



# Befehlsreferenz

## Beschreibung der seriellen Befehle der Nanotec-Firmware

NANOTEC ELECTRONIC GmbH & Co. KG  
Gewerbestraße 11  
D-85652 Landsham bei München

Tel. +49 (0)89-900 686-0  
Fax +49 (0)89-900 686-50  
[info@nanotec.de](mailto:info@nanotec.de)

---

## Impressum / Zu diesem Handbuch

© 2008

**Nanotec<sup>®</sup> Electronic GmbH & Co. KG**

Gewerbestraße 11

D-85652 Landsham / Pliening

Tel.: +49 (0)89-900 686-0

Fax: +49 (0)89-900 686-50

Internet: [www.nanotec.com](http://www.nanotec.com)

Alle Rechte vorbehalten!

MS-Windows 98/NT/ME/2000/XP sind eingetragene Warenzeichen der Microsoft Corporation.

### Zielgruppe

Dieses Dokument richtet sich an Programmierer, die eine eigene Steuerungssoftware für die Kommunikation mit den Steuerungen für folgende Nanotec Motoren programmieren wollen:

- Nanotec SMC I Schrittmotoren
- Nanotec Plug & Drive Motoren

### Zu diesem Handbuch

Vor der Verwendung der Befehlsreferenzen der Nanotec Firmware zur Erstellung eigener Steuerungsprogramme ist dieses Technische Handbuch sorgfältig durchzulesen.

Nanotec<sup>®</sup> behält sich im Interesse seiner Kunden das Recht vor, technische Änderungen und Weiterentwicklungen von Hard- und Software zur Verbesserung der Funktionalität dieses Produktes ohne besondere Ankündigung vorzunehmen.

Dieses Handbuch wurde mit der gebotenen Sorgfalt zusammengestellt. Es dient ausschließlich der technischen Beschreibung der Befehlsreferenzen der Nanotec Firmware. Die Gewährleistung erstreckt sich gemäß unseren allgemeinen Geschäftsbedingungen ausschließlich auf Reparatur oder Umtausch defekter Geräte der Nanotec Schrittmotoren, eine Haftung für Schäden und Fehler durch fehlerhafte Verwendung der Befehlsreferenzen in der Programmierung für eigene Motorensteuerungen ist ausgeschlossen.

Für Kritik, Anregungen und Verbesserungsvorschläge wenden Sie sich bitte an die oben angegebene Adresse oder per Email an: [info@nanotec.de](mailto:info@nanotec.de)

### Version/Änderungsübersicht

Version	Datum	Änderungen
V1.0	30.06.2008	Neuerstellung

---

# Inhalt

<b>Impressum / Zu diesem Handbuch</b> .....	<b>2</b>
<b>Inhalt</b> .....	<b>3</b>
<b>1 Allgemeine Informationen</b> .....	<b>5</b>
1.1 Aufbau eines Befehls .....	5
1.2 Hinweis zum alten Kommandierungsformat .....	5
1.3 Entwicklungsunterstützung .....	6
<b>2 Lesebefehl</b> .....	<b>10</b>
<b>3 Sätze</b> .....	<b>11</b>
<b>4 Allgemeine Befehle</b> .....	<b>12</b>
4.1 Bootloader starten.....	12
4.2 Totbereich Joystickmodus .....	12
4.3 Fehlerspeicher auslesen.....	13
4.4 Drehgeberposition auslesen .....	13
4.5 Fehlerkorrekturmodus einstellen.....	14
4.6 Filter für Analog- und Joystickmodus einstellen .....	14
4.7 Schrittwinkel einstellen.....	15
4.8 Firmwareversion auslesen .....	15
4.9 Eingänge maskieren .....	16
4.10 Eingänge und Ausgänge auslesen .....	17
4.11 Firmwareversion auslesen (alt).....	17
4.12 Status auslesen.....	18
4.13 Motoradresse auslesen.....	18
4.14 Parameter auslesen .....	19
4.15 Position auslesen .....	19
4.16 Motoradresse einstellen .....	20
4.17 Schrittmodus einstellen .....	20
4.18 Phasenstrom einstellen.....	21
4.19 Phasenstrom im Stillstand einstellen .....	21
4.20 Motormodus einstellen.....	22
4.21 Endschalterverhalten einstellen .....	23
4.22 Endschaltertyp einstellen .....	24
4.23 Satz speichern .....	25
4.24 Umkehrspiel einstellen.....	26
4.25 Position zurückstellen .....	26
4.26 Automatisches Senden des Status einstellen.....	27
4.27 Satz für Autokorrektur einstellen.....	27
4.28 Encoderrichtung einstellen.....	28
4.29 Ausschwingzeit einstellen .....	28
4.30 Maximale Abweichung Drehgeber einstellen.....	29

**Inhalt**

---

4.31	Minimalspannung für Analogmodus einstellen .....	29
4.32	Maximalspannung für Analogmodus einstellen .....	30
4.33	Totbereich für Joystickmodus einstellen .....	30
4.34	Satz starten .....	30
4.35	Satz stoppen .....	31
4.36	Drehzahl erhöhen.....	31
4.37	Drehzahl verringern.....	31
4.38	Trigger auslösen .....	32
4.39	Positionsfehler zurücksetzen .....	32
4.40	Satz aus EEPROM laden.....	32
4.41	Aktuellen Satz auslesen.....	33
<b>5</b>	<b>Satzbefehle .....</b>	<b>34</b>
5.1	Positionierart setzen.....	34
5.2	Verfahrweg einstellen.....	36
5.3	Minimalfrequenz einstellen .....	36
5.4	Maximalfrequenz einstellen .....	37
5.5	Maximalfrequenz 2 einstellen .....	37
5.6	Rampe einstellen .....	38
5.7	Richtung einstellen.....	38
5.8	Richtungsumkehr einstellen.....	39
5.9	Wiederholungen einstellen.....	39
5.10	Satzpause einstellen .....	40
5.11	Folgesatz einstellen .....	40
<b>Index</b>	.....	<b>41</b>

# 1 Allgemeine Informationen

## 1.1 Aufbau eines Befehls

### Aufbau von Steuerungsbefehlen

Ein Befehl beginnt immer mit dem Startzeichen '#' und endet mit einem Carriage-Return '\r'. Alle dazwischenliegenden Zeichen sind ASCII-Zeichen (also keine Steuerzeichen).

Nach dem Startzeichen folgt zuerst die Adresse des Motors als ASCII-Dezimalzahl. Dieser Wert darf von 1 bis 254 betragen. Wird ein '\*' anstatt der Zahl gesendet, werden alle am Bus hängende Steuerungen angesprochen.

Darauf folgt der eigentliche Befehl, der im Allgemeinen aus einem ASCII-Zeichen und einer optionalen ASCII-Zahl besteht. Diese Zahl ist in der Dezimaldarstellung mit einem führenden Vorzeichen (,+ ' und ,-') zu schreiben.

Sendet der Nutzer eine Einstellung an die Firmware, ist bei positiven Zahlen das ,+'-Zeichen nicht zwingend erforderlich.

Hinweis: Manche Befehle bestehen aus mehreren Zeichen und andere wiederum benötigen keine Zahl als Parameter.

### Reaktion der Steuerung

Hat eine Steuerung einen Befehl als für sich gültig erkannt, sendet sie als Bestätigung den Befehl als Echo, aber ohne das Startzeichen '#' zurück.

Hat die Steuerung einen unbekanntem Befehl empfangen, antwortet diese mit einem dem Befehl nachgestellten Fragezeichen '?'.

Die Antwort der Steuerung wird wie der Befehl selbst mit einem Carriage-Return '\r' abgeschlossen.

### Beispiele

Setzen des Fahrweges von Steuerung 1: „#1s1000\r“ -> „1s1000\r“

Starten eines Satzes: „#1A\r“ -> „1A\r“

Ungültiger Befehl: „#1\r“ -> „1/?\r“

## 1.2 Hinweis zum alten Kommandierungsformat

### Abwärtskompatibilität

Um abwärtskompatibel zu alten Steuerungen der Firma Nanotec zu bleiben, wird das alte Kommandierungsformat auch in der aktuellen Version der Firmware (09-11-2007) unterstützt.

