

Anwenderhandbuch

Ausgabe 07/2002
Datei MAE-PBstepper-D.xxx
Best.Bez.: MAE-Pbstepper-D

PROFIBUS DP - STEPPER 6411-PBX

**für Firmware
Version 5.1**



Danaher Motion GmbH
Robert-Bosch-Str. 10
D-64331 Weiterstadt
Tel.: +49 (0)6151 - 8796 - 10
Fax: +49 (0)6151 - 8796 - 123
Email: info@danahermotion.net
Internet: www.danahermotion.net

BAUTZ

Verwendete Warenzeichen:

IBM, PC AT sind eingetragene Warenzeichen der International Business Machines Corp.
Simatic S5 ist eingetragenes Warenzeichen der Siemens AG

Dieses Dokument ist urheberrechtlich geschützt.

(Copyright 2002, Danaher Motion GmbH).

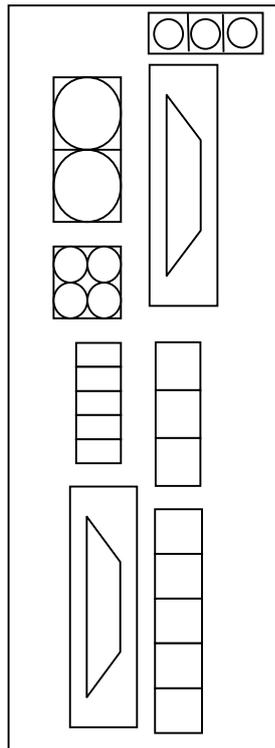
Die Firma Danaher Motion GmbH behält sich das Recht vor, Veränderungen, die der Verbesserung des in diesem Handbuch beschriebenen Produktes dienen, ohne gesonderte Ankündigung vorzunehmen. Für Hinweise in Bezug auf Fehler, Unklarheiten oder Unstimmigkeiten sind die Autoren jederzeit dankbar.

Inhaltsverzeichnis

1. Allgemeines.....	4
2. Aufbau der Hardware	5
3. Konfiguration der PROFIBUS DP Adresse.....	6
4. Handhabung.....	7
5. Diagnose-LED und Alarmzustand.....	11
6. Programmierung des PBS Schrittmotor-Controllers	12
6.1 Aufbau des Kommandowortes und Beschleunigungscode	13
6.2 Aufbau der Statuswortes	15
6.3 Vorgabe des Geschwindigkeitscodes.....	16
6.4 Positionierung	19
6.5 Nutzung der Positionsvorgabe	20
7. Kommandobeispiele	21
8. Aufruf der Referenzfahrt	25
9. Die Notstopp-Funktion.....	28
Anhang A – Gehäuseabmessungen	29

1. Allgemeines

Der DP STEPPER ist die Lösung, wenn es darum geht, einzelne im Feld verteilte Schrittmotor-Antriebe über den PROFIBUS DP anzusteuern. Der DP STEPPER ist eine kompakte Einachspositioniersteuerung mit integrierter Schrittmotorendstufe. Er erfasst zwei Endschalter, einen Stop Schalter (Interrupt-Eingang) und einen Referenzschalter. Zur Ansteuerung des STEPPER über den PROFIBUS DP genügen jeweils acht Eingangs- und Ausgangsbytes, um Positionieraufgaben für eine Achse zu bewältigen. Da nur der Prozessdatenkanal genutzt wird, kann der STEPPER ohne zusätzlichen Aufwand in jedes Steuerungssystem, das den PROFIBUS DP als Sensor-/Aktor-Bus nutzt, integriert werden. Durch die schnelle und zeitgleiche Übertragung der Ein- und Ausgangsbytes bei allen PROFIBUS DP Teilnehmern ergeben sich vielfältige Möglichkeiten zur Realisierung von Mehrachsantrieben über den Bus, ohne Synchronisationsprobleme in Kauf zu nehmen.

