden, können Sie das Installationsprogramm auch wie folgt umgehen (manuelle Installation):

- 1 Legen Sie die Produkt-CD in das Laufwerk Ihres PCs ein.
- 2 Schließen oder ignorieren Sie das Installationsprogramm, falls es automatisch startet.
- 3 Öffnen Sie den Windows-Explorer und wählen Sie das CD-Laufwerk aus.
- 4 Kopieren Sie die Inhalte des Verzeichnisses *Mercury_Native_Software* von der CD in ein Verzeichnis auf Ihrem PC.
- 5 Greifen Sie von diesem Verzeichnis aus auf die Programme zu, die Sie benötigen. Wenn Sie z. B. die Software *MMCRun* starten möchten, dann gehen Sie in dieses Verzeichnis, öffnen Sie das Unterverzeichnis *MMC_Run* und doppelklicken Sie auf die `.exe`-Datei für *MMCRun* in diesem Verzeichnis.

2.2 Inbetriebnahme

Dieser Abschnitt geht von der Annahme aus, dass Sie Controller der Mercury-Baureihe einzeln oder im Netzwerk über RS-232- oder USB-Anschluss mit dem PC verbinden und ansteuern möchten, wobei an jedem Controller je ein Motor angeschlossen ist. Beachten Sie auch die Informationen in den Benutzerhandbüchern aller sonstigen Systembestandteile.

- 1 Geräteadresse einstellen:
Stellen Sie sicher, dass die verwendeten Controller geeignet nummeriert sind. Ab Werk sind die DIP-Schalter für die Adresse alle auf ON voreingestellt, was der Gerätenummer 1 entspricht. Innerhalb eines Netzwerks muss jedem Controller der Mercury-Reihe eine eindeutige Adresse zugewiesen sein. Weitere Informationen zum Einstellen der Geräteadresse siehe "Einstellung der DIP-Schalter" auf S. 18.
- 2 Controller oder Netzwerk mit PC verbinden:
Verwenden Sie für die Verbindung zwischen dem ersten Controller und dem PC entweder das USB-Kabel oder das RS-232-Nullmodem-Kabel, aber keinesfalls beide Kabel gleichzeitig.
Ein Netzwerk aus mehreren Controllern wird in Reihe geschaltet. Die Verbindung erfolgt jeweils vom "RS-232 Out"-Ausgang eines Gerätes zum "RS-232 In"-Eingang des nachfolgenden Gerätes. Die Baudrate (DIP-Schalter 5 und 6 am C-863) muss für alle Controller im Netzwerk gleich eingestellt werden. Sie muss derjenigen Baudrate entsprechen, die von der Software erwartet wird.

VORSICHT

Verbinden Sie niemals einen Schrittmotorantrieb mit einem C-863 DC-Motor Controller. Dies kann irreparable Schäden verursachen.



- 3 Verbinden Sie jede Motorachse mit dem entsprechenden Controller.
- 4 Schließen Sie einen Schutzleiter an die Erdungsschraube an der Rückwand jedes Controllers an.

VORSICHT

Die Ausgangsspannung an den Anschlüssen "Motor +" und "Motor -" kann so hoch wie die Versorgungsspannung sein. Falls Sie kein Standardnetzteil für den Mercury-Controller verwenden und ein Motor an diese Leitungen angeschlossen ist (d. h. Anschluss ohne separaten PWM-Verstärker), dann stellen Sie sicher, dass die Betriebsspannung des Motors nicht überschritten wird.

Beachten Sie stets die Kennzeichnung der Versorgungsspannung.



- 5 Verbinden Sie jeden Controller mit der Netzspannung. Dabei ist besondere Sorgfalt erforderlich, wenn auch Mercury-Controller des Typs C-862 (schwarz) im System eingebunden sind.

Die Status-LED "STA" leuchtet nun entweder rot (Servobetrieb aus; Standardmodus) oder grün (Servobetrieb ein; z. B. durch Autostart-Makro eingeschaltet) und zeigt damit Normalbetrieb an.

Beachten Sie, dass bei der erstmaligen Verwendung der USB-Schnittstelle zwei FTDI-USB-Treiber auf dem PC installiert werden müssen. Diese Treiber befinden sich auf der Mercury-CD im Verzeichnis \USB_Driver. Folgen Sie den Bildschirmanweisungen. Die Installation der USB-Treiber erfordert Administratorenrechte auf dem PC.

- 6 Starten Sie die gewünschte PI-Software und richten Sie sie für den (oder die) mit Ihrem System verbundenen Verstärker ein. Bei den nachfolgenden Abbildungen handelt es sich um Screenshots aus *PIMikroMove®*; stattdessen kann auch *MMCRun* verwendet werden; weitere Einzelheiten zur Benutzung der Software sind in den entsprechenden Software-Handbüchern zu finden.

HINWEIS

Bei angeschlossenem und eingeschalteten Controller wird, sobald die USB-Treiber installiert sind, die USB-Verbindung von der Software wie ein zusätzlicher COM-Port behandelt.

Die Baudrate, die vom Rechner verwendet wird, muss mit der über die DIP-Schalter eingestellten Baudrate übereinstimmen. Das gilt auch dann, wenn die USB-Schnittstelle verwendet wird!

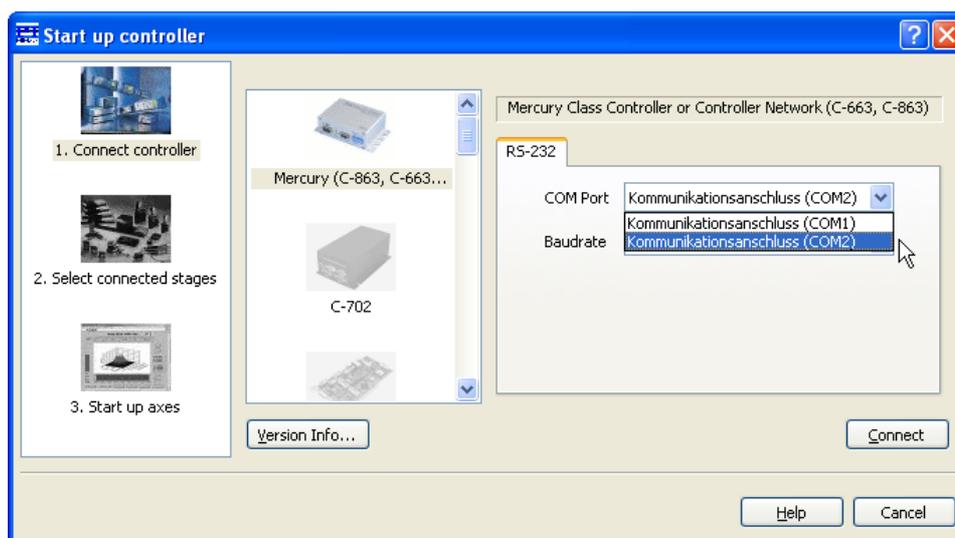


Abb. 2: *PIMikroMove®*, Bildschirmdialog "Start up controller"; Schritt 1: Ein "zusätzlicher" COM-Port für den USB-Anschluss ist ausgewählt.

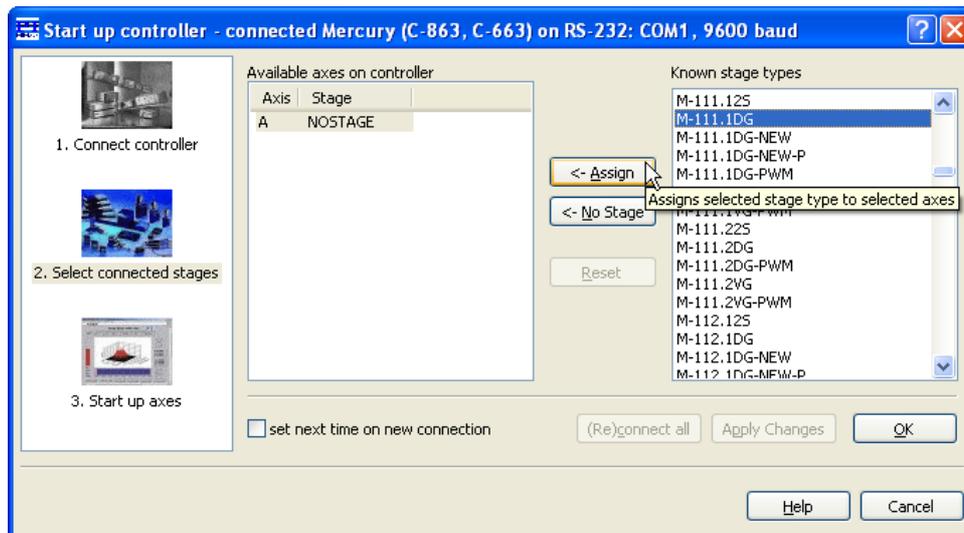


Abb. 3: PIMikroMove®, Start up controller; Schritt 2: Wählen Sie den Versteller aus. Bevor Sie mit Schritt 3 fortfahren, klicken Sie auf Assign und bestätigen Sie mit OK.

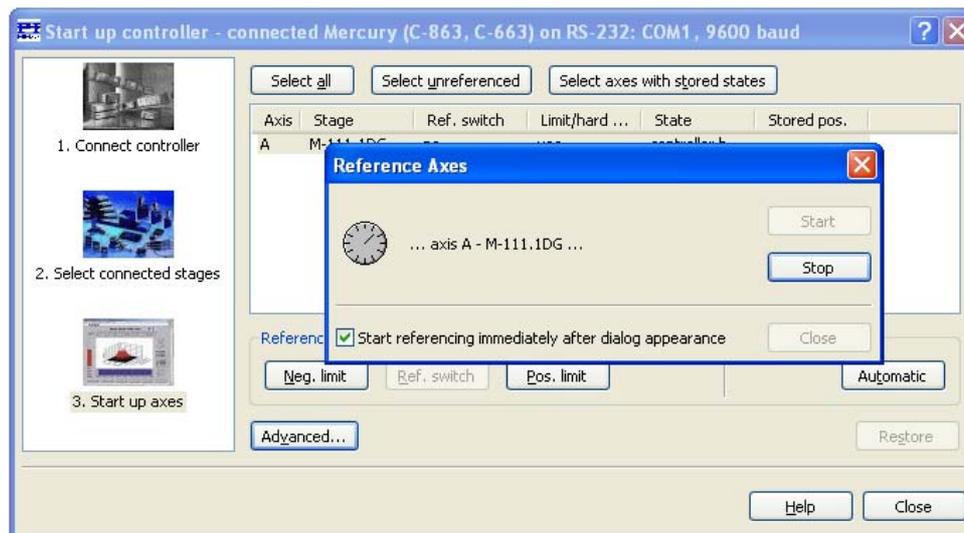


Abb. 4: PIMikroMove®, Start up controller, Schritt 3: Referenzieren Sie die Achse.

- 7 Stellen Sie nach dem Aufbau der Verbindung sicher, dass die Bewegungsparameter aller Achsen so eingestellt sind, wie sie für die Mechanik benötigt werden. Sofern Informationen über den Typ oder die Baureihe des Verstellers an die Software übertragen wurden (wie in Abb. 3), werden Startparameter eingestellt. Andernfalls kann es erforderlich sein, die Werte für die Verstellparameter direkt einzugeben. Geeignete Startwerte sind im Benutzerhandbuch des Verstellers angegeben. Beachten Sie, dass PIMikroMove® bei der Verstellerauswahl einen INI-Befehl sendet, der die Achsen initialisiert und darüber hinaus den Servobetrieb einschaltet.