Bei dem alten Kommandierungsformat wird bei Zahlen, die aus mehreren Bytes bestehen, jedes Byte einzeln in eine Dezimalzahl (0 bis 255) gewandelt und dann übertragen (z.B. 255 => 255 ; 256 => 1000 ; 257 => 1001).

Dieses Art der Umrechnung findet bei **KEINEM** der hier beschriebenen Befehle Anwendung. Alle hier beschriebenen Befehle verwenden normale Dezimalzahlen.

### Beschreibung des alten Kommandierungsformates

Die Beschreibung des alten Kommandierungsformates ist nicht Gegenstand dieser Dokumentation. Altes Format siehe Handbuch Plug & Drive Serie PDx-I (Version 2.1).

## 1.3 Entwicklungsunterstützung

### Überblick

Im folgenden Manual werden die Befehle für die Kommunikation mit den Nanotec-Steuerungen über die serielle bzw. die USB Schnittstelle beschrieben. Dies bietet Ihnen die Möglichkeit, mittels jeder beliebigen Programmiersprache und von jedem dazu geeigneten programmierfähigen Gerät unsere Steuerungen anzusprechen.

In diesem Abschnitt werden folgende Punkte kurz erläutert:

- DLL-Bibliothek
- Anwendungsbeispiel der DLL-Bibliothek
- Windows-Hilfe zur DLL-Bibliothek

### DLL-Bibliothek

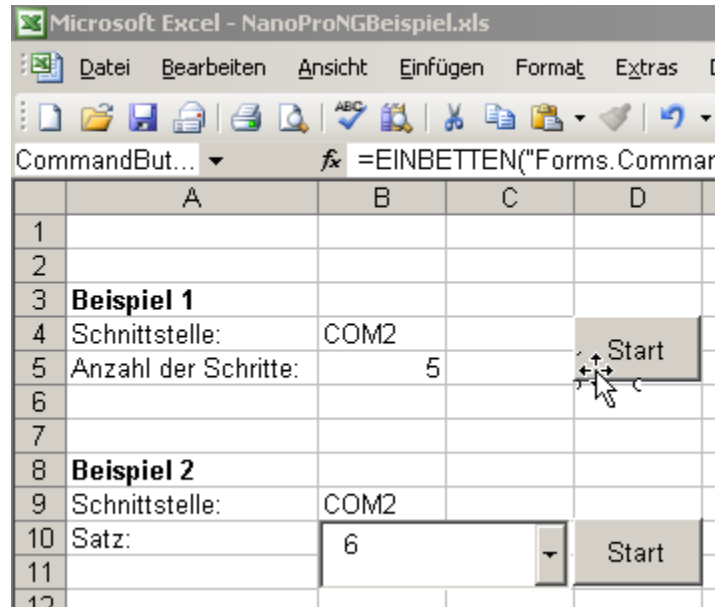
Sollten Sie unter Windows mit einer .NET-fähigen Programmiersprache (z.B. Visual Basic ab Version 8, C#, Delphi.NET) arbeiten, bieten wir Ihnen die komfortablere Möglichkeit, unsere DLL-Bibliothek in Ihre Applikation einzubinden. Sie können dann die Funktionen der Steuerung bequem über einen Funktionsaufruf ansteuern, ohne sich um die Details der Kommunikation kümmern zu müssen.

Die für Sie zur Entwicklungsunterstützung mitgelieferte Dynamic Link Library (DLL) ermöglicht Ihnen also den so zur Verfügung gestellten Befehlsatz schnell, komfortabel und fehlerfrei in Ihre individuellen Anwendungen zur Motorensteuerung zu integrieren.

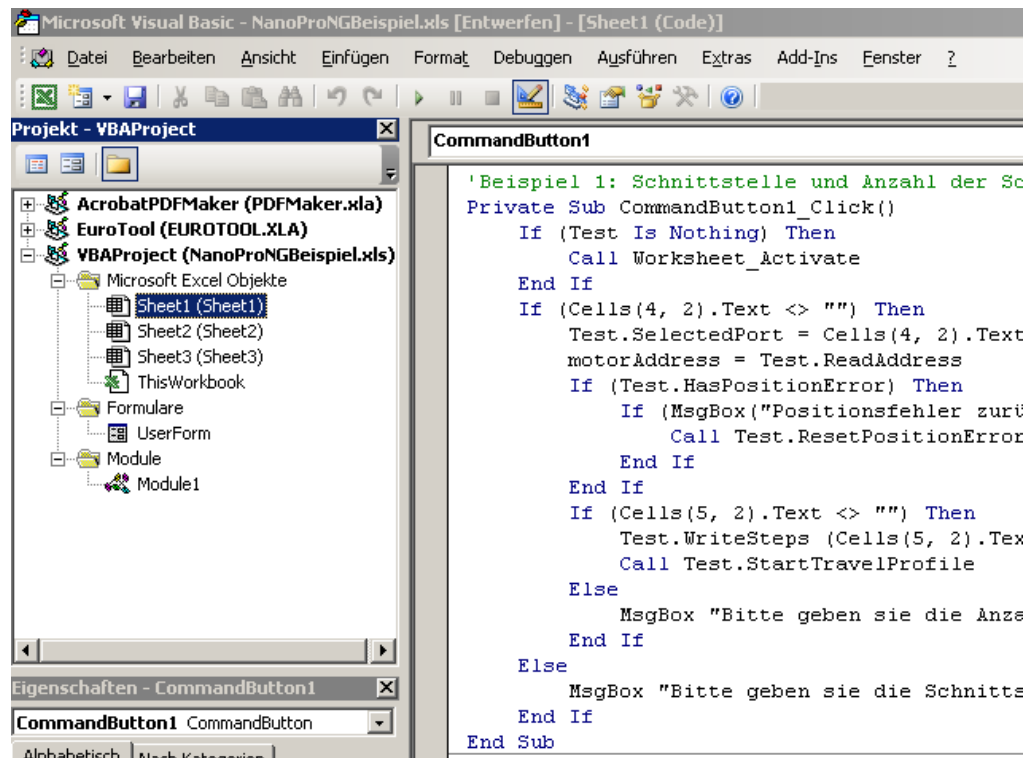
## Anwendungsbeispiel der DLL-Bibliothek

Das nachfolgend abgebildete EXCEL-Beispiel veranschaulicht, wie unsere Steuerungen mittels der Scriptsprache Visual Basic for Applications (VBA) unter Einbeziehung der DLL schnell und unkompliziert angesprochen werden können.