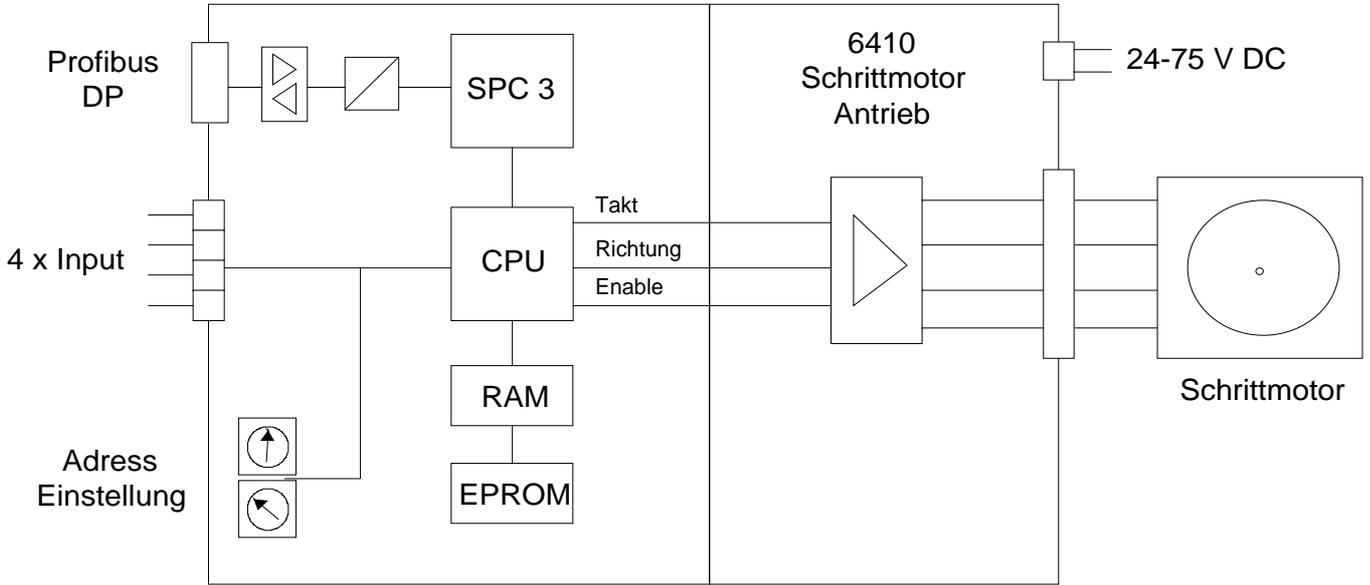


Die Antriebs-Taktrate kann flexibel von der Steuerung eingestellt werden, auch während der Antrieb läuft. Auch die Eingabe einer Beschleunigung bis zu einer Endgeschwindigkeit sowie die Vorwahl einer Zielposition sind möglich, auch während der Antrieb läuft.

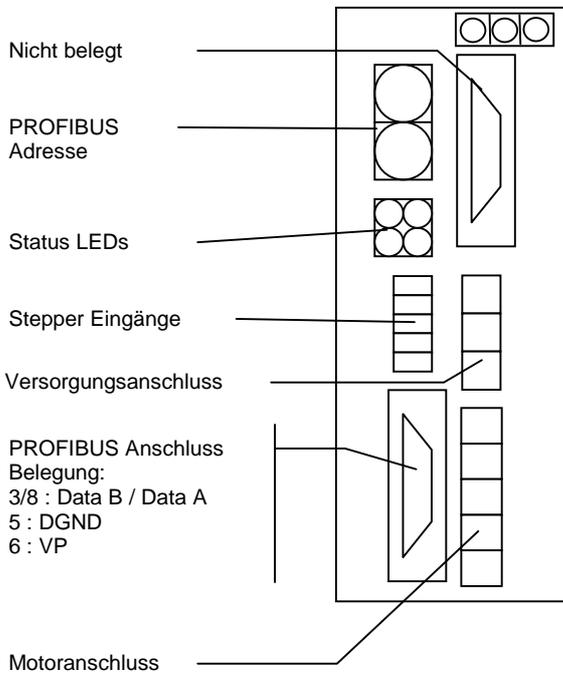
Alle Zustände können über zwei Eingangsbytes von der Steuerung eingelesen werden. Die Implementierung von schnellen Positionieraufgaben mit dem STEPPER ist daher sehr flexibel realisierbar. Für spezielle Aufgaben ist eine Implementierung von kundenspezifischen Funktionen in die Firmware des STEPPER schnell und kostengünstig möglich.