- 8 Stellen Sie sicher, dass der Controller die absolute Position jeder Achse kennt (d. h. dass die Achse referenziert wird). Falls Sie die Software nicht zur Referenzierung aufgefordert hat (wie in Abb. 4), dann nimmt sie möglicherweise an, dass die aktuelle Position 0 ist. Sorgen Sie dafür, dass sich die Achse in Richtung eines End- oder Referenzschalters bewegt und dort anhält. Einzelheiten finden Sie in den entsprechenden Software- oder Befehlshandbüchern.
- 9 Starten Sie einige Testbewegungen auf der Achse:
Im Hauptfenster von *PI MikroMove*® können Sie z. B. Schritte mit einer bestimmten Schrittweite ausführen, indem Sie die entsprechenden Pfeiltasten für die Achse drücken.

3 Angaben zum Betrieb

3.1 Bedienelemente

3.1.1 Bedienelemente an der Vorderwand



Abb. 5: C-863 Mercury DC-Motor Controller, Vorderansicht

Vorderwand		
Beschriftung	Typ	Funktion
RS-232 In	D-Sub 9(m)	Serielle Verbindung zum PC oder zum vorhergehenden Controller in einem Daisy-Chain-Netzwerk
RS-232 Out	D-Sub 9(f)	Serielle Verbindung zum nachfolgenden Controller in einem Daisy-Chain-Netzwerk
	Mini-USB Typ B	Universal Serial Bus für Verbindung zum PC; nicht verbinden, wenn "RS-232 In" bereits verbunden ist.
STA	LED rot/grün/aus	Controller-Status: grün: Servobetrieb ein (geregelter Betrieb) rot: Servobetrieb aus aus: Firmware-Update-Modus (Auswahl über DIP-Schalter 8)
ERR	LED rot/aus	Fehlerstatus: rot: Fehler. Nach der Auslese des Fehlerstatus über den entsprechenden Befehl wird der Fehlerstatus wieder gelöscht (LED wird ausgeschaltet). aus: Kein Fehler
Mode, Baud, Addr	8-Bit-DIP-Schalter	Einstellung der Geräteadresse, der RS-232-Baudrate für die Kommunikation mit dem PC, der Endschaltersignallogik und des Firmware-Update-Modus (siehe Einstellung der DIP-Schalter, S. 18).

3.1.2 Einstellung der DIP-Schalter

Schalter 1 bis 4:	Adressierung / Gerätenummer (16 mögliche Kombinationen)
Schalter 5 und 6:	Baudrate
Schalter 7	Signallogik der Endschalter
Schalter 8	Firmware-Update-Modus

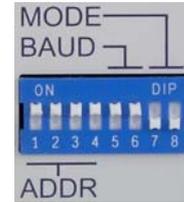


Abb. 6: Schalter oben
ON
Schalter unten
OFF

Werkseitige Voreinstellungen sind fett gedruckt

Geräte-Nr.	Schalter-wert	SW1	SW2	SW3	SW4
1	0	ON	ON	ON	ON
2	1	ON	ON	ON	OFF
3	2	ON	ON	OFF	ON
4	3	ON	ON	OFF	OFF
5	4	ON	OFF	ON	ON
6	5	ON	OFF	ON	OFF
7	6	ON	OFF	OFF	ON
8	7	ON	OFF	OFF	OFF
9	8	OFF	ON	ON	ON
10	9	OFF	ON	ON	OFF
11	10	OFF	ON	OFF	ON
12	11	OFF	ON	OFF	OFF
13	12	OFF	OFF	ON	ON
14	13	OFF	OFF	ON	OFF
15	14	OFF	OFF	OFF	ON
16	15	OFF	OFF	OFF	OFF

Baudrate*	SW5	SW6
9600	ON	ON
19200	ON	OFF
38400	OFF	ON

*Die anderen Einstellungen sind hardwareseitig mit 8 Daten-Bits und 1 Stopp-Bit festgelegt, ohne Parität; interne Zwischenspeicher erfordern keinen Handshake; *MMCRun* unterstützt nur 9600 Baud.

Endschalter (Hardware-Einstellung)*	SW7
Active Low	ON
Active High	OFF

*Die Hardware-Einstellung des Endschalters muss mit der Software-Einstellung übereinstimmen.

Firmware-Update	SW8
Update	ON
Normalbetrieb	OFF

HINWEISE

Geänderte DIP-Schalter-Einstellungen werden nach dem nächsten Einschalten oder Reset wirksam.

Beim nativen Befehlssatz entspricht die Controller-Adresse dem Wert, der mit den DIP-Schaltern 1 bis 4 angegeben wird (siehe "Schalterwert").
Beim GCS-Befehlssatz entspricht die Controller-Adresse der Gerätenummer (siehe "Geräte-Nr.").

3.1.3 Bedienelemente an der Rückwand



Abb. 7: C-863 Mercury DC-Motor Controller, Rückansicht

Rückwand

Beschriftung	Typ	Funktion
15-30 VDC	Hohlstecker-Buchse	Anschluss für die Versorgungsspannung, Mittelleiter
I/O	Mini-DIN-Buchse, 9-polig	Digitaler Ein-/Ausgang, analoger Eingang
Joystick	Mini-DIN-Buchse, 6-polig	Analoger Joystick (Eingang)
DC Motor only	D-Sub 15(f)	Verbindung zum Motor/Versteller (I/O): Nur für DC-Motoren!
	Schraube und Unterlegscheibe	Erdungs- bzw. Schutzleiteranschluss

VORSICHT

Verbinden Sie niemals einen Schrittmotor mit einem C-863 DC-Motor Controller. Dies kann irreparable Schäden verursachen!



3.2 Anschluss des Netzteils

VORSICHT

Die Ausgangsspannung an den Anschlüssen "Motor +" und "Motor -" kann so hoch wie die Versorgungsspannung sein. Falls Sie kein Standardnetzteil für den Mercury-Controller verwenden und ein Motor an diese Leitungen angeschlossen ist (d. h. Anschluss ohne separaten PWM-Verstärker), dann stellen Sie sicher, dass die Betriebsspannung des Motors nicht überschritten wird.

Der C-863 wird mit einem 15-V-Weitbereichsnetzteil geliefert, das für Netzspannungen zwischen 100 und 240 Volt Wechselfspannung bei 50 oder 60 Hz geeignet ist. Es ist möglich, ein anderes geeignetes Netzteil zu verwenden, das 15 bis 30 Volt Gleichspannung liefert.

Um den C-863 einzuschalten, gehen Sie wie folgt vor:

- 1 Stellen Sie sicher, dass das Gehäuse des C-863 über die mit dem Erdungssymbol gekennzeichnete Erdungsschraube an der Geräterückwand geerdet bzw. mit einem Schutzleiter verbunden ist (siehe Abbildung unten).



- 2 Verbinden Sie das mitgelieferte Weitbereichsnetzteil mit dem Anschluss "15-30 VDC" des C-863.
- 3 Verbinden Sie das Netzkabel des Netzteils mit der Wandsteckdose. Die Status-LED "STA" an der Vorderwand des C-863 leuchtet nun entweder rot (Servobetrieb aus; Standardmodus) oder grün (Servobetrieb ein; z. B. durch Autostart-Makro eingeschaltet) und zeigt damit Normalbetrieb an. Falls die LED "STA" nicht leuchten sollte, überprüfen Sie die Einstellungen der DIP-Schalter an der Vorderwand des C-863: Alle LEDs bleiben ausgeschaltet, wenn sich der C-863 im Firmware-Update-Modus befindet (DIP-Schalter 8 steht auf Stellung "ON").

3.3 Standardeinstellungen

Beim Einschalten des C-863 Mercury DC-Motor Controllers sind folgende Werte eingestellt:

Parameter	Werkseitige Einstellung	Einheit
Geschwindigkeit:	45000	counts/s
Beschleunigung:	400000	counts/s ²
P-Term für Positionsregelung:	35	
I-Term für Positionsregelung:	0	
D-Term für Positionsregelung:	0	
I-Limit für Positionsregelung:	2000	
Regelungsmodus	0 (Positionsregelung)	
P-Term für Kraftregelung:	35	
I-Term für Kraftregelung:	0	
D-Term für Kraftregelung:	0	
I-Limit für Kraftregelung:	2000	
Tiefpassfilter für zusätzlichen Sensoreingang	10000	Hz
Notchfilterfrequenz	10000	Hz
Notchfilterflanke	0,8	
Endschalterlogik:	LH (active high)	
Endschalteraktivierung:	LN (Endschalter aktiviert)	
Bremsmodus:	BN (Bremse ein)	
Obergrenze des Verfahrbereichs (softwareseitig)	+100.000.000	counts
Untergrenze des Verfahrbereichs (softwareseitig)	-100.000.000	counts
Einschwingzeit	0 ms (d. h. das On-Target-Flag wird gesetzt, wenn das Ende der errechneten Trajektorie erreicht ist)	
Einschwingtoleranz	5	counts

Die Werte können jederzeit über einen entsprechenden Befehl geändert werden (z. B. über ein Makro). Sobald ein bekannter Versteller angeschlossen wird,

schickt die Software *PIMikroMove*® automatisch Befehle, um geeignete Werte einzustellen. *MMCRun* verhält sich genauso, sobald eine Versteller-Baureihe auf dem Startbildschirm ausgewählt wird.

Häufig sind auch die Werte, die in der Software oder im Benutzerhandbuch für den Versteller vorgegeben sind, nicht optimal für Ihre Anwendung.

Kundenspezifische Parameterwerte können leicht über das Autostart-Makro des Controllers (Makro #0) eingegeben werden. Die werkseitigen Parameter (Default-Einstellungen) können nicht überschrieben werden.

3.4 Verbindung zum PC

VORSICHT

Verbinden Sie niemals gleichzeitig den USB- und den RS-232-Anschluss eines Controllers mit Ihrem PC. Dies kann Schäden verursachen.

Verwenden Sie für die Verbindung des Controllers oder Controller-Netzwerks mit dem PC entweder die RS-232- oder die USB-Schnittstelle. Verbinden Sie zusätzliche Mercury-Controller in einem RS-232-Netzwerk in Reihe mit 1:1-Kabeln, indem Sie jeweils den Anschluss "RS-232 Out" des vorhergehenden Controllers mit dem Anschluss "RS-232 In" des nachfolgenden Controllers verbinden. Jedem Controller im Netzwerk muss über die DIP-Schalter 1 bis 4 eine eindeutige Adresse zugewiesen werden.

Die Baudrate des PCs muss mit der über die DIP-Schalter 5 und 6 eingestellten Baudrate übereinstimmen. Das gilt auch, wenn die USB-Schnittstelle verwendet wird. Die Standard-Baudrate beträgt 9600 Baud (DIP-Schalter 5 und 6 in Stellung "ON"). Die anderen Mercury-Einstellungen sind hardwareseitig mit 8 Daten-Bits und 1 Stopp-Bit festgelegt, ohne Parität; interne Zwischenspeicher erfordern keinen Handshake.

HINWEISE

Geänderte DIP-Schalter-Einstellungen werden nach dem nächsten Einschalten oder Reset wirksam.