### Aufruf-Button im EXCEL-Worksheet



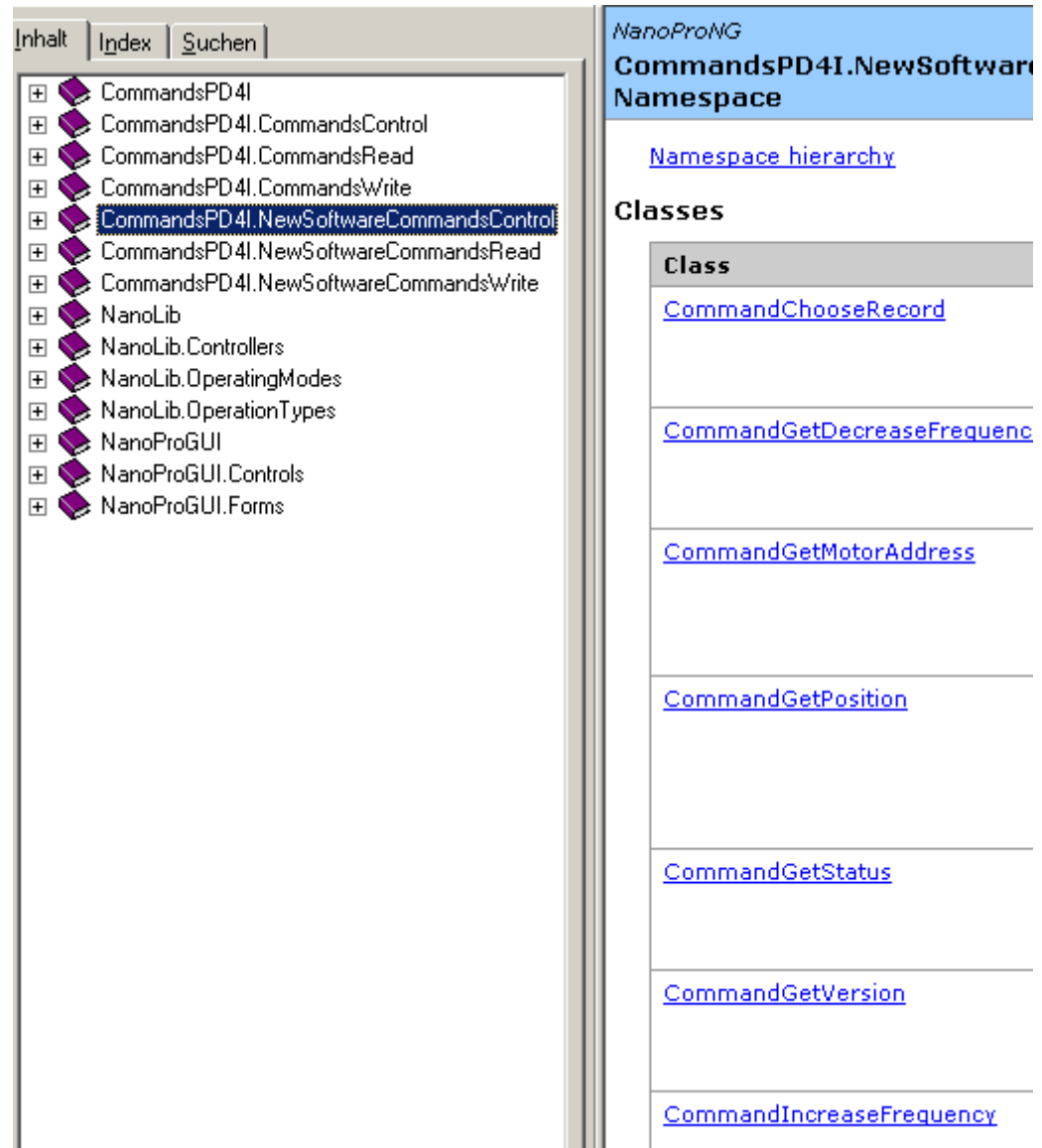
### Zugehöriges VBA-Script



## Windows-Hilfe zur DLL-Bibliothek

Die Befehle der DLL-Bibliothek sind in einer Windows-Hilfedatei nach Classes strukturiert und pro Befehl dokumentiert.

### Übersicht der CommandsPD4I.NewSoftwareCommandsControl Classes



The screenshot shows a Windows Help window with a tree view on the left and a content pane on the right. The tree view lists various namespaces and classes, with 'CommandsPD4I.NewSoftwareCommandsControl' selected. The content pane displays the 'CommandsPD4I.NewSoftwareCommandsControl Namespace' and lists several classes: CommandChooseRecord, CommandGetDecreaseFrequency, CommandGetMotorAddress, CommandGetPosition, CommandGetStatus, CommandGetVersion, and CommandIncreaseFrequency.

**NanoProNG**  
**CommandsPD4I.NewSoftwareCommandsControl Namespace**

[Namespace hierarchy](#)

**Classes**

Class
<a href="#">CommandChooseRecord</a>
<a href="#">CommandGetDecreaseFrequency</a>
<a href="#">CommandGetMotorAddress</a>
<a href="#">CommandGetPosition</a>
<a href="#">CommandGetStatus</a>
<a href="#">CommandGetVersion</a>
<a href="#">CommandIncreaseFrequency</a>



## Übersicht der CommandChooseRecord Class

Inhalt | Index | Suchen

- [-] CommandsPD4I
- [-] CommandsPD4I.CommandsControl
- [-] CommandsPD4I.CommandsRead
- [-] CommandsPD4I.CommandsWrite
- [-] CommandsPD4I.NewSoftwareCommandsControl
  - [-] **CommandChooseRecord Class**
  - [-] CommandGetDecreaseFrequency Class
  - [-] CommandGetMotorAddress Class
  - [-] CommandGetPosition Class
  - [-] CommandGetStatus Class
  - [-] CommandGetVersion Class
  - [-] CommandIncreaseFrequency Class
  - [-] CommandResetAllSettings Class
  - [-] CommandResetCounter Class
  - [-] CommandResetPosion Class
  - [-] CommandResetPositionError Class
  - [-] CommandSetRecord Class
  - [-] CommandStartTravelProfile Class
  - [-] CommandStopTravelProfile Class
  - [-] CommandTriggerOn Class
  - [-] CommandVersion Class
- [-] CommandsPD4I.NewSoftwareCommandsRead
- [-] CommandsPD4I.NewSoftwareCommandsWrite
- [-] NanoLib
- [-] NanoLib.Controllers
- [-] NanoLib.OperatingModes
- [-] NanoLib.OperationTypes

NanoProNG

### CommandChooseRecord Class

Stellt die Steuerungsfunktion für Auswählen eines Satzes bereit.

For a list of all members of this [CommandChooseRecord Member](#) [System.Object](#) [BaseCommand](#) [BaseSetCommand](#) **CommandChooseRecord**

```
public class CommandChooseRecord : BaseSetCommand
```

**Thread Safety**

Public static (**Shared** in Visual Basic) members of this type are safe for multithreaded operations. Instance members are **not** guaranteed to be thread-safe.

**Requirements**

**Namespace:** [CommandsPD4I.NewSoftwareCommandsControl](#)

**Assembly:** CommandsPD4I (in CommandsPD4I.dll)

**See Also**

[CommandChooseRecord Member](#)  
[CommandsPD4I.NewSoftwareCommandsControl Namespace](#)

## Übersicht der CommandChooseRecord Members

Inhalt | Index | Suchen

- [-] CommandsPD4I
- [-] CommandsPD4I.CommandsControl
- [-] CommandsPD4I.CommandsRead
- [-] CommandsPD4I.CommandsWrite
- [-] CommandsPD4I.NewSoftwareCommandsControl
  - [-] CommandChooseRecord Class
    - [-] **CommandChooseRecord Members**
    - [-] CommandChooseRecord Constructors
  - [-] CommandGetDecreaseFrequency Class
  - [-] CommandGetMotorAddress Class
  - [-] CommandGetPosition Class
  - [-] CommandGetStatus Class
  - [-] CommandGetVersion Class
  - [-] CommandIncreaseFrequency Class
  - [-] CommandResetAllSettings Class
  - [-] CommandResetCounter Class
  - [-] CommandResetPosion Class
  - [-] CommandResetPositionError Class
  - [-] CommandSetRecord Class
  - [-] CommandStartTravelProfile Class
  - [-] CommandStopTravelProfile Class
  - [-] CommandTriggerOn Class

NanoProNG

### CommandChooseRecord Member

[CommandChooseRecord overview](#)

**Public Instance Constructors**

<a href="#">CommandChooseRecord Constructor</a>	Initialisiert die CommandChooseRecord
---	---------------------------------------

**See Also**

[CommandChooseRecord Class](#) | [CommandsPD4I.NewSoftwareCommandsControl Namespace](#)

---

## 2 Lesebefehl

### Funktion

Eine ganze Reihe von Einstellungen, die mit einem bestimmten Befehl gesetzt werden können, können mit einem entsprechenden Lesebefehl ausgelesen werden.

Dieser Lesebefehl setzt sich aus dem Zeichen 'Z' und dem Befehl für den entsprechenden Parameter zusammen.

### Beispiel

Auslesen des Verfahrenweges: „#1Zs\r“ -> „1Zs1000\r“

---

## 3 Sätze

### Speichern von Verfahrenwegen

Die Firmware unterstützt das Speichern von Verfahrenwegen in Sätzen. Diese Daten werden in einem EEPROM abgelegt und gehen somit auch im ausgeschalteten Zustand nicht verloren.

Im EEPROM finden 32 Sätze mit den Satznummern 1 bis 32 Platz.