2. Aufbau der Hardware

Das untenstehende Blockschaltbild zeigt den Aufbau der Hardware des STEPPER :

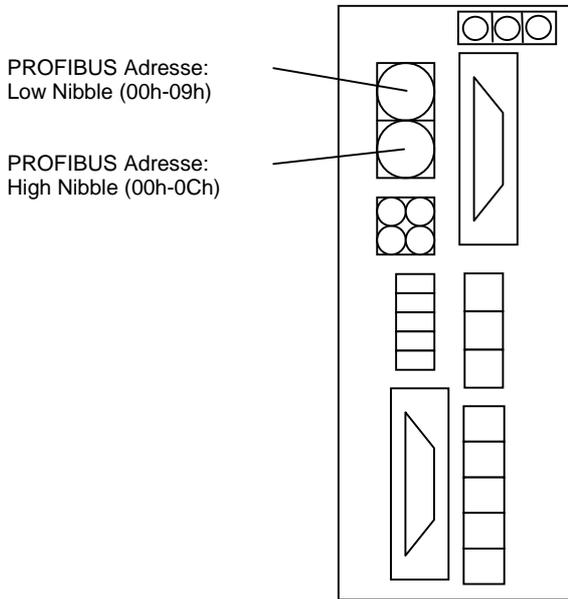


Position und Verwendung der Anschlüsse:



3. Konfiguration der PROFIBUS DP Adresse

Die PROFIBUS Adresse wird manuell über zwei Drehschalter eingestellt. Gültiger Wertebereich der PROFIBUS Adresse: 1-126

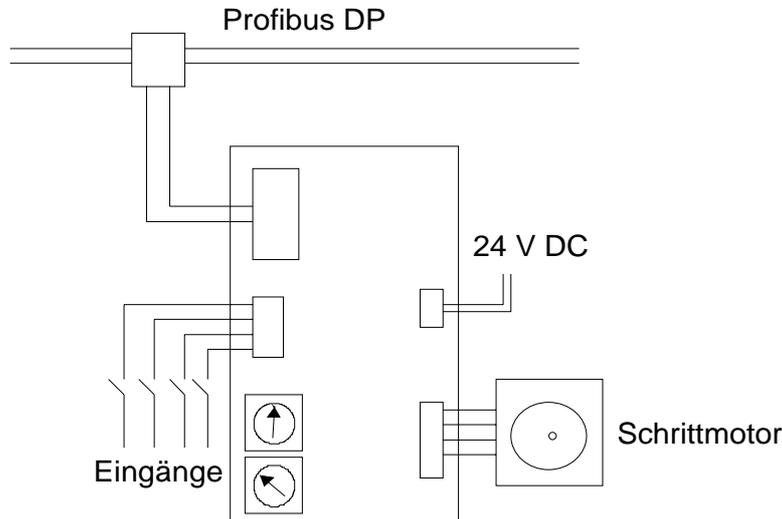


Beispiele für PROFIBUS Adressen:

Adresse High Nibble		Adresse Low Nibble		PROFIBUS Adresse
Hex	Dez	Hex	Dez	Dez
00	00	08	8	008
03	03	00	0	030
09	09	05	5	095
0A	10	03	3	103
0C	12	06	6	126

4. Handhabung

Der DP Stepper besitzt einen Feldbus-Anschluss zum PROFIBUS DP und wird wie ein Teilnehmer in den PROFIBUS DP integriert. Die Ein- und Ausgänge sind galvanisch von der Steuerelektronik getrennt und werden extern mit 24 V DC gespeist. Alle Einstellungen und die Referenzfahrt werden komplett über die PROFIBUS DP Schnittstelle eingestellt und können während des Laufs über den Bus verändert werden. Ein Ausgang ist mit dem Busy-Signal belegt, um elektrische Verriegelungsfunktionen bei laufendem Antrieb zu realisieren.