Beachten Sie, dass *MMCRun* nur 9600 Baud unterstützt.

3.4.1 Anschluss über RS-232

Verbinden Sie das RS-232-Nullmodemkabel C-815.34 (F-F) mit der "RS-232 In"-Buchse und dem gewünschten COM-Port des PCs.

3.4.2 Anschluss über USB

Wenn Sie die Verbindung über die USB-Schnittstelle erstmalig herstellen, dann erscheint nach dem Einschalten des C-863 ein Dialog mit dem Inhalt, dass neue Hardware gefunden wurde. Folgen Sie den Bildschirmanweisungen und öffnen Sie

im Hardware-Assistenten das Verzeichnis \USB_Driver auf der Produkt-CD. Die Installation der USB-Treiber erfordert Administratorenrechte auf dem betreffenden PC.

HINWEIS

Die USB-Schnittstelle wird für die auf dem PC installierte Software als neuer COM-Port konfiguriert. Dieser COM-Port ist nur dann sichtbar, wenn der Controller tatsächlich über den USB-Anschluss verbunden und eingeschaltet ist. Auch bei Verwendung der USB-Schnittstelle muss die Baudrate am PC und an allen angeschlossenen Geräten einheitlich eingestellt sein.

Für den Kommunikationsaufbau kann es mit der aktuellen Firmware erforderlich sein, den Controller nach der Treiberinstallation aus- und wieder einzuschalten.

3.4.3 Netzwerkbetrieb

C-863 Mercury DC-Motor Controller können untereinander oder z. B. mit C-663 Mercury Step Schrittmotor-Controllern vernetzt werden.

Dabei können bis zu 16 Controller über einen einzigen RS-232- oder USB-Anschluss angeschlossen werden. Die Vernetzung erfolgt in Reihe über eine RS-232-Busarchitektur (Daisy-Chain-Prinzip) mit 1:1-Netzwerkkabeln (Bestellnummer C-862.CN), indem jeweils der Anschluss "RS-232 Out" des vorhergehenden Controllers mit dem Anschluss "RS-232 In" des nachfolgenden Controllers verbunden wird.



Abb. 8: C-862.CN Netzwerkkabel für die Mercury-Baureihe, Länge 28 cm, andere Maße auf Anfrage

Der RS-232-Ausgang mancher PCs unterstützt maximal 6 Einheiten; bei Problemen verwenden Sie die USB-Schnittstelle des PCs.

Jedem Controller im Netzwerk muss eine eindeutige Adresse zugewiesen werden. Die Reihenfolge des Anschlusses ist unerheblich. Beim C-863 wird die Controlleradresse über die ersten vier DIP-Schalter an der Gerätevorderwand eingestellt (siehe S. 18). Außerdem muss die Baudrate für alle Controller im Daisy-Chain-Netzwerk gleich eingestellt werden (DIP-Schalter 5 und 6).

Innerhalb des Netzwerks ist keine simultane ASCII-Kommunikation mit den einzelnen Controllern möglich. Dennoch können die angeschlossenen Achsen

gleichzeitig Bewegungen ausführen, Servoprozeduren und/oder Makrobefehle abarbeiten.

Der GCS-Befehlssatz unterscheidet sich vom nativen Befehlssatz prinzipiell in der Art der Kommunikation. Einzelheiten finden Sie in den jeweiligen Befehlshandbüchern. Die derzeitige Firmware verwendet für die Hardware den nativen Befehlssatz, d. h. die GCS-Befehle werden mit Hilfe der GCS-DLL in native Mercury-Befehle übersetzt, und der Controller wird über die native Adress-Codierung ausgewählt.

Alle Controller im Netzwerk können ihre individuellen Makros gleichzeitig ausführen. Dabei gibt es keine direkte Kommunikation zwischen den Controllern, sondern nur zwischen Controller und PC. Die Adressierung und die Sequenzierung der Bewegung der jeweiligen Achse werden vom PC-Programm übernommen. Diese Art der Kommunikation begrenzt die Möglichkeiten der Pfad-Interpolation und der Mehrachsen-Steuerung, ebenso wie die bedingte Ausführung von Bewegungsbefehlen.

Beispiele für die Programmierung finden Sie in den jeweiligen Befehlshandbüchern.

3.5 Ansteuerungsarten

Der C-863 erlaubt den geregelten Betrieb der angeschlossenen Mechanik (= Servobetrieb eingeschaltet). Der geregelte Betrieb kann per Befehl aktiviert oder deaktiviert werden. Einzelheiten finden Sie in den jeweiligen Befehlshandbüchern.

Es können die folgenden Ansteuerungsarten über Befehle oder Parameter ausgewählt werden (abhängig vom Befehlssatz):

- **Positionsregelung**, Standardeinstellung nach dem Einschalten, siehe S. 24
- **Kraftregelung** (ab Firmware-Version 2.21), siehe S. 25
- **Positions- und Kraftregelung** (ab Firmware-Version 2.21), siehe S. 25

3.5.1 Positionsregelung

Die Positionsregelung ist nach dem Einschalten standardmäßig eingestellt. Die Regelungsvariable ist die Position des Verstellers. Das im Regelkreis verwendete Sensorsignal stammt vom inkrementellen Positionencoder des Verstellers.

Der Positionsregelkreis umfasst bestimmte Parameter (P-, I-, D-Term, I-Limit), die über Befehle oder Parameter konfiguriert werden können (abhängig vom Befehlssatz; Einzelheiten siehe die jeweiligen Befehlshandbücher).

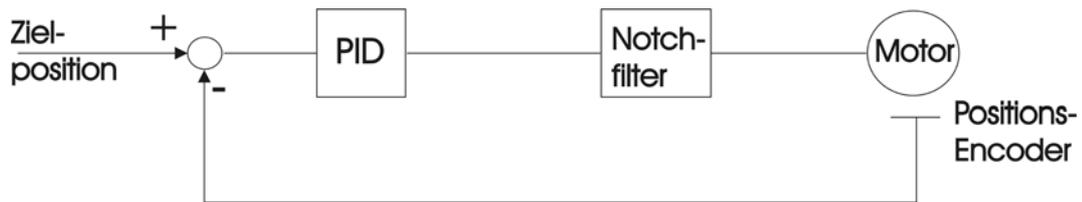


Abb. 9: Positionsregelungsalgorithmus, mit Sensorauswertung der Position (Positionencodier im Versteller) und PID-Korrektur

3.5.2 Kraftregelung

Ab der Firmware-Version 2.21 ist es möglich, die Kraft oder eine andere Variable zu regeln, die von einem zusätzlichen Sensor gemessen wird, der an die Buchse "Joystick" angeschlossen ist (z. B. ein Kraftsensor).

Diese Ansteuerungsart können Sie z. B. verwenden, um die vom Versteller auf einen Gegenstand ausgeübte Kraft gleichmäßig aufrechtzuerhalten. Nimmt die Kraft zu, dann fährt der Versteller zurück, und nimmt die Kraft ab, dann fährt der Versteller nach vorne, bis die tatsächliche Kraft wieder der Zielkraft entspricht.

Sensorsignalebereich: -10 V bis +10 V

Sensorsignaleingang: Buchse "Joystick", Pin 2

Auflösung: 12 Bit

Der Kraftregelkreis umfasst bestimmte Parameter (P-, I-, D-Term, I-Limit), die über Befehle oder Parameter konfiguriert werden können (abhängig vom Befehlssatz; Einzelheiten siehe die jeweiligen Befehlshandbücher).

Der Tiefpassfilter für den Sensoreingang kann je nach verwendetem Befehlssatz per Befehl oder Parametereingabe eingestellt werden. Der Frequenzbereich liegt zwischen 40 Hz und 10000 Hz.

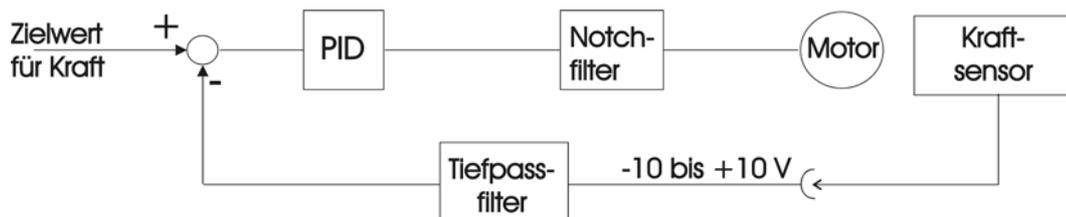


Abb. 10: Kraftregelungsalgorithmus mit PID-Korrektur und Notchfilter. Die Sensorsignale des zusätzlichen (Kraft-) Sensors werden durch einen Tiefpassfilter korrigiert.

3.5.3 Positions- und Kraftregelung

Ab der Firmware-Version 2.21 ist die Positionsregelung über einen integrierten (Kraft-) Regelkreis möglich, d. h. die Position und eine zusätzliche Variable werden geregelt. Die für die Positionsregelung verwendeten Sensorsignale stammen von einem inkrementellen Positionencodier in der Mechanik, während die Sensorsignale für den integrierten Regelkreis von einem zusätzlichen Sensor an der "Joystick"-Buchse stammen (z. B. ein Kraftsensor). Weitere Einzelheiten siehe die Abschnitte 3.5.1 und 3.5.2.

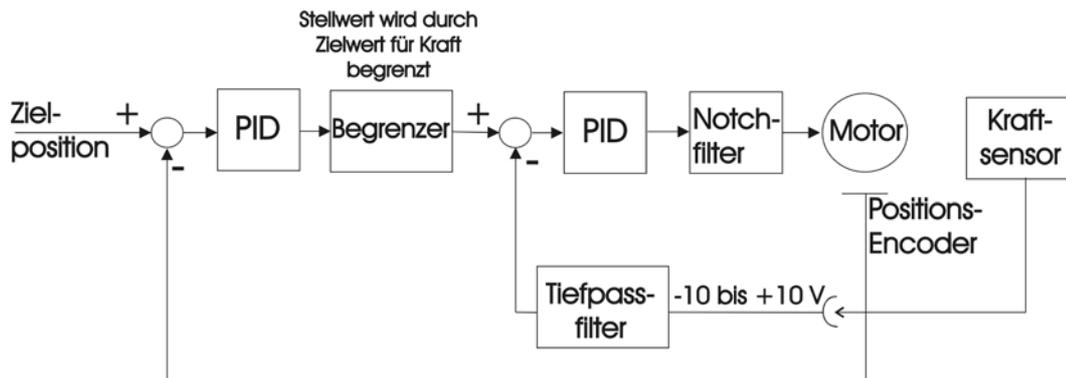


Abb. 11: Regelalgorithmus zur Aufrechterhaltung von Position und Kraft, mit Positionencoder und zusätzlichem (Kraft-) Sensor; zwei Regelkreise mit jeweils separater PID-Korrektur; der eingestellte Maximalwert der Kraft beschränkt den Stellwert aus dem Regelkreis für die Position.

3.6 Notchfilter

Mechanische Resonanzen des Systems führen bei bestimmten Steuersignalfrequenzen zu einem übersteigerten Ansprechverhalten. Der C-863 besitzt am Ende des Regelkreises einen Notchfilter, der Resonanzen in der Mechanik ausgleicht, indem er die entsprechenden Frequenzanteile im Steuersignal vermindert. Notchfilterfrequenz und -flanke können über Befehle oder Parameter eingestellt werden (abhängig vom Befehlssatz; Einzelheiten siehe die jeweiligen Befehlshandbücher). Der Frequenzbereich liegt zwischen 40 Hz und 10000 Hz.