### Gespeicherte Einstellungen pro Satz

Folgende Einstellungen werden in jedem Satz gespeichert:

- 5.1 Positionierart setzen 'p':                    Positionsmodus
- 5.2 Verfahrenweg einstellen 's':                Verfahrenweg
- 5.3 Minimalfrequenz einstellen 'u':            Anfangsschrittfrequenz
- 5.4 Maximalfrequenz einstellen 'o':            Maximalschrittfrequenz
- 5.5 Maximalfrequenz 2 einstellen 'n':        Zweite Maximalschrittfrequenz
- 5.6 Rampe einstellen 'b':                    Beschleunigungs- und Bremsrampe
- 5.7 Richtung einstellen 'd':                    Drehrichtung
- 5.8 Richtungsumkehr einstellen 't':            Drehrichtungsumkehr bei  
Wiederholungssätzen
- 5.9 Wiederholungen einstellen 'w':            Wiederholungen
- 5.10 Satzpause einstellen 'P':                Pause zwischen Wiederholungen und  
Folgesätzen
- 5.11 Folgesatz einstellen 'N':                Satznummer des Folgesatzes

## 4 Allgemeine Befehle

### 4.1 Bootloader starten

#### Parameter

Zeichen	Parameter
'@A'	Keiner

#### Antwort der Firmware

Keine Antwort, Bootloader antwortet mit '@OK'

#### Beschreibung

Dieser Befehl weist die Firmware an, den Bootloader zu starten. Die Firmware antwortet selbst nicht auf den Befehl. Der Bootloader antwortet mit '@OK'.

Der Bootloader selbst benötigt diesen Befehl ebenfalls, damit er sich nicht automatisch nach einer halben Sekunde wieder beendet. Deswegen muss dieser Befehl so oft gesendet werden, bis der Bootloader mit '@OK' antwortet. Der Bootloader verwendet das gleiche Adressierungsschema wie die Firmware selbst.

### 4.2 Totbereich Joystickmodus

#### Parameter

Zeichen	Parameter
'='	Integer zwischen 0 und 100

#### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

#### Beschreibung

Stellt den Totbereich im Joystickmodus ein.

Im Joystickmodus kann der Motor über eine Spannung am Analogeingang vorwärts und rückwärts verfahren werden.

Der Wertebereich in der Mitte zwischen Maximal- und Minimal-Spannung, bei dem der Motor sich nicht dreht, ist der Totbereich. Er wird in Prozent zur Größe des Bereichs angegeben.

#### Auslesen

Mit dem Befehl 'Z=' kann der aktuell eingestellte Totbereich ausgelesen werden.

---

## 4.3 Fehlerspeicher auslesen

### Parameter

Zeichen	Parameter
'E'	keiner

### Antwort der Firmware

Liefert den Index des Fehlerspeichers mit dem zuletzt aufgetretenen Fehler.

### Beschreibung

Die Firmware beinhaltet 32 Fehlerspeicher. Dieser Befehl liefert den Index des Speicherplatzes mit dem zuletzt aufgetretenen Fehler zurück.

### Auslesen

Mit dem Befehl 'Z'+Index+'E' kann die Fehlernummer des jeweiligen Fehlerspeichers ausgelesen werden.

## 4.4 Drehgeberposition auslesen

### Parameter

Zeichen	Parameter
'T'	keiner

### Antwort der Firmware

Liefert die aktuelle Position des Motors laut Drehgeber zurück.

### Beschreibung

Bei Motoren mit einem Drehgeber gibt dieser Befehl die aktuelle Position laut Drehgeber in Motorschritten zurück. Solange der Motor keine Schritte verloren hat, stimmen die Werte des Befehls *4.15 Position auslesen 'C'* und des Befehls *4.41 Aktuellen Satz auslesen 'I' (Pipe)* überein.

Es ist dabei aber zu beachten, dass der Drehgeber für Schrittmodi höher als 1/10 bei 1,8° Motoren und höher als 1/5 bei 0,9° Motoren über eine zu geringe Auflösung verfügt und deswegen trotzdem Differenzen zwischen den beiden oben genannten Werten auftreten.

## 4.5 Fehlerkorrekturmodus einstellen

### Parameter

Zeichen	Parameter
'U'	Integer, erlaubte Werte 0, 1 und 2

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Setzt den Modus der Fehlerkorrektur.

Parameter 0 : Aus

Parameter 1: Korrektur nach einer Fahrt

Parameter 2: Korrektur während der Fahrt (nicht implementiert)

Bei einem Motor ohne Drehgeber muss dieser Wert explizit auf 0 gesetzt werden, sonst versucht dieser ständig zu korrigieren, weil er von Schrittverlusten ausgeht.

### Auslesen

Mit dem Befehl 'ZE'+Index kann die Fehlernummer des jeweiligen Fehlerspeichers ausgelesen werden.

## 4.6 Filter für Analog- und Joystickmodus einstellen

### Parameter

Zeichen	Parameter
'f'	Integer, erlaubte Werte von 0 bis 16

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Im Analog- und Joystickmodus wird der Analogeingang verwendet, um die Drehzahl einzustellen. Mit dem Befehl 'f' kann die Anzahl der Samples eingestellt werden, über die der endgültige Wert gemittelt wird.

### Auslesen

Mit dem Befehl 'Zf' kann der aktuell eingestellte Wert ausgelesen werden.

## 4.7 Schrittwinkel einstellen

### Parameter

Zeichen	Parameter
'a'	Integer, erlaubte Werte: 9 und 18

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Zum Umrechnen der Drehgeber-Position in die Rotorlage benötigt die Firmware Kenntnis über den Schrittwinkel des Motors. Für 0,9° Motore ist ein Wert von 9 und für 1,8° Motoren ist ein Wert von 18 einzustellen. Andere Werte werden nicht unterstützt.

### Auslesen

Mit dem Befehl 'Za' kann der aktuell eingestellte Wert ausgelesen werden.

## 4.8 Firmwareversion auslesen

### Parameter

Zeichen	Parameter
'v'	keine

### Antwort der Firmware

Liefert den Versionsstring der Firmware zurück.

### Beschreibung

Rückgabestring setzt sich aus mehreren Blöcken zusammen:

'v' Echo des Befehls

' ' Trennzeichen (Space)

Hardware: Möglich: 'PD4','PD4lc','PD2lc','SMCI32','SMCI47'

'\_' Trennzeichen

Kommunikation: 'USB' oder 'RS485'

'\_' Trennzeichen

Releasedatum: tt-mm-jjjj z.B. 26-09-2007

### Beispiel einer kompletten Antwort

„001v PD4\_RS485\_26-09-2007\r“

## 4.9 Eingänge maskieren

### Parameter

Zeichen	Parameter
'L'	Bitmaske als Integer

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert (d.h. die gesamte Maske wird verworfen).

### Beschreibung

Gültig ab Firmware Version 09-11-2007.

Diese Bitmaske hat 32 Bit.

Setzt eine Bitmaske, die die Nutzung der Ein- und Ausgänge durch den Nutzer zulässt. Ist das Bit der entsprechenden I/Os auf '1' gesetzt, verwendet die Firmware diese I/Os. Ist es auf '0', sind die I/Os für den Nutzer verwendbar. Siehe auch Befehl *4.10 Eingänge und Ausgänge auslesen 'Y'*.

Folgend die Belegung der Bits:

Bit0: Eingang 1

Bit1: Eingang 2

Bit2: Eingang 3

Bit3: Eingang 4

Bit4: Eingang 5

Bit5: Eingang 6

Bit16: Ausgang 1

Bit17: Ausgang 2

Alle anderen Bits sind '0'

Bei den PDx-N Varianten können nur Eingang 1 und Eingang 6 verwendet werden.

Werden ungültige Bitmasken gesetzt, werden diese verworfen, auch wenn die Firmware diese korrekt bestätigt.

### Auslesen

Mit dem Befehl 'ZL' kann die aktuell eingestellte Maske ausgelesen werden.



## 4.10 Eingänge und Ausgänge auslesen

### Parameter

Zeichen	Parameter
'Y'	Bitmaske als Integer

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Gültig ab Firmware Version 09-11-2007.

Diese Bitmaske hat 32 Bit.

Setzt die Ausgänge der Firmware, sofern diese für die freie Verwendung mittels des Befehls *4.9 Eingänge maskieren 'L'* maskiert sind.

Ausgang 1 entspricht Bit 16 und Ausgang 2 Bit 17.

### Auslesen

Mit dem Befehl 'ZY' kann der aktuell eingestellte Wert ausgelesen werden.

Zusätzlich wird der Status der Eingänge angezeigt.

Bit0: Eingang 1

Bit1: Eingang 2

Bit2: Eingang 3

Bit3: Eingang 4

Bit4: Eingang 5

Bit5: Eingang 6

Bit6: '0' wenn Drehgeber gerade am Indexstrich, sonst '1'

Bit 16: Ausgang 1 (so wie er vom Nutzer eingestellt ist, auch wenn die Firmware diesen gerade bedient)

Bit 17: Ausgang 2 (so wie er vom Nutzer eingestellt ist, auch wenn die Firmware diesen gerade bedient)

Alle anderen Bits sind 0.

Bei den Low-Cost Varianten existieren nur Eingang 1 (Bit0) und Eingang 2 (Bit5).

## 4.11 Firmwareversion auslesen (alt)

### Parameter

Zeichen	Parameter
' ' (Space)	keiner

### Antwort der Firmware

String der Firmwareversion (const, da neuer Befehl 'v' die Funktion übernommen hat).

### Beschreibung

Für Bootloader benötigt, sonst nutzlos.

## 4.12 Status auslesen

### Parameter

Zeichen	Parameter
'\$'	keiner

### Antwort der Firmware

Liefert den Status der Firmware als Bitmaske zurück.

### Beschreibung

Die Bitmaske hat 8 Bit.

Bit 0: 1: Steuerung bereit

Bit 1: 1: Nullposition erreicht

Bit 2: 1: Positionsfehler

Bit 3: 1: Eingang 1 ist gesetzt während Steuerung wieder bereit ist. Tritt dann auf, wenn die Steuerung über Eingang 1 gestartet wurde und die Steuerung schneller wieder bereit ist, als der Eingang zurückgesetzt wurde.

Bit 4 bis 6 geben den aktuellen Modus als Integer an:

0: unbenutzt

1: Steuerung im Positioniermodus

2: Steuerung im Drehzahlmodus

3: Steuerung im Flagpositioniermodus

4: Steuerung Taktrichtungsmodus

5: Analogmodus

6: Joystickmodus

7: unbenutzt

Bit 7 ist unbelegt

## 4.13 Motoradresse auslesen

### Parameter

Zeichen	Parameter
'M'	keine

### Antwort der Firmware

Liefert die Motoradresse zurück.

### Beschreibung

Liefert die serielle Adresse zurück. Ist vor allem im Zusammenhang mit der Adressierungsart '\*' sinnvoll, wenn die Motoradresse nicht bekannt ist.

## 4.14 Parameter auslesen

### Parameter

Zeichen	Parameter
'Z'	Auslesbarer Befehl und optional zugehörige Satznummer

### Antwort der Firmware

Liefert den jeweils gewünschten Parameter zurück.

### Beschreibung

Dient zum Auslesen der aktuell gesetzten Werte einiger Befehle. Das Auslesen des Verfahrenweges geschieht beispielsweise mit 'Zs', worauf die Firmware mit 'Zs1000' antwortet.

Soll der Parameter eines bestimmten Satzes gelesen werden, ist dem jeweiligen Befehl die Nummer des Satzes voranzustellen.

Beispiel: 'Z5s' -> 'Z5s2000'

Eine Liste der Satzbefehle findet sich unter „2 Lesebefehl“

## 4.15 Position auslesen

### Parameter

Zeichen	Parameter
'C'	keiner

### Antwort der Firmware

Liefert die aktuelle Position zurück.

### Beschreibung

Liefert die aktuelle Position des Motors in Schritten des jeweils eingestellten Schrittmodus. Diese Position ist relativ zu der Position der letzten Referenzfahrt.

Verfügt der Motor über einen Winkelgeber, sollte dieser Wert mit dem des Befehls 'I' bis auf eine kleine Toleranz übereinstimmen.

Die Toleranz ist abhängig von Schrittmodus und Motortyp (0,9° oder 1,8°), da der Winkelgeber eine geringere Auflösung als der Motor im Mikroschrittbetrieb hat.

Der Wertebereich ist -100000000 bis +100000000.

## 4.16 Motoradresse einstellen

### Parameter

Zeichen	Parameter
'm'	Integer, erlaubte Werte zwischen 1 und 254

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Setzt die Motoradresse. Es ist darauf zu achten, dass die neu gesetzte Adresse nicht bereits von einem anderen Motor belegt ist, sonst ist keine Kommunikation mehr möglich.

Adresse 0 und 255 sind für Fehlerfälle des EEPROMS reserviert.

### Auslesen

Mit dem Befehl 'Zm' kann die aktuelle Adresse ausgelesen werden. Siehe auch Befehl 4.13 *Motoradresse auslesen 'M'*.

## 4.17 Schrittmodus einstellen

### Parameter

Zeichen	Parameter
'g'	Integer, erlaubte Werte: 1, 2, 4, 5, 8, 10, 16, 32, 64 und 255

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Setzt den Schrittmodus. Die übergebene Zahl entspricht der Anzahl der Mikroschritte pro Vollschritt mit Ausnahme des Wertes 255, welche den adaptiven Schrittmodus auswählt.

### Auslesen

Mit dem Befehl 'Zg' kann der aktuell gültige Wert ausgelesen werden.

---

## 4.18 Phasenstrom einstellen

### Parameter

Zeichen	Parameter
'I'	Integer, erlaubte Werte zwischen 0 und 150

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Setzt den Phasenstrom in Prozent. Werte über 100 sollten vermieden werden.

### Auslesen

Mit dem Befehl 'Zi' kann der aktuell gültige Wert ausgelesen werden.

## 4.19 Phasenstrom im Stillstand einstellen

### Parameter

Zeichen	Parameter
'I'	Integer, erlaubte Werte zwischen 0 und 150

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Setzt den Strom der Stromreduzierung in Prozent. Dieser Wert ist wie der Phasenstrom relativ zum Endwert und nicht relativ zum Phasenstrom. Werte über 100 sollten vermieden werden.

### Auslesen

Mit dem Befehl 'Zr' kann der aktuell gültige Wert ausgelesen werden.

## 4.20 Motormodus einstellen

### Parameter

Zeichen	Parameter
'!	Integer, erlaubte Werte zwischen 1 und 6

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Setzt den Motormodus. Es sind 6 verschiedene Modi verfügbar:

- 1: Positionsmodus
- 2: Drehzahlmodus
- 3: Flagpositionsmodus
- 4: Takt-Richtungsmodus
- 5: Analogmodus
- 6: Joystickmodus

### Auslesen

Mit dem Befehl 'Z!' kann der aktuell gültige Wert ausgelesen werden.

## 4.21 Endschalerverhalten einstellen

### Parameter

Zeichen	Parameter
'I'	Integer, Bitmaske, Werte siehe Beschreibung

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Setzt das Endschalerverhalten. Der Integer-Parameter wird als Bitmaske interpretiert. Die Bitmaske hat 16 Bit.

Freifahrt bedeutet, dass die Steuerung bei Erreichen des Schalters mit der eingestellten unteren Geschwindigkeit vom Schalter herunterfährt.

Stop bedeutet, dass die Steuerung bei Erreichen des Schalters sofort anhält. Der Schalter bleibt dabei gedrückt.

#### Verhalten des internen Endschalers bei Referenzfahrt:

Bit0: Freifahrt vorwärts

Bit1: Freifahrt rückwärts

Es muss genau eines der beiden Bits gesetzt sein.

#### Verhalten bei Auslösen des internen Endschalers bei Normalfahrt:

Bit2: Freifahrt vorwärts

Bit3: Freifahrt rückwärts

Bit4: Stopp

Bit5: Ignorieren

Es muss genau eines der vier Bits gesetzt sein.

Diese Einstellung ist dann sinnvoll, wenn der Motor sich nicht mehr als eine Umdrehung drehen darf.

#### Verhalten des externen Endschalers bei Referenzfahrt:

Bit9: Frei vorwärts

Bit10: Frei rückwärts

Es muss genau eines der beiden Bits gesetzt sein.