3.7 Stellwegsbegrenzung

3.7.1 Endschalter

Während des Betriebs können Endsensoren (Endschalter) verwendet werden, um die Bewegung am Ende des zulässigen Stellweges anzuhalten und/oder absolute Positionsinformationen (Referenzierung) zu übermitteln; Einzelheiten zu Befehlen, die diese Signale verwenden, finden Sie in den jeweiligen Befehlshandbüchern. Jeder Endschalter unterbricht die Bewegung für eine bestimmte Richtung. Positioniersysteme von PI sind aufeinander abgestimmt und Endschalter und Sensoren bereits für den Betrieb mit dem Mercury-Controller eingestellt.

Der Mercury-Controller kann sowohl für "Active High"- als auch für "Active Low"- Stoppsignale der Endschalter konfiguriert werden; die Signallogik beider Schalter muss dabei gleich sein. Voreingestellt ist die Verwendung von "Active High"- Signalen.

Die Endschaltersignale werden sowohl von der Controller-Hardware als auch von der Software ausgewertet. Die Hardware-Einstellung kann über den DIP-Schalter 7 von Active High (OFF) zu Active Low (ON) geändert werden. Die softwareseitige Einstellung ist im Befehlshandbuch der verwendeten Software beschrieben (nativ: Lx-Befehle, GCS: SPA).

HINWEIS

Wenn die Konfiguration der Endschalter in Hard- und Software nicht übereinstimmt, kann keine Bewegung ausgeführt werden. DC-Motor-Versteller von PI sind mit "Active High"-Endschaltern ausgerüstet, so dass der DIP-Schalter 7 in der Stellung "OFF" sein muss.

3.7.2 Softwareseitige Verfahrbereichsgrenzen

Ab Firmware-Version 2.30 können in der Software Verfahrbereichsgrenzen (soft limits) eingestellt werden, die einen Sicherheitsabstand festlegen. Dieser Abstand darf vom Versteller an beiden Enden des Stellweges nicht unterschritten werden. An den Verfahrbereichsgrenzen wird die Bewegung der jeweiligen Achse unter Berücksichtigung der aktuellen Einstellungen für die Abbremsung sanft angehalten. Da die Endschalter keine Abbremsung berücksichtigen, besteht sonst die Gefahr, dass der Versteller mit voller Geschwindigkeit am mechanischen Anschlag aufprallt. Durch die Festlegung von Verfahrbereichsgrenzen kann dies verhindert und der Versteller vor Schäden geschützt werden.

Es hängt vom Befehlssatz ab, wie der Versteller bei der Kommandierung einer Position reagiert, die jenseits der softwareseitigen Verfahrbereichsgrenze liegt:

- Nativer Befehlssatz: Der Versteller bewegt sich und wird an der entsprechenden Verfahrbereichsgrenze sanft angehalten.
- GCS-Befehlssatz: Der Versteller bewegt sich nicht und ein entsprechender Fehlercode wird übermittelt.

Die softwareseitige Verfahrbereichsgrenze wird auch bei der joystickgesteuerten Bewegung angewendet.

Die softwareseitigen Verfahrbereichsgrenzen beziehen sich immer auf die aktuelle Nullposition. Daher werden bei einer Änderung der Nullposition auch die softwareseitigen Verfahrbereichsgrenzen verschoben.

Einzelheiten finden Sie in den entsprechenden Befehlshandbüchern.

3.8 Referenzsignal

Ein Referenzschaltersignal (auch Referenzsensor) kann zusammen mit einem Erfassungsalgorithmus des Motion-Prozessors verwendet werden, um absolute Positionierungsinformationen zu liefern (Referenzierung).

Der Mercury DC-Motor-Controller nimmt die Signale von einem richtungserkennenden Referenzschalter entgegen. Je nach dem, auf welcher Seite der Referenzposition die Achsenposition momentan liegt, ist der Signalpegel entweder hoch oder niedrig.

Einzelheiten zu Befehlen, die das Referenzsignal verwenden, finden Sie in den entsprechenden Befehlshandbüchern.

3.9 Ein- und Ausgangsleitungen

3.9.1 Überblick

An den C-863 Mercury DC-Motor Controllern stehen jeweils 4 digitale Ausgänge und 4 Eingänge für digitale oder analoge Signale zur Verfügung (8 Bit; Pinbelegung der I/O-Buchse siehe S. 48).

Eine Reihe von Befehlen steuern die Verarbeitung dieser Ein- und Ausgangssignale (Beschreibung von nativen Befehlen oder GCS-Befehlen siehe das jeweilige Befehlshandbuch).

Es gibt Befehle, die eine bedingte Ausführung erlauben, abhängig von den digitalen Eingängen. Dies ermöglicht z. B. die Steuerung über eine Pushbutton-Box oder einen Trackball (siehe die Seiten 37 und 38), die Kommunikation zwischen einzelnen Motorachsen im Stand-Alone-Betrieb (d. h. ohne Verbindung zum PC) oder die Steuerung externer Geräte über einen Trigger-Ausgang (siehe unten).

3.9.2 Trigger-Ausgang

Sie können den digitalen Signalausgang 4 des C-863 für die Ansteuerung sonstiger Geräte verwenden (Pin 8 der I/O-Buchse, siehe S. 48).

Der C-863 unterstützt die folgenden Trigger-Modi:

- Einfacher Triggerpuls: Ein Triggerpuls wird gesendet, wenn eine bestimmte Position auf der Achse erreicht ist.
- Mehrfache Triggerpulse: Der erste Triggerpuls wird gesendet, sobald eine bestimmte Position auf der Achse erreicht worden ist; danach wird jeweils ein Triggerpuls gesendet, wenn auf der Achse eine bestimmte Strecke zurückgelegt worden ist.

Die Auswahl des Trigger-Modus und die Einstellungen für die Triggerposition und -Strecke können per Befehl vorgenommen werden. Einzelheiten und Beispiele finden Sie in den entsprechenden Befehlshandbüchern.

4 Updates

4.1 Aktualisierung der Firmware

Die Firmware-Version Ihres Mercury-Controllers kann mit Hilfe folgender Befehle ausgelesen werden:

- Nativer Befehlssatz: VE
- GCS-Befehlssatz: VER?

Die Mercury-Firmware kann über die RS-232-Schnittstelle des PCs aktualisiert werden.

Der Aufbau des Controllers mit separatem Bootloader und eigener Firmware soll verhindern, dass das Gerät nach einem abgebrochenen Update blockiert ist. Es ist dennoch empfehlenswert, Firmware-Aktualisierungen sorgfältig durchzuführen und keine Experimente damit zu betreiben. Aufgrund von Problemen hinsichtlich der Zuverlässigkeit von bestimmter Hardware sollte die USB-Schnittstelle für die Aktualisierung der Firmware nicht verwendet werden.

HINWEIS

Auf dem Controller gespeicherte Makros werden bei der Aktualisierung der Firmware nicht gelöscht.

Um die Firmware zu aktualisieren, gehen Sie wie folgt vor:

1. Verbinden Sie den zu aktualisierenden Mercury-Controller mit dem PC über die RS-232-Schnittstelle (die Verwendung der USB-Schnittstelle wird nicht empfohlen).
2. Stellen Sie den DIP-Schalter 8 auf "ON" (Firmware-Update-Modus) und schalten Sie das Gerät aus und wieder ein; keine der beiden LEDs leuchtet.
3. Öffnen Sie im Windows-Explorer ein Festplattenverzeichnis, in dem sich sowohl das direkt ausführbare Programm *MMC_update* als auch die Image-Datei mit der neuen Firmware befinden (falls erforderlich, legen Sie ein solches Verzeichnis an und kopieren Sie die Dateien dort hinein).
4. Stellen Sie sicher, dass die Image-Datei mit der neuen Firmware so benannt ist, dass der Dateiname mit "C863" beginnt und die Datei die Endung ".hex" hat (benennen Sie die Datei um, falls erforderlich).
5. Starten Sie das Programm *MMC_update*.

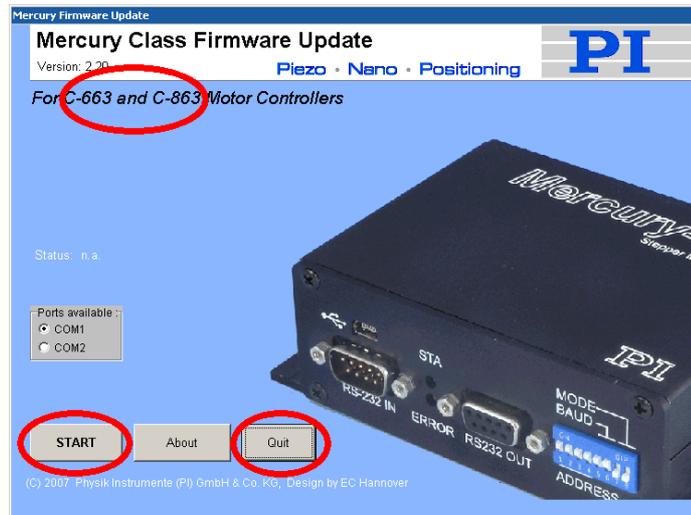


Abb. 12: MMC_update, Startbildschirm; unterstützte Controller-Modelle eingekreist.

6. Vergewissern Sie sich, dass Ihr Controller in der Überschrift genannt ist; der am Bildschirm dargestellte Controller kann davon abweichen (wie in der obigen Abbildung).
7. Stellen Sie sicher, dass der richtige COM-Port ausgewählt ist (die Verwendung des virtuellen COM-Ports über USB wird nicht empfohlen).
8. Klicken Sie auf *START*, um zum Hauptbildschirm zu gelangen.

HINWEIS

MMC_Update kann den exakten Typ des angeschlossenen Controllers nicht erkennen. Wenn Firmware für Mercury Step Controller in einen Mercury DC-Motor-Controller oder umgekehrt geladen wird, dann startet das System nicht. Starten Sie in solchen Fällen *MMC_update* neu mit der korrekten Firmware.