#### Verhalten des externen Endschalers bei Normalfahrt:

Bit11: Freifahrt vorwärts

Bit12: Freifahrt rückwärts

Bit13: Stopp

Bit14: Ignorieren

Es muss genau eines der vier Bits gesetzt sein.

Mit dieser Einstellung kann der Verfahrweg des Motors durch einen Endschalter hart begrenzt werden.

### Auslesen

Mit dem Befehl 'ZI' kann der aktuell gültige Wert ausgelesen werden.

## 4.22 Endschalertyp einstellen

### Parameter

Zeichen	Parameter
'e'	Integer, erlaubte Werte sind 0 und 1

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Gibt den Typ des Endschalters an.

'0' bedeutet Öffner, '1' bedeutet Schließer.

Mit diesem Parameter wird der Firmware kenntlich gemacht, wann diese den externen Endschalter betätigt sieht. Der Endschalter ist dabei zwischen Versorgungsspannung (bei SMC1xx an +5V) und Eingang 6 zu schalten.

Deswegen bedeutet 'Öffner', dass im Normalfall ein High-Level an Eingang anliegt, da der Schalter im Normalfall geschlossen ist. Wird der Schalter betätigt, öffnet ("Öffner") dieser den Kontakt und am Eingang liegt keine Spannung mehr an.

### Auslesen

Mit dem Befehl 'Ze' kann der aktuell gültige Wert ausgelesen werden.



## 4.23 Satz speichern

### Parameter

Zeichen	Parameter
'>'	Integer, erlaubte Werte zwischen 1 und 32

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Dient zum Speichern der aktuell eingestellten Befehle (im RAM) in einem Satz im EEPROM. Der Parameter ist die Satznummer, in der die Daten gespeichert werden.

Während einer Fahrt sollte dieser Befehl nicht aufgerufen werden, da die aktuellen Werte sich durch Folgefahrten ändern.

Zu einem Satz gehören die Befehle:

- 5.1 Positionierart setzen 'p'
- 5.2 Verfahrenweg einstellen 's'
- 5.3 Minimalfrequenz einstellen 'u'
- 5.4 Maximalfrequenz einstellen 'o'
- 5.5 Maximalfrequenz 2 einstellen 'n'
- 5.6 Rampe einstellen 'b'
- 5.7 Richtung einstellen 'd'
- 5.8 Richtungsumkehr einstellen 't'
- 5.9 Wiederholungen einstellen 'w'
- 5.10 Satzpause einstellen 'P'
- 5.11 Folgesatz einstellen 'N'

## 4.24 Umkehrspiel einstellen

### Parameter

Zeichen	Parameter
'z'	Integer, erlaubte Werte zwischen 0 und 9999

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Gibt das Umkehrspiel in Schritten an.

Die Einstellung dient dazu, das Spiel von nachgeschalteten Getrieben bei einem Drehrichtungswechsel auszugleichen.

Hierzu macht der Motor bei einem Drehrichtungswechsel die im Parameter eingestellte Anzahl von Schritten, bevor er beginnt, die Position zu inkrementieren.

### Auslesen

Mit dem Befehl 'Zz' kann der aktuell gültige Wert ausgelesen werden.

## 4.25 Position zurückstellen

### Parameter

Zeichen	Parameter
'c'	keine

### Antwort der Firmware

Bestätigt den Befehl durch Echo.

### Beschreibung

Setzt die Position des Motors auf 0 zurück.

Die aktuelle Position des Motors ist danach die Referenzposition.

---

## 4.26 Automatisches Senden des Status einstellen

### Parameter

Zeichen	Parameter
'J'	Integer, erlaubte Werte sind 0 und 1

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Ist der Parameter auf '1' gesetzt, sendet die Firmware von sich aus nach Ende einer Fahrt den Status. Siehe Befehl *4.12 Status auslesen '\$'*, mit dem Unterschied, dass statt dem '\$' ein kleines 'j' gesendet wird.

### Auslesen

Mit dem Befehl 'ZJ' kann der aktuell gültige Wert ausgelesen werden.

## 4.27 Satz für Autokorrektur einstellen

### Parameter

Zeichen	Parameter
'F'	Integer, erlaubte Werte zwischen 1 und 32

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Setzt den Satz, der für die Korrekturfahrt verwendet wird.  
Siehe Befehl *4.5 Fehlerkorrekturmodus einstellen 'U'*.

### Auslesen

Mit dem Befehl 'ZF' kann der aktuell gültige Wert ausgelesen werden.

## 4.28 Encoderrichtung einstellen

### Parameter

Zeichen	Parameter
'q'	Integer, erlaubte Werte sind 0 und 1

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Wenn der Parameter auf '1' gesetzt ist, wird die Richtung des Drehencoders umgekehrt.

### Auslesen

Mit dem Befehl 'Zq' kann der aktuell gültige Wert ausgelesen werden.

## 4.29 Ausschwingzeit einstellen

### Parameter

Zeichen	Parameter
'O'	Integer, erlaubte Werte zwischen 0 und 255

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Gibt die Ausschwingzeit in 10ms Schritten zwischen Ende der Fahrt und der Überprüfung der Position durch den Drehgeber an.

Dieser Parameter ist nur gültig für die Positionsprüfung nach der Fahrt.  
Siehe Befehl *4.5 Fehlerkorrekturmodus einstellen 'U'*.

Zwischen Wiederholungs- oder Folgesätzen wird die Position nur geprüft, wenn die Pausezeit (siehe Befehl *5.10 Satzpause einstellen 'P'*) länger als die Ausschwingzeit ist.

Nach einem Satz wird zuerst die Ausschwingzeit abgewartet, bevor der Motor sich wieder bereit meldet.

### Auslesen

Mit dem Befehl 'ZO' kann der aktuell gültige Wert ausgelesen werden.

## 4.30 Maximale Abweichung Drehgeber einstellen

### Parameter

Zeichen	Parameter
'X'	Integer, erlaubte Werte zwischen 0 und 100

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Gibt die maximale Abweichung in Schritten zwischen Soll-Position und Drehgeber-Position an.

Bei Schrittmodi größer als 1/10-Schritt bei 1,8° und 1/5 Schritt bei 0,9° Motoren muss dieser Wert größer 0 sein, da der Drehgeber selbst dann eine geringere Auflösung als die Mikroschritte des Motors hat.

### Auslesen

Mit dem Befehl 'ZX' kann der aktuell gültige Wert ausgelesen werden.

## 4.31 Minimalspannung für Analogmodus einstellen

### Parameter

Zeichen	Parameter
'Q'	Integer, erlaubte Werte zwischen -100 und 100

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Gibt in 0,1V-Schritten den Bereichsanfang des Analogeingangs an.

### Auslesen

Mit dem Befehl 'ZQ' kann der aktuell gültige Wert ausgelesen werden.

## 4.32 Maximalspannung für Analogmodus einstellen

### Parameter

Zeichen	Parameter
'R'	Integer, erlaubte Werte zwischen -100 und 100

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Gibt in 0,1V-Schritten das Bereichsende des Analogeingangs an.

### Auslesen

Mit dem Befehl 'ZR' kann der aktuell gültige Wert ausgelesen werden.

## 4.33 Totbereich für Joystickmodus einstellen

### Parameter

Zeichen	Parameter
'%'	Integer, erlaubte Werte zwischen 0 und 100

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Gibt den Totbereich des Analogeingangs in Prozent des eingestellten Bereichs für den Joystickmodus an.

### Auslesen

Mit dem Befehl 'Z%' kann der aktuell gültige Wert ausgelesen werden.

## 4.34 Satz starten

### Parameter

Zeichen	Parameter
'A'	keine

### Antwort der Firmware

Bestätigt den Befehl durch Echo.

### Beschreibung

Startet die Fahrt mit den aktuell eingestellten Parametern.

## 4.35 Satz stoppen

### Parameter

Zeichen	Parameter
'S'	keine

### Antwort der Firmware

Bestätigt den Befehl durch Echo.

### Beschreibung

Bricht die aktuelle Fahrt ab.

Im Drehzahl-, Analog- und Joystickmodus die einzige Möglichkeit, den Motor in den Bereit-Zustand zu bringen.

Es werden keine Rampen gefahren, sondern der Motor sofort zum Stillstand gebracht. Dadurch können bei hohen Geschwindigkeiten Schrittverluste entstehen.

In den 3 oben genannten Modi sollte deswegen vor dem Stopp-Befehl die Geschwindigkeit heruntergefahren werden.

## 4.36 Drehzahl erhöhen

### Parameter

Zeichen	Parameter
'+'	keine

### Antwort der Firmware

Bestätigt den Befehl durch Echo.

### Beschreibung

Erhöht die Drehzahl im Drehzahlmodus um 100 Schritte/s.

## 4.37 Drehzahl verringern

### Parameter

Zeichen	Parameter
'-'	keine

### Antwort der Firmware

Bestätigt den Befehl durch Echo.

### Beschreibung

Verringert die Drehzahl im Drehzahlmodus um 100 Schritte/s.

## 4.38 Trigger auslösen

### Parameter

Zeichen	Parameter
'T'	keine

### Antwort der Firmware

Bestätigt den Befehl durch Echo.

### Beschreibung

Auslöser für den Flagpositionsmodus.

Vor Auslösen des Trigger fährt der Motor mit konstanter Drehzahl.

Nach Auslösen des Triggers fährt der Motor noch die eingestellte Strecke ab der Position, bei der der Trigger ausgelöst wurde und stoppt dann.

## 4.39 Positionsfehler zurücksetzen

### Parameter

Zeichen	Parameter
'D'	keine

### Antwort der Firmware

Bestätigt den Befehl durch Echo.

### Beschreibung

Setzt einen Fehler der Drehüberwachung zurück und setzt die aktuelle Position auf die, die der Drehgeber meldet.

## 4.40 Satz aus EEPROM laden

### Parameter

Zeichen	Parameter
'y'	Integer von 1 bis 32

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Lädt die Satzdaten des im Parameter übergebenen Satzes aus dem EEPROM.

Siehe auch Befehl 4.23 Satz speichern '>'.



## 4.41 Aktuellen Satz auslesen

### Parameter

Zeichen	Parameter
' ' (Pipe)	Integer; erlaubte Werte sind 0 und 1

### Antwort der Firmware

Bestätigt den Befehl durch Echo, wenn der Parameter auf '1' gesetzt wird. Sonst keine Antwort.

### Beschreibung

Implementiert ab Firmware Version 09-11-2007.

Ist der Parameter auf '0', antwortet die Firmware überhaupt nicht mehr auf Befehle, führt diese aber nach wie vor aus. Dies dient dazu, schnell Einstellungen an die Firmware zu schicken, ohne auf Bestätigungen zu warten.

### Auslesen

Mit dem Befehl 'Z|' schickt die Firmware alle Einstellungen des geladenen Satzes in einem Stück.

Mit 'Z5|' werden die Daten des Satz 5 im EEPROM gesendet.

Das Format entspricht dem der jeweiligen Befehle.

Es ist zu beachten, dass bei der Antwort das '|' Zeichen nicht gesendet wird. Siehe folgende Beispiele.

### Beispiele

```
#1Z|\r'
```

```
-> 'Zp+1s+1u+400o+860n+1000b+55800d+1t+0W+1P+0N+0\r'
```

```
#1Z5|\r'
```

```
-> 'Z5p+1s+400u+400o+1000n+1000b+2364d+0t+0W+1P+0N+0\r'
```

## 5 Satzbefehle

### 5.1 Positionierart setzen

#### Parameter

Zeichen	Parameter
'p'	Integer, erlaubte Werte zwischen 1 und 4 (je nach Motormodus)

#### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

#### Beschreibung

In den 6 verschiedenen Motormodi hat dieser Befehl jeweils eine andere Bedeutung:

##### Positionsmodus (!=1)

p=1

Relative Positionierung;

Der Befehl 5.2 *Verfahrweg einstellen* 's' gibt den Verfahrweg relativ zur aktuellen Position an.

Der Befehl 5.7 *Richtung einstellen* 'd' gibt die Richtung an.

Der Parameter 5.2 Verfahrweg einstellen 's' muss positiv sein.

p=2

Absolute Positionierung;

Der Befehl 5.2 *Verfahrweg einstellen* 's' gibt die Zielposition relativ zur Referenzposition an.

Der Befehl 5.7 *Richtung einstellen* 'd' wird ignoriert.

p=3

Interne Referenzfahrt;

Der Motor läuft mit der unteren Geschwindigkeit in die Richtung, die in Befehl 5.7 *Richtung einstellen* 'd' eingestellt ist, bis er den Indexstrich des Drehgeber erreicht. Danach läuft der Motor eine feste Anzahl von Schritten, so dass er den Indexstrich wieder verlässt. Für die Richtung des Freifahrens siehe Befehl 4.21

*Endschalterverhalten einstellen* 'l'. Dieser Modus macht nur bei Motoren mit eingebauten und angeschlossenen Drehgeber Sinn.

p=4

Externe Referenzfahrt;

Der Motor läuft mit der oberen Geschwindigkeit in die Richtung, die in Befehl 5.7 *Richtung einstellen* 'd' eingestellt ist, bis er den Endschalter erreicht hat. Danach wird je nach Einstellung eine Freifahrt durchgeführt.

Siehe Befehl 4.21 *Endschalterverhalten einstellen* 'l'.

##### Drehzahlmodus (!=2)

p=1

Drehzahlmodus;

Wird der Motor gestartet, dreht der Motor bis zur Maximaldrehzahl mit der eingestellten Rampe hoch. Änderungen in der Geschwindigkeit oder Drehrichtung werden mit der eingestellten Rampe sofort angefahren, ohne dass der Motor zwischendurch gestoppt werden muss.

p=2

Nicht belegt

p=3  
Interne Referenzfahrt;  
siehe Positionsmodus

p=4  
Externe Referenzfahrt;  
siehe Positionsmodus

#### **Flagpositionsmodus (!=3 )**

p=1  
Flagpositionsmodus;  
Nach dem Start fährt der Motor auf die Maximaldrehzahl hoch. Nach Eintreffen des Trigger-Events (Befehl 4.38 *Trigger auslösen 'T'* oder Trigger-Eingang) fährt der Motor noch den eingestellten Verfahrweg (Befehl 5.2 *Verfahrweg einstellen 's'*) und verändert hierzu seine Geschwindigkeit auf die Maximalgeschwindigkeit<sup>2</sup> (Befehl 5.5 *Maximalfrequenz 2 einstellen 'n'*).

p=2  
Nicht belegt

p=3  
Interne Referenzfahrt;  
siehe Positionsmodus

p=4  
Externe Referenzfahrt;  
siehe Positionsmodus

#### **Taktrichtungsmodus (!=4 )**

p=1  
Auto-Modus;  
Motor macht 10 Einzelschritte und steigert seine Drehzahl dann auf den eingestellten Wert der Maximaldrehzahl, bis die Freigabe wieder aufgehoben wird.

p=2  
Nicht belegt

p=3  
Interne Referenzfahrt;  
siehe Positionsmodus

p=4  
Externe Referenzfahrt;  
siehe Positionsmodus

#### **Analogmodus (!=5 )**

Nicht zutreffend

#### **Joystickmodus (!=6 )**

Nicht zutreffend

### **Auslesen**

Mit dem Befehl 'Z!' kann der aktuell gültige Wert ausgelesen werden.

## 5.2 Verfahrenweg einstellen

### Parameter

Zeichen	Parameter
's'	Integer

### Antwort der Firmware

Bestätigt den Befehl durch Echo.

### Beschreibung

Befehl gibt den Verfahrenweg in (Mikro-)Schritten an. Für die relative Positionierung sind nur positive Werte erlaubt. Die Richtung wird mit Befehl 5.7 *Richtung einstellen 'd'* eingestellt.

Für die absolute Positionierung gibt dieser Befehl die Zielposition an. Negative Werte sind hier erlaubt. Die Drehrichtung aus Befehl 5.7 *Richtung einstellen 'd'* wird ignoriert, da diese sich aus der aktuellen Position und der Zielposition ergibt.