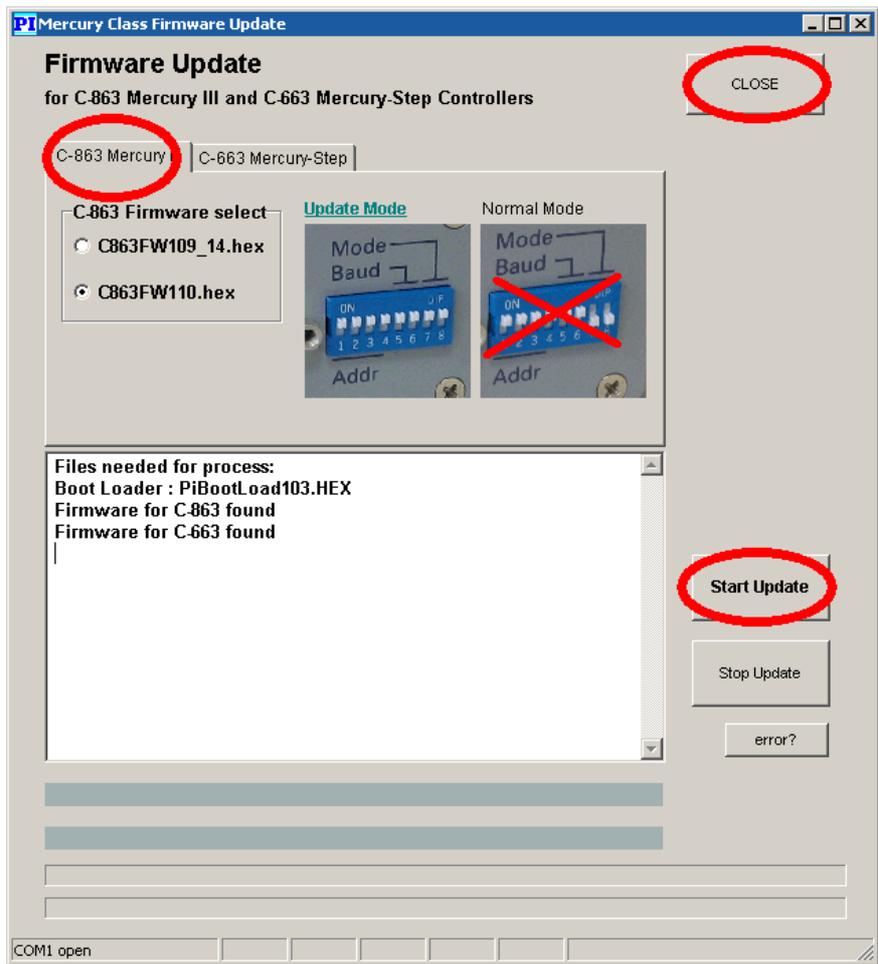


Abb. 13: MMC_update, Hauptbildschirm; Registerkarte C-863 und Schaltfläche Start Update eingekreist.

9. Stellen Sie sicher, dass die Registerkarte C-863 ausgewählt ist.
10. Stellen Sie sicher, dass im Bereich C-863 Firmware select die richtige Firmware ausgewählt ist. Es gibt dort eine Optionsschaltfläche für jede Datei, deren Name mit "C863" beginnt und auf ".hex" endet, und die sich in demselben Verzeichnis wie die ausführbare Datei MMC_update befindet.

HINWEIS

Die Verwendung der Schaltfläche Stop Update wird nicht empfohlen.

11. Klicken Sie einmal (und nicht mehrfach) auf Start Update. Die vier Balken über der Statuszeile liefern nun Informationen über den Fortschritt der Aktualisierung, und im Feld darüber erscheinen Protokollmeldungen (siehe Tabelle unten). Falls ein Fehler auftritt, schalten Sie den Controller aus und wieder ein und wiederholen Sie die Schritte 9-11.

12. Sobald die Aktualisierung abgeschlossen ist, kehren Sie zum Startbildschirm zurück, indem Sie auf *CLOSE* klicken; verlassen Sie dann das Programm mit *Quit*, um den COM-Port wieder freizugeben.

Meldungen

In der nachfolgenden Tabelle sind einige der Meldungen, die während der Aktualisierung erscheinen können, aufgeführt und erklärt.

Meldung	Mögliche Ursache	Maßnahmen
Can not detect baud rate / Verify switch 8 ON?	Firmware-Update-Modus nicht aktiviert	Stellen Sie den DIP-Schalter 8 auf "ON" und schalten Sie den Controller aus und wieder ein; keine der beiden LEDs sollte leuchten.
	RS-232-Kabel ist nicht oder am falschen Port angeschlossen	Kabel anschließen; Software neu starten und anderen Port wählen
	Falsches RS-232-Kabel mit falschem RS-232-Anschluss verbunden	Verbinden Sie den seriellen Port des PCs über ein Null-Modem-Kabel mit dem Anschluss "RS-232 In" des zu aktualisierenden Mercury-Controllers.
	Mercury nicht eingeschaltet	An die Spannungsversorgung anschließen
	Mehr als eine Instanz von <i>MMC_Update</i> läuft auf demselben COM-Port	Schließen Sie alle Instanzen und öffnen Sie dann eine einzige Instanz — beachten Sie, dass minimierte Instanzen sich möglicherweise an einer Ecke des Bildschirms angedockt haben.
	Ein anderes Programm verwendet den COM-Port.	Schließen Sie alle anderen Programme, die den COM-Port verwenden.
Can not connect bootloader	Sie haben mehrfach auf <i>Start Update</i> geklickt.	Klicken Sie nur ein einziges Mal auf <i>Start Update</i> .
Error while sending boot loader	Kommunikationsfehler	
Error: 0	Sie haben auf die Schaltfläche <i>error?</i> geklickt (Fehlercode 0 zeigt an, dass kein Fehler vorliegt, oder dass der Fehlerstatus bereits von der Software zurückgesetzt wurde).	Keine Maßnahme erforderlich
File xxx will be loaded Start_update: 0	Normaler Ablauf	
Copy boot loader	Normaler Ablauf	Nach diesem Zeitpunkt sollten Sie nur dann auf <i>Stop Update</i> klicken, wenn Sie feststellen, dass Sie Firmware für den falschen Gerätetyp aufspielen.
Connect Bootloader Erase Flash Send Data Block Update Complete	Normaler Ablauf	

4.2 Software-Aktualisierungen

Die neuesten Software-Versionen und Benutzerhandbücher stehen unter www.pi.ws zum Herunterladen zur Verfügung. Zum Herunterladen von Handbüchern und Software benötigen Sie ein Passwort. Dieses Passwort finden Sie auf der Mercury-CD in der PDF-Datei mit den Mercury-Release-News im Verzeichnis \Manuals.

Um die neueste Software (vollständige Kopie der CD) von der PI-Website herunterzuladen, gehen Sie wie folgt vor:

- 1 Stellen Sie auf der PI-Startseite (www.pi.ws) den Mauszeiger im Bereich *Service* am linken Seitenrand auf *Manuals, Software, ISO Statements*. Eine Auswahlliste erscheint.
- 2 Klicken Sie in der nun erscheinenden Liste auf *Software*.
- 3 Geben Sie im Bereich *User login* am linken Seitenrand den Benutzernamen (username) und das Passwort (password) aus der MercuryReleasenews xxxxx.pdf von der Mercury-CD ein, und klicken Sie auf *Login*.
- 4 Klicken Sie auf die Kategorie *C Motion Controllers*.
- 5 Klicken Sie auf *C-663*.
Hinweis: Bei der Auswahl des *C-863* wird ein Link zum *C-663* angezeigt, da die Mercury-CD auch für *C-663*-Modelle gilt.
- 6 Klicken Sie auf *Software* (wenn Sie auf *Documents* klicken, dann werden die neuesten Versionen der entsprechenden Handbücher angezeigt).
- 7 Klicken Sie unterhalb der neuesten CD-Kopie (CD-Mirror) auf die Schaltfläche *Download* (beinhaltet auch die Handbücher).

5 Manuelle Bedienhilfen

5.1 Joystick-Steuerung

Die manuelle Ansteuerung von C-863 Mercury DC-Motor Controllern ist bequem über einen analogen Joystick (Bestellnummern C-819.20, C-819.30) möglich, der an die "Joystick"-Buchse angeschlossen wird (Pinbelegung auf S. 49). Die Joystick-Position steuert hier direkt die Geschwindigkeit der kontrollierten Achse. Die Befehle für die Joystick-Steuerung (z. B. zu deren Aktivierung/Deaktivierung) sind in den jeweiligen Befehlshandbüchern beschrieben.

VORSICHT

Aktivieren Sie den Joystick-Eingang in der Software nur dann, wenn tatsächlich ein Joystick an den Controller angeschlossen ist. Andernfalls können unkontrollierte Bewegungen der angeschlossenen Achse auftreten, die die Anwendung gefährden können.

5.1.1 Zubehör zum Joystick

- C-819.20 Analoger Joystick, 2 Achsen
- C-819.20Y Y-Kabel zum Anschluss zweier Mercury-Controller an einen C-819.20-Joystick. Jeder Controller sieht nur eine Achse und die Einstellungen der entsprechenden Bedienelemente.
- C-819.30 Analoger Joystick für Mercury-Controller, 3 Achsen

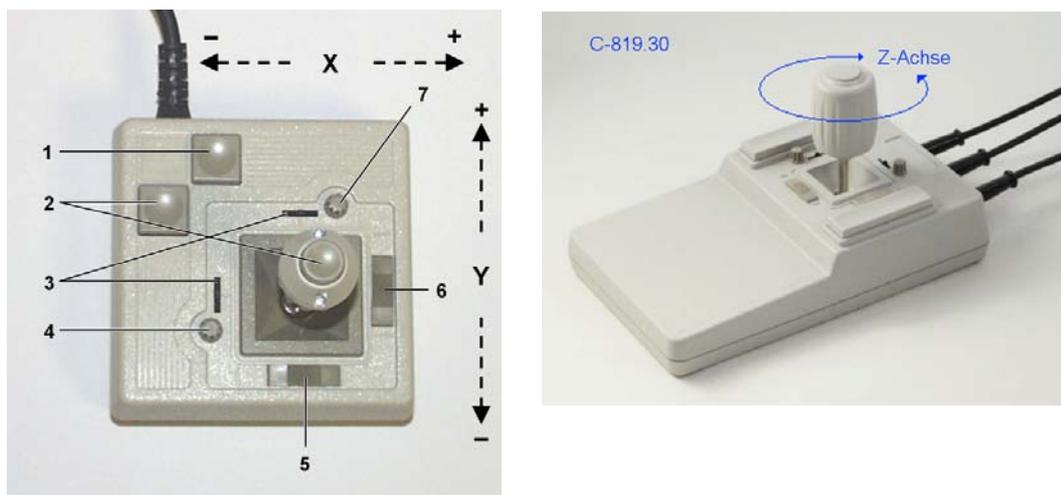


Abb. 14: Joystick C-819.20 (links) und Joystick C-819.30 (rechts) für Mercury-Controller

- 1 Drucktaste Funktion 1
- 2 Drucktaste Funktion 2
- 3 Justageanzeiger
- 4 Drehknopf für Justierung der Y-Achse
- 5 Arretierung der X-Achse
- 6 Arretierung der Y-Achse
- 7 Drehknopf für Justierung der X-Achse

HINWEISE

Während der C-819.30 mit separaten Anschlusskabeln für drei Controller ausgestattet ist, brauchen Sie für den C-819.20 ein Y-Kabel (C-819.20Y), um ihn an zwei verschiedene Mercury-Controller anzuschließen.

Die Spannungsversorgung des Joysticks erfolgt beim Anschluss eines Y-Kabels (C-819.20Y) über die Abzweigung für die X-Achse. Daher muss diese mit einem eingeschalteten Controller verbunden sein.

5.1.2 Joystick-Lookup-Tabellen

Normalerweise wird das Verhalten des Joysticks durch das Laden einer vordefinierten Lookup-Tabelle festgelegt.

Für spezielle Zwecke ist es möglich, eine benutzerdefinierte Joystick-Lookup-Tabelle zu erzeugen.

Einzelheiten finden Sie in den entsprechenden Befehlshandbüchern.

Eine Sonderanwendung siehe auch im nachfolgenden Abschnitt 5.1.3.