Der Wertebereich ist von -100000000 bis +100000000.

Im adaptiven Modus bezieht sich dieser Parameter auf Halbschritte.

### Auslesen

Mit dem Befehl 'Zs' kann der aktuell gültige Wert ausgelesen werden.

## 5.3 Minimalfrequenz einstellen

### Parameter

Zeichen	Parameter
'u'	Integer, erlaubte Werte zwischen 60 und 25000 (Minimalfrequenz von 1 Hz ist in Arbeit)

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Gibt die Minimalgeschwindigkeit in Hertz (Schritte pro Sekunde) an.

Bei einem Start eines Satzes beginnt der Motor, sich mit der Minimalgeschwindigkeit zu drehen. Er fährt dann mit der eingestellten Rampe (Befehl 5.6 *Rampe einstellen 'b'*) bis zur Maximalgeschwindigkeit (Befehl 5.4 *Maximalfrequenz einstellen 'o'*) hoch.

### Auslesen

Mit dem Befehl 'Zu' kann der aktuell gültige Wert ausgelesen werden.

## 5.4 Maximalfrequenz einstellen

### Parameter

Zeichen	Parameter
'o'	Integer, erlaubte Werte zwischen 60 und 25000 (Minimalfrequenz von 1 Hz ist in Arbeit)

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Gibt die Maximalgeschwindigkeit in Hertz (Schritte pro Sekunde) an.

Die Maximalgeschwindigkeit wird erst nach Durchfahren der Beschleunigungsrampe erreicht.

### Auslesen

Mit dem Befehl 'Zo' kann der aktuell gültige Wert ausgelesen werden.

## 5.5 Maximalfrequenz 2 einstellen

### Parameter

Zeichen	Parameter
'n'	Integer, erlaubte Werte zwischen 60 und 25000 (Minimalfrequenz von 1 Hz ist in Arbeit)

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Gibt die Maximalgeschwindigkeit 2 in Hertz (Schritte pro Sekunde) an.

Die Maximalgeschwindigkeit2 wird erst nach Durchfahren der Beschleunigungsrampe erreicht.

Dieser Wert findet ausschließlich im Flagpositionsmodus Anwendung. Siehe Befehl 5.1 *Positionierart setzen 'p'*.

### Auslesen

Mit dem Befehl 'Zn' kann der aktuell gültige Wert ausgelesen werden.

## 5.6 Rampe einstellen

### Parameter

Zeichen	Parameter
'b'	Integer, erlaubte Werte zwischen 1 und 65535

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Gibt die Beschleunigungsrampe (und momentan noch gleichzeitig die Bremsrampe) an.

Zum Umrechnen der Parameters in die Beschleunigung in Hz/ms wird die folgende Formel verwendet:

Beschleunigung in Hz/ms = ( (3000.0 / sqrt((float)<parameter>)) - 11.7 ).

### Auslesen

Mit dem Befehl 'Zb' kann der aktuell gültige Wert ausgelesen werden.

## 5.7 Richtung einstellen

### Parameter

Zeichen	Parameter
'd'	Integer, erlaubte Werte sind 0 und 1

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Setzt die Drehrichtung.

### Auslesen

Mit dem Befehl 'Zd' kann der aktuell gültige Wert ausgelesen werden.

## 5.8 Richtungsumkehr einstellen

### Parameter

Zeichen	Parameter
't'	Integer, erlaubte Werte sind 0 und 1

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Bei Wiederholungssätzen wird die Drehrichtung des Motors bei jeder Wiederholung umgedreht falls dieser Parameter auf '1' gesetzt ist. Siehe Befehl 5.9 *Wiederholungen einstellen 'w'*.

### Auslesen

Mit dem Befehl 'Zt' kann der aktuell gültige Wert ausgelesen werden.

## 5.9 Wiederholungen einstellen

### Parameter

Zeichen	Parameter
'W'	Integer, erlaubte Werte zwischen 0 und 254

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Gibt die Anzahl der Durchgänge des aktuellen Satzes an.

Ein Wert von 0 bedeutet unendliche Wiederholungen.

Normalerweise ist ein Wert von 1 für einen Durchgang eingestellt.

### Auslesen

Mit dem Befehl 'ZW' kann der aktuell gültige Wert ausgelesen werden.

## 5.10 Satzpause einstellen

### Parameter

Zeichen	Parameter
'P'	Integer, erlaubte Werte zwischen 0 und 65535

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Gibt die Pause zwischen Wiederholungen von Sätzen oder zwischen Satz und Folgesatz in ms (Millisekunden) an.

Hat ein Satz keinen Folgesatz oder Wiederholung, wird die Pause nicht durchgeführt und der Motor ist sofort nach Ende der Fahrt wieder bereit.

### Auslesen

Mit dem Befehl 'ZP' kann der aktuell gültige Wert ausgelesen werden.

## 5.11 Folgesatz einstellen

### Parameter

Zeichen	Parameter
'N'	Integer, erlaubte Werte zwischen 0 und 32

### Antwort der Firmware

Bestätigt den Befehl durch Echo (auch ungültige Werte).

Ungültige Werte werden ignoriert.

### Beschreibung

Gibt die Nummer des Folgesatzes an. Ist der Parameter auf '0', wird kein Folgesatz ausgeführt.

### Auslesen

Mit dem Befehl 'ZN' kann der aktuell gültige Wert ausgelesen werden.



---

## Index

### A

Aktuellen Satz auslesen .....	33
Altes Kommandierungsformat .....	5
Analogmodus setzen .....	35
Aufbau Steuerungsbehl .....	5
Ausgänge auslesen .....	17
Ausschwingzeit einstellen.....	28
Automatisches Senden des Status einstellen .....	27

### B

Bootloader starten .....	12
--------------------------	----

### D

Drehgeberposition auslesen.....	13
Drehzahl erhöhen .....	31
Drehzahl verringern .....	31
Drehzahlmodus setzen .....	34

### E

Eingänge auslesen .....	17
Eingänge maskieren.....	16
Encoderrichtung einstellen .....	28
Endschaltertyp einstellen.....	24
Endschalterverhalten einstellen .....	23

### F

Fehlerkorrekturmodus einstellen .....	14
Fehlerspeicher auslesen .....	13
Filter für Analogmodus einstellen .....	14
Filter für Joystickmodus einstellen .....	14
Firmwareversion auslesen (alt) .....	17
Flagpositionsmodus setzen .....	35
Folgesatz einstellen.....	40

### J

Joystickmodus setzen .....	35
----------------------------	----

### L

Lesebefehl .....	10
------------------	----

### M

Maximale Abweichung Drehgeber einstellen	29
Maximalfrequenz 2 einstellen .....	37
Maximalfrequenz einstellen .....	37
Maximalspannung für Analogmodus einstellen .....	30
Minimalfrequenz einstellen .....	36
Minimalspannung für Analogmodus einstellen .....	29
Motoradresse auslesen .....	18
Motoradresse einstellen.....	20
Motormodus einstellen.....	22

### P

Parameter auslesen.....	19
Phasenstrom einstellen .....	21
Phasenstrom im Stillstand einstellen .....	21
Position auslesen.....	19
Position zurückstellen .....	26
Positionierart setzen .....	34
Positionsfehler zurücksetzen .....	32
Positionsmodus setzen.....	34

### R

Rampe einstellen .....	38
Reaktion der Steuerung.....	5
Richtung einstellen.....	38
Richtungsumkehr einstellen.....	39

### S

Satz aus EEPROM laden .....	32
Satz für Autokorrektur einstellen.....	27
Satz speichern .....	25
Satz starten.....	30
Satz stoppen.....	31
Sätze .....	11
Satzpause einstellen.....	40
Schrittmodus einstellen.....	20
Speichern von Verfahrenswegen .....	11

Status auslesen.....18

## **T**

Taktrichtungsmodus setzen.....35

Totbereich für Joystickmodus einstellen .....30

Totbereich Joystickmodus .....12

Trigger auslösen.....32

## **U**

Umkehrspiel einstellen .....26

## **V**

Verfahrweg einstellen ..... 36

Verfahrwege speichern..... 11

## **W**

Wiederholungen einstellen ..... 39