5.1.3 Kalibrierung der Z-Achse mit dem C-819.30-Joystick

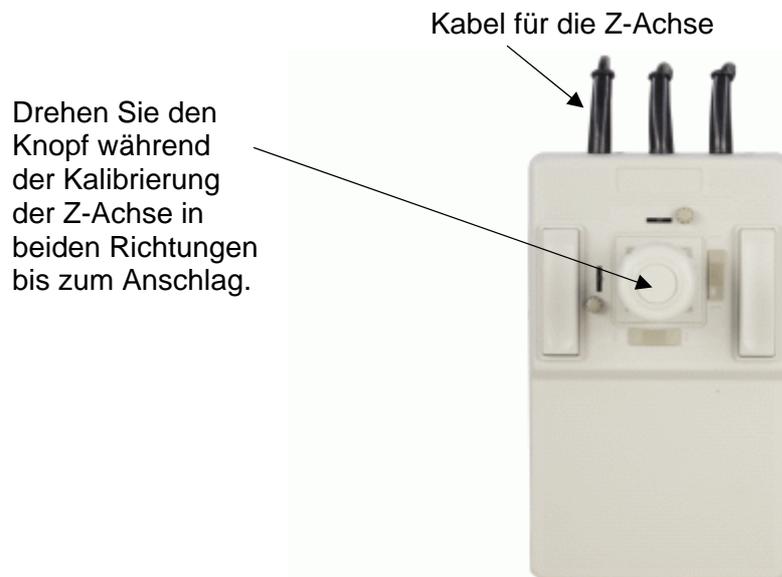
Der analoge, 3-achsige Joystick "C-819.30" kann über separate Anschlusskabel an bis zu drei C-863 oder C-663 Mercury Controller angeschlossen werden.

X- und Y-Achse des Joysticks arbeiten nach demselben Prinzip wie bei konventionellen 2-achsigen Geräten. Die Z-Achse des C-819.30 wird durch Drehen des Knopfes am Ende des Joysticks im Uhrzeiger- oder Gegenuhrzeigersinn betrieben.

Alle drei Joystick-Achsen sind durch eine Feder zentriert, so dass der Hebel beim Loslassen wieder in die Mittelstellung zurückkehrt. Die Mittelstellung sollte einer Geschwindigkeit von Null entsprechen, so dass sich die angeschlossenen Verstärker in der Ruhestellung befinden. Um sicherzustellen, dass die Mittelstellung einer Geschwindigkeit von Null entspricht, verfügt der C-819.30 über Einstellknöpfe für die X- und Y-Achse. Für die Z-Achse ist ein besonderes Kalibrierungsverfahren erforderlich, da die Z-Achse ein anderes Ansprechverhalten aufweist und über keinen Einstellknopf verfügt.

Für die Kalibrierung der Z-Achse wird das Software-Tool *MMC_Z-Joy* (*MMC_Z-Joy.exe*) verwendet. Für eine aktuelle Version von *MMC_Z-Joy* wenden Sie sich an Ihren PI-Vertriebsingenieur oder schreiben Sie uns eine E-Mail (info@pi.ws).

Auf der Grundlage des Joystick-Ausgangssignals für die Z-Achse erzeugt dieses Tool eine neue, kubische Lookup-Tabelle, damit der Versteller seine Höchstgeschwindigkeit erreichen kann und darüber hinaus eine hohe Empfindlichkeit bei niedrigen Geschwindigkeiten besitzt. Die neue Lookup-Tabelle bleibt so lange im Mercury-Controller gespeichert, bis sie durch eine neue Definition des Joystick-Verhaltens überschrieben wird.



Gehen Sie wie folgt vor:

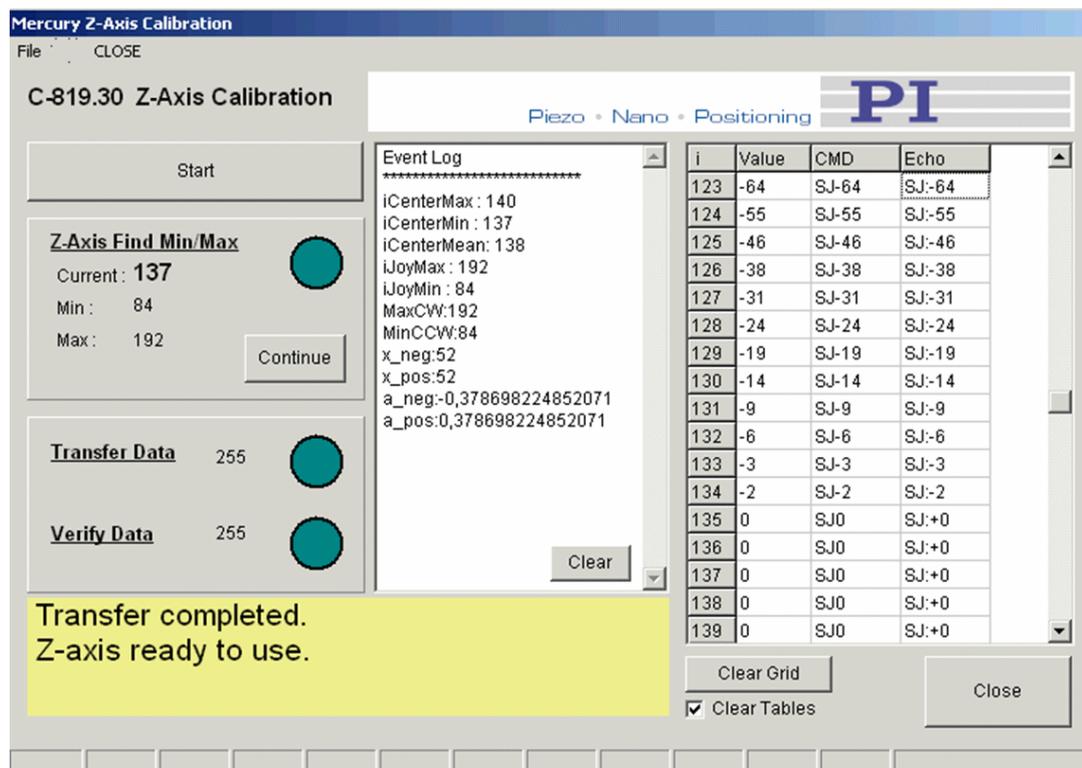
1. Verbinden Sie das Joystick-Kabel für die Z-Achse mit dem Mercury-Controller, der auf der Z-Achse mit dem Joystick betrieben werden soll.

Hinweise:

Das Kabel für die Z-Achse und der Controller sollten markiert werden, da sie nach der Kalibrierung zusammengehören. Die Änderung der Joystick-Achse bzw. der Austausch des Controllers machen eine erneute Kalibrierung notwendig.

Für die Kalibrierung ist es nicht notwendig, einen Versteller an den Controller anzuschließen.

2. Schließen Sie den Mercury-Controller an den PC an.
3. Schalten Sie den Mercury-Controller ein.
4. Starten Sie das Software-Tool *MMC_ZJoy* auf dem PC, indem Sie die entsprechende ausführbare Datei (*.exe) aufrufen.



5. Klicken Sie auf *Start*, um ins Hauptfenster zu gelangen. Dann klicken Sie erneut auf *Start*, um den Kalibriervorgang zu starten.
6. Wenn der Bereich *Z-Axis Find Min/Max* zu blinken beginnt, dann drehen Sie den Knopf für die Z-Achse in beiden Richtungen bis zum Anschlag.
7. Klicken Sie auf *Continue*. Nun wird die Lookup-Tabelle für den Joystick berechnet, übermittelt und überprüft. Der Vorgang ist abgeschlossen, wenn die Meldung *Z-axis ready to use* angezeigt wird.
8. Klicken Sie auf *Close*, um das Software-Tool zu verlassen.

Der Mercury-Controller wird nun auf den Betrieb mit der Z-Achse des Joysticks vorbereitet. Die korrigierte Lookup-Tabelle wird dauerhaft im Controller gespeichert. Wenn Sie den speziellen Abgleich rückgängig machen und den Controller mit einer "normalen" Joystick-Achse betreiben möchten, dann laden Sie eine geeignete, vordefinierte Lookup-Tabelle per Befehl. Einzelheiten finden Sie in den entsprechenden Befehlshandbüchern.

5.2 Steuerung über die Pushbutton-Box

Die Pushbutton-Box "C-170.PB" wird an die I/O-Buchse (Pinbelegung siehe S. 48) des Mercury-Controllers angeschlossen. Sie versorgt die Eingänge mit TTL-Signalen und zeigt den Status der Ausgänge anhand der LEDs.

Sie können die Pushbutton-Box dazu verwenden, einen Befehl oder ein Makro bedingt auszuführen, d. h. in Abhängigkeit davon, ob ein Knopf gedrückt wird oder

nicht. Einzelheiten zu Makros sowie Beispiele finden Sie in den entsprechenden Befehlshandbüchern.



Abb. 15: Pushbutton-Box "C-170.PB" mit LEDs für Mercury-Controller

5.3 Trackball-Steuerung

Ein Trackball-Gerät kann zur Vorgabe der Zielposition verwendet werden. Verbinden Sie die digitalen TTL-Signalausgänge A und B (auch Quadratur-Signale genannt) des Trackballs mit den digitalen Signaleingängen 3 und 4 der I/O-Buchse (Pinbelegung siehe S. 48) des Mercury-Controllers. Die Leitungen werden mit einem Widerstand von 10k auf Masse abgeschlossen.

Bei jeder Signalübertragung wird die Zielposition um einen bestimmten Wert in positiver oder negativer Richtung verschoben. Dies ermöglicht eine äußerst feine Positionierung mit hoher Auflösung.

Einzelheiten zur Trackball-Steuerung finden Sie in den entsprechenden Befehlshandbüchern.

6 Makros

Die Makrofunktionalität des Mercury-Controllers erlaubt es, Befehlsfolgen zu definieren und diese in einem nicht-flüchtigen Speicher im Controller abzulegen. Jedes zuvor definierte Makro kann durch einen Befehl aufgerufen und dann ausgeführt werden. Man kann ein Autostart-Makro definieren, das jedesmal nach dem Einschalten des Controllers ausgeführt wird, wodurch eine kundenindividuelle Steuerung auch ohne PC möglich ist.

Falls Sie die Anwendersoftware von PI verwenden (*PI MikroMove®* oder *MMCRun*), können Makros auch auf dem PC gespeichert werden (so genannte *Host Makros*). Details finden Sie im entsprechenden Softwarehandbuch.

Der Aufbau der Makros ist je nach Befehlssatz (nativer Befehlssatz oder GCS-Befehlssatz) unterschiedlich. Ein Makro besteht jedoch immer aus zulässigen Befehlen der jeweiligen Befehlssätze. Die GCS-Makrofunktionalität ist für Controller mit nativer Firmware über die GCS-DLL implementiert. Diese DLL übersetzt die Makros zwischen den beiden Befehlssätzen. Einzelheiten finden Sie in den entsprechenden Befehlshandbüchern.

Nachfolgend finden Sie einige Beispiele dafür, wie der Mercury-Controller vom Benutzer direkt ohne PC gesteuert werden kann, wenn z. B. ein Autostart-Makro und der Einsatz von Pushbutton-Box und/oder Joystick miteinander kombiniert werden:

- Ausführen von Befehlen, die Betriebsparameter setzen
- Ausführen von Bewegungsbefehlen
- Ausführen von Befehlen, die nach dem Ende der Bewegung wirksam werden
- Eine oder mehrere LEDs aufleuchten lassen (z. B. um einen bestimmten Status anzuzeigen)
- Bedingtes Ausführen eines Befehls oder Makros, abhängig davon, ob eine Taste der Pushbutton-Box oder am Joystick gedrückt wird
- Bedingtes Ausführen eines Befehls oder Makros, abhängig davon, wie bestimmte andere Befehle ausgeführt werden
- Bedingtes Ausführen eines Befehls oder Makros, abhängig von der Position der angeschlossenen Joystick-Achse
- Bedingtes Ausführen eines Befehls oder Makros, abhängig von der Ausgabe eines anderen Mercury oder externen Signals (nur bei zusätzlicher kundenspezifischer Verkabelung)

Generell kann mit Makros wie folgt gearbeitet werden:

- eigene Makros für verschiedene Initialisierungs- und Betriebsfunktionen verwenden
- von einem Makro aus ein weiteres aufrufen, eventuell bedingt
- die wiederholte Ausführung eines Makros aufrufen

- Ausgangssignale eines Mercury-Controllers mit den Eingängen eines weiteren verknüpfen

Einzelheiten finden Sie in den entsprechenden Befehlshandbüchern.

7 Störungsbehebung

HINWEIS

Der Controller muss aus- und wieder eingeschaltet werden, um neue Einstellungen der DIP-Schalter zu erkennen. Dabei werden interne Parameter wieder auf die ursprünglichen, werkseitigen Werte zurückgesetzt.

Problem (1):	Der Mercury-Controller ist eingeschaltet, aber keine der LEDs auf der Vorderwand leuchtet (im Normalbetrieb leuchtet die obere LED rot oder grün).
Lösung:	<p>Stellen Sie Folgendes sicher:</p> <ul style="list-style-type: none"> ■ DIP-Schalter #8 auf der Gerätevorderwand ist in unterer Position (OFF; Firmware-Update-Modus ausgeschaltet). ■ Ein Netzteil mit geeigneter Spannung ist angeschlossen und arbeitet ordnungsgemäß.
Problem (2):	Die Verstellerachse bewegt sich, bevor ein entsprechender Befehl gesendet wurde.
Lösung:	<ul style="list-style-type: none"> ■ Ist das Autostart-Makro aktiv? Lösung: Löschen Sie das Autostart-Makro. ■ Ist der Joystick ist zwar angeschlossen, aber nicht kalibriert? Vor der Benutzung eines Joysticks ist es erforderlich, das System zu kalibrieren. Wird der Joystick ohne Kalibrierung aktiviert, so kann eine Bewegung der Verstellerachse erfolgen, auch wenn sich der Joystick in der neutralen (mittleren) Position befindet. Um den Joystick zu kalibrieren, drehen Sie den Drehknopf für die entsprechende Achse so lange, bis der Motor anhält. <i>MMCRun</i> unterstützt Sie dabei, wenn Sie im Mercury Command Editor rechts oben auf die Registerkarte <i>Adjust</i> klicken. Einzelheiten finden Sie im <i>MMCRun</i>-Softwarehandbuch. Für die Kalibrierung der Z-Achse des Joysticks ist ein spezieller Kalibriervorgang erforderlich; Einzelheiten siehe S. 35. ■ Wurde der Joystick aktiviert, obwohl kein Joystick angeschlossen ist? Deaktivieren Sie den Joystick.

Problem (3):	Die PI-Anwendersoftware kann keine Verbindung mit dem Mercury aufbauen. Fehlermeldung "No controller found" oder "Could not connect..."
Lösung:	Überprüfen Sie bzw. stellen Sie Folgendes sicher: <ul style="list-style-type: none"> ■ Der COM-Port wird von keinem anderen Programm verwendet. ■ Die Baudrate ist richtig eingestellt (z. B. 9600 Baud: Die DIP-Schalter 5 und 6 stehen auf "ON"). Beachten Sie, dass <i>MMCRun</i> 9600 Baud benötigt. ■ Leuchtet die obere LED (STA) rot oder grün? Falls nein, siehe Problem (1). ■ Die Firmware hat die aktuelle Version. ■ Bei RS-232-Verbindung: Nullmodemkabel C-815.34 ist angeschlossen. ■ Entweder das RS-232-Kabel oder das USB-Kabel ist angeschlossen, aber nicht beide Kabel.
Problem (4):	Das System kann nicht mit einem Kunden-Terminalprogramm angesprochen werden.
Lösung:	Überprüfen Sie bzw. stellen Sie Folgendes sicher: <ul style="list-style-type: none"> ■ Ist die Kommunikation möglich, wenn die Software von PI verwendet wird? ■ Der Controller wurde über die richtige Adressierung ausgewählt, bzw. das Autostart-Makro enthält einen SC-Befehl mit der richtigen Adresse. Die Geräteadresse wird mit den DIP-Schaltern 1 bis 4 auf der Gerätevorderwand eingestellt. Einzelheiten zur Adressierung finden Sie im Benutzerhandbuch für den nativen Befehlssatz. ■ Die Baudrate ist am Controller und in der Software gleich.

Problem (5):	Der Controller kommuniziert und gibt Positionswerte aus, aber der angeschlossene Versteller oder Motor bewegt sich nicht.
Lösung:	<ul style="list-style-type: none"> ■ Stellen Sie sicher, dass die Motorbremse ausgeschaltet ist (OFF). Einige Versteller ohne Bremse sind so eingestellt, dass sich der Motor nicht bewegt, wenn das Bremssignal den Wert ON hat. ■ Sind die Parameter, die die Signallogik der Endschalter definieren und das Vorhandensein von Endschaltern anzeigen, softwareseitig richtig eingestellt (nativer Befehlssatz: Lx; GCS-Befehlssatz: SPA)? ■ Überprüfen Sie, ob die Signallogik der Endschalter am Controller richtig eingestellt ist (DIP-Schalter 7 steht auf "OFF" für "Active High", für Standard-DC-Motorantriebe von PI). ■ Wenn Sie einen Versteller mit integriertem PWM-Verstärker (.PD-Versteller) verwenden, stellen Sie sicher, dass das 24-V-DC-Netzteil mit dem Versteller verbunden ist und ordnungsgemäß arbeitet. ■ Der Controller könnte im Joystick- oder Trackball-Modus sein, so dass er keine Bewegungsbefehle annimmt. Um die Bewegungssteuerung durch Befehle zu aktivieren, deaktivieren Sie den Joystick- oder Trackball-Modus (nativer Befehlssatz: JF; GCS-Befehlssatz: JON oder SPA).

Problem (6)	Befehle werden nicht erkannt.
Lösung:	<ul style="list-style-type: none"> ■ <i>MMCRun</i>: Verwenden Sie den Befehlsmodus "straight" anstelle von "protected". Möglicherweise wurde der Befehl neu implementiert und die verwendete Programmversion erkennt ihn nicht. ■ Stellen Sie sicher, dass alle Befehle, die aus mehreren Buchstaben bestehen, durch eine gültige Endmarke beendet werden (nativer Befehlssatz: CR; GCS-Befehlssatz: LF).
Problem (7)	Der Controller scheint nicht richtig auf die Endschalter-Signale des Verstellers zu reagieren.
Lösung:	<ul style="list-style-type: none"> ■ Vergewissern Sie sich, dass die Parameter, die die Signallogik der Endschalter definieren und das Vorhandensein von Endschaltern anzeigen, softwareseitig richtig eingestellt sind (nativer Befehlssatz: Lx-Befehle; GCS-Befehlssatz: SPA). ■ Bei Verwendung von Nicht-PI-Komponenten: Das Endschalersignal des Verstellers ist nicht kompatibel mit den Eingangsleitungen des Mercury-Controllers (z. B. ungenügender Sink/Source-Strom). Wenden Sie sich an PI und an den Hersteller des Verstellers, um das Problem zu lokalisieren. ■ Überprüfen Sie, ob die Signallogik der Endschalter am Controller richtig eingestellt ist (DIP-Schalter 7 steht auf "OFF" für "Active High", für Standard-DC-Motorantriebe von PI).
Problem (8)	Der Versteller reagiert nicht auf Joystick-Bewegungen.
Lösung:	<p>Stellen Sie sicher, dass der Joystick richtig angeschlossen und aktiviert ist.</p> <p>Möglicherweise ist es nötig, den Joystick zu kalibrieren; Einzelheiten siehe "Joystick-Steuerung" auf S. 34.</p>

8 Technische Daten

8.1 Spezifikationen

	C-863.10
Funktion	DC-Motorsteuerung, servogeregelt, 1 Kanal
Antriebsart	DC-Motor oder Voice-Coil-Antrieb
Kanäle	1
Bewegung und Regler	
Reglertyp	P-I-D-Servoregler
Trajektorienprofile	Trapez, Punkt-zu-Punkt
Encodereingang	A/B-Quadratur-TTL-Pegel, differenziell gem. RS-422; 20 MHz
Blockiererkennung	Automatischer Motorstopp bei Überschreitung eines programmierbaren Positionsfehlers
Eingang Endschalter	2 x TTL-Signale, programmierbare Signallogik, Hardware Sperre
Eingang Referenzschalter	1 x TTL-Signal
Ausgang Motorbremse	1 x TTL-Signal, programmierbar
Elektrische Eigenschaften	
Ausgangsleistung	3 A x Versorgungsspannung bzw. begrenzt durch Netzteil (30 W mit Standardnetzteil 15 V, 2 A C-890.PS)
Ausgangsspannung	0 bis \pm Versorgungsspannung (15 V mit Standardnetzteil C-890.PS); der verwendete Motor muss für die Versorgungsspannung geeignet sein.
Schnittstellen und Bedienung	
Schnittstellen / Kommunikation	RS-232 (Busarchitektur), USB 1.1
Motoranschluss	D-Sub 15 (f)
Controller-Netzwerk	Bis zu 16 Einheiten an einer Schnittstelle *
Ein-/Ausgänge (I/O-Leitungen)	4 analoge/digitale Eingänge, 4 digitale Ausgänge
Befehlssatz	Nativer (Mercury-) Befehlssatz, GCS (über DLL)
Anwendersoftware	MMCRun, PIMikroMove®

*Der RS-232 Ausgang mancher PCs unterstützt maximal 6 Einheiten; falls dies ein Problem darstellt, verwenden Sie die USB-Schnittstelle für die Verbindung zum PC.

Software-Treiber	Mercury-GCS-DLL (PI General Command Set), GCS-LabVIEW-Treiber, native Mercury-DLL, native LabVIEW-Treiber
Manuelle Bedienungshilfe (optional)	Joystick, Pushbutton-Box, Trackball
Umgebung	
Betriebsspannung	15 bis 30 VDC, 15 V empfohlen; bei höherer Spannung muss der angeschlossene Motor hierfür geeignet sein.
Stromaufnahme	80 mA + Motorstrom (max. 3 A)
Betriebstemperaturbereich	5 bis 50 °C
Masse	0,3 kg
Abmessungen	130 x 76 x 40 mm

8.2 Abmessungen

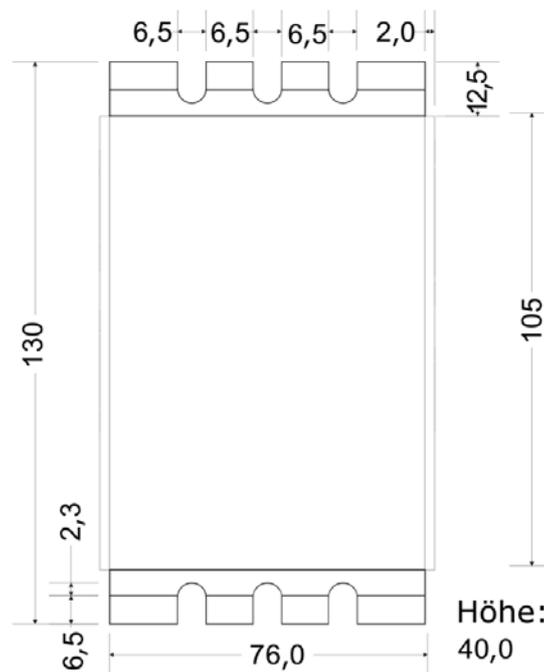


Abb. 16: C-863 Abmessungen in Millimetern

8.3 Pinbelegungen

8.3.1 Motoranschluss

Die Signale zur Motorsteuerung befinden sich alle auf dem 15-poligen D-Sub-Anschluss "DC Motor only"; sie umfassen die Motorsteuersignale (Ausgänge) sowie die Encoder- und Endschaltersignale (Eingänge). Dieser Anschluss ist mit allen PI-Verstellern mit DC-Motoren und den aktuellen Voice-Coil-Antrieben kompatibel. Versteller mit abnehmbarem Kabel werden mit dem Motorkabel C-815.38 ausgeliefert.

VORSICHT

Schrittmotorantriebe verwenden denselben Anschluss wie DC-Motorversteller, sind jedoch nicht mit ihnen kompatibel. Es kann ein dauerhafter Schaden entstehen, wenn Versteller und Controller nicht zueinander passen.

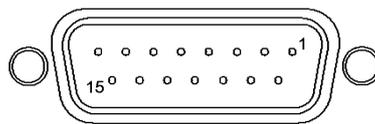


Abb. 17: Motorbuchse, D-Sub, 15-polig (f)

Pin	Pin	Funktion
1		Programmierbarer Ausgang für Motorbremse (0 oder + 5 V)
	9	Motor – (Power PWM*-Ausgang für Motor; – differentiell)
2		Motor + (Power PWM*-Ausgang für Motor; + differentiell)
	10	Power GND
3		PWM Magnitude (Ausgang; TTL)
	11	PWM Sign (Ausgang für Bewegungsrichtung; TTL)
4		+5 V (Ausgang)
	12	Negativer Endschalter (Eingang)
5		Positiver Endschalter (Eingang)
	13	Referenzschalter (Eingang)
6		GND Endschalter
	14	Encoder: A(+) / ENCA (Eingang)
7		Encoder: A(-) (Eingang)
	15	Encoder: B (+) / ENCB (Eingang)
8		Encoder: B (-) (Eingang)

* Bei der Pulsweitenmodulation (PWM) ist der HIGH-Pegel auf diesen Leitungen etwa so hoch wie die Versorgungsspannung.

VORSICHT

Die Ausgangsspannung an den Anschlüssen "Motor +" und "Motor -" kann so hoch wie die Versorgungsspannung sein. Falls Sie kein Standardnetzteil für den Mercury-Controller verwenden und ein Motor an diese Leitungen angeschlossen ist (d. h. Anschluss ohne separaten PWM-Verstärker), dann stellen Sie sicher, dass die Betriebsspannung des Motors nicht überschritten wird.



8.3.2 I/O-Anschluss

Pin	Signal- richtung	Funktion
1	Eingang	Eingang 1 (analog: 0 bis +5V/ digital: TTL)
2	Eingang	Eingang 2 (analog: 0 bis +5V/ digital: TTL)
3	Eingang	Eingang 3 (analog: 0 bis +5V/ digital: TTL)
4	Eingang	Eingang 4 (analog: 0 bis +5V/ digital: TTL)
5	Ausgang	Ausgang 1 (digital: TTL)
6	Ausgang	Ausgang 2 (digital: TTL)
7	Ausgang	Ausgang 3 (digital: TTL)
8	Ausgang	Ausgang 4 (digital: TTL)
9	Ausgang	Vcc (+5 V)

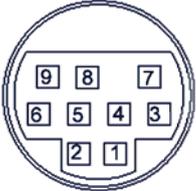


Abb. 18: I/O-Anschluss, Mini-DIN, 9-polig

8.3.3 Joystick-Anschluss

Pin	Signal- richtung	Funktion
1		GND
2		nicht verbunden
3	Ausgang	Vcc (3,3 V)
4	Eingang	Eingang: analoger Joystick 0 bis 3,3 V
5		nicht verbunden
6	Eingang	Eingang: Joystick-Taste #1



Abb. 19: Joystick-Anschluss, Mini-DIN, 6-polig (PS2)

8.3.4 Y-Kabel für Joystick

Das Y-Kabel für zwei Joystick-Achsen (C-819.20Y) bildet die Signale für Position und Tasten der zweiten Joystick-Achse (Y) auf die Joystick-Eingänge des zweiten Controllers folgendermaßen ab:

Joystick- Pin	Signal	Controller 1 (X) Pin	Controller 2 (Y) Pin
1	GND	1	1
2	Taste 2 (0 oder 3,3 V)		6
3	Vcc (3,3 V Eingang)	3 (3,3 V Ausgang)	nicht verbunden
4	X-Achsen-Pos., 0 bis 3,3 V	4	
5	Y-Achsen-Pos., 0 bis 3,3 V		4
6	Taste 1 (0 oder 3,3 V)	6	

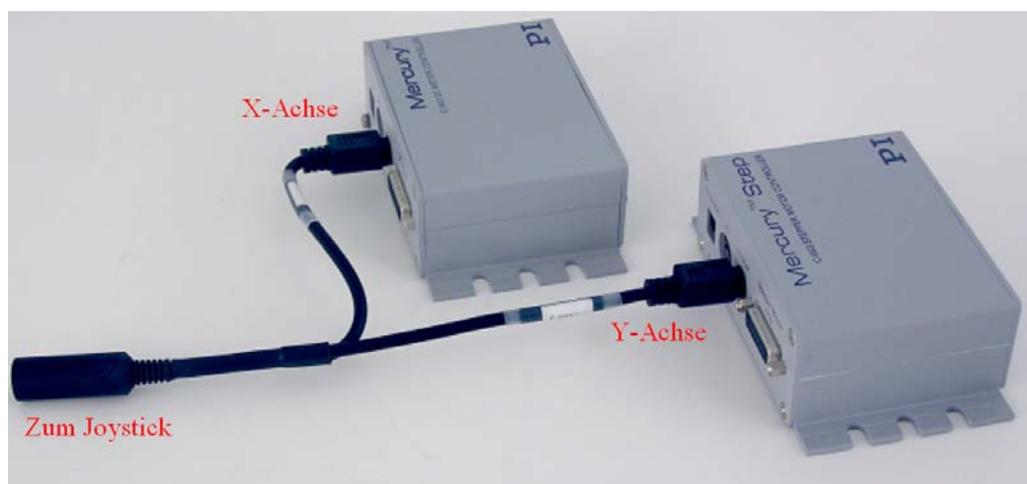


Abb. 20: Y-Kabel (C-819.20Y) für Joystick mit 2 Controllern

8.3.5 RS-232-Anschlüsse

Bezeichnung der Anschlüsse: RS-232 In und RS-232 Out

Anschlussypen: D-Sub, 9-polig, (m) für "Out", (f) für "In"

Die In- und Out-Leitungen sind dauerhaft über Datenbusleitungen verbunden. Ausnahme: Bei aktivierter USB-Schnittstelle bestätigt der Controller Signale auf der RxD-Leitung (Empfangsleitung), Pin 2, die ansonsten nur der Kommunikation mit dem PC dient. Aus diesem Grund kann ein Netzwerk, das nur über RS-232 gesteuert wird, eventuell nur 6 Controller beinhalten.

Hinweis: Der RS-232-Bus am Controller ist als DTE verkabelt und erfordert ein spezielles Nullmodemkabel zum Anschluss an einen PC.

Anschlüsse RS-232 In und RS-232 Out

Pin-Nummer an allen Mercury- Anschlüssen	Name des Signals auf den Mercury- Anschlüssen	Signalrichtung	Funktion
1			nicht verbunden
2	RxD*	PC zum Controller**	Befehlsübertragung
3	TxD*	Controller** zum PC	Report (Antwortübertragung)
4			nicht verbunden
5	GND		GND
6			nicht verbunden
7			nicht verbunden
8			nicht verbunden
9			nicht verbunden

*Die RS-232-Verbindung mit dem PC erfolgt über ein Nullmodem-Kabel. Dadurch sind die Bezeichnungen der Signale auf PC-Seite umgekehrt.

**Wenn der PC über USB angeschlossen wird, dann leitet der angeschlossene Mercury-Controller alle PC-Signale auf die RxD-Leitung *beider* RS-232-Anschlüsse. Alle auf der TxD-Leitung vorhandenen Daten überträgt der Mercury über USB an den PC.

VORSICHT

Verbinden Sie niemals gleichzeitig den USB- und den RS-232-Anschluss eines Controllers mit Ihrem PC. Dies kann Schäden verursachen.





Abb. 21: Mini-USB-Anschluss, Typ B (oben) und RS-232-Anschluss, D-Sub, 9-polig (unten)

8.3.6 USB-Anschluss

Handelsüblicher USB-Minstecker Typ B

8.3.7 Spannungsversorgungsanschluss und Erdungsschraube

Die Anschlüsse für Spannungsversorgung und Erdung bzw. Schutzleiter befinden sich an der Rückwand des Controllers:

Hohlstecker: Mittelstift	Eingang	15 bis 30 V DC	
Hohlstecker: Außenleiter	Eingang	GND (Power)	
Erdungsschraube (unten links)	-	Erdung/ Schutzleiter	

VORSICHT

Bei fehlendem Schutzleiter können gefährliche Berührungsspannungen entstehen, und es besteht Stromschlagsgefahr. Im Falle eines Fehlers oder Defekts kann das Berühren des Controllers zu schweren Verletzungen oder zum Tode führen.

Stellen Sie sicher, dass das Metallgehäuse über die Erdungsschraube unten links an der Geräterückwand geerdet bzw. an einen Schutzleiter angeschlossen ist.



9 Entsorgung

Gemäß EU-Richtlinie 2002 / 96 / EG (WEEE) dürfen Elektrogeräte ab dem 13. August 2005 in den Mitgliedsstaaten der EU nicht mehr über den kommunalen Restmüll entsorgt werden.

Um der Produktverantwortung als Hersteller gerecht zu werden, übernimmt die Physik Instrumente (PI) GmbH & Co. KG kostenfrei die umweltgerechte Entsorgung eines PI-Altgerätes, sofern es nach dem 13. August 2005 in Verkehr gebracht wurde.

Falls Sie ein solches Altgerät von PI besitzen, können Sie es versandkostenfrei an folgende Adresse senden:

Physik Instrumente (PI) GmbH & Co. KG
Auf der Römerstr. 1
76228 Karlsruhe