Auf Seiten des PROFIBUS DP verhält sich der STEPPER wie ein Teilnehmer mit **acht Eingangsbytes** und **zwölf Ausgangsbytes**. Von der Steuerung können Aktionen des Antriebs durch das Setzen verschiedener Bits im **Kommandowort (Ausgangsbyte 7+8)** gestartet oder geändert werden. Der aktuelle **Status** und die aktuelle Position können durch einfaches Lesen der **Eingangsbytes 3-8** jederzeit von der Steuerung abgefragt werden. In die Ausgangsbytes 5-8 wird die absolute Zielposition eingetragen, bis zu der der Stepper nach dem nächsten Motor Start Befehl laufen soll. Diese Zielposition kann auch während der Fahrt verändert werden. Auf diese Weise kann eine exakte Wegpositionierung vorgenommen werden, ohne den PROFIBUS DP Master zu belasten.

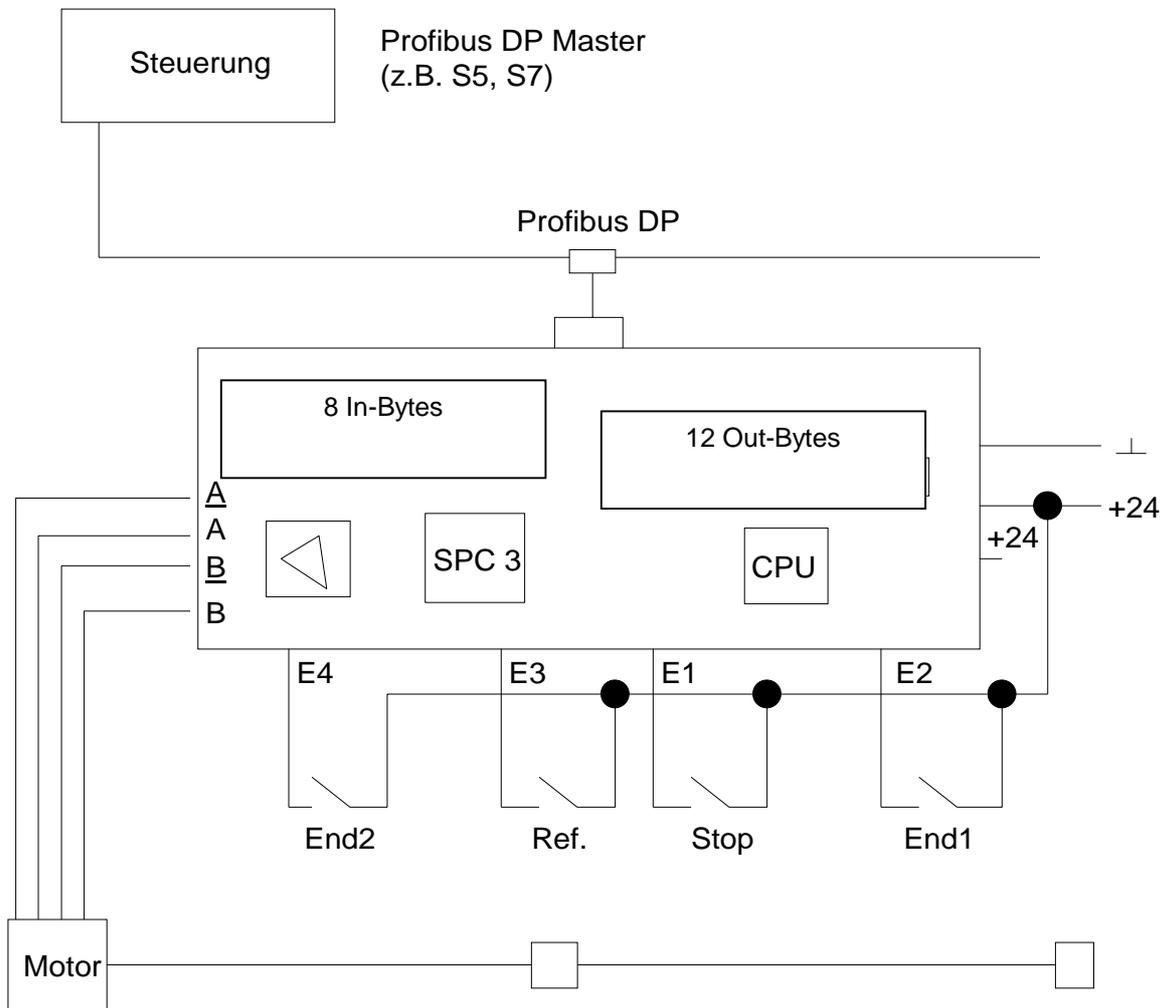
12 Ausgangsbytes:

Byte-Nr.	Bedeutung
1+2	Geschwindigkeitscode: Soll-Geschwindigkeit, bei Normalfahrt , Soll-Endgeschwindigkeit, bei Beschleunigungsfahrt .
3+4	Start-Geschwindigkeit bei Beschleunigungsfahrten
5+6	Beschleunigungscode, bestehend aus Zeitbasis und Werteteilung
7+8	Kommandowort
9+10	Zielpositionsvorgabe low-Wort
11+12	Zielpositionsvorgabe high-Wort

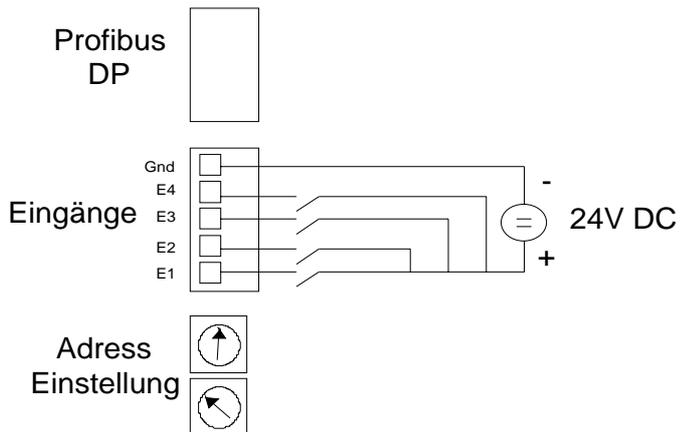
8 Eingangsbytes:

Byte-Nr.	Bedeutung
1+2	Anzeige der aktuellen Geschwindigkeit
3+4	Statusbytes
5+6	Ist-Position low-Wort
7+8	Ist-Position high-Wort

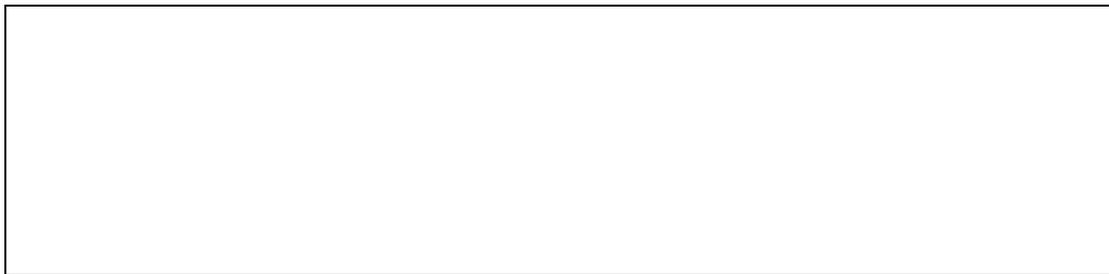
Beschaltung der Ein- und Ausgänge



Beschaltung der Eingänge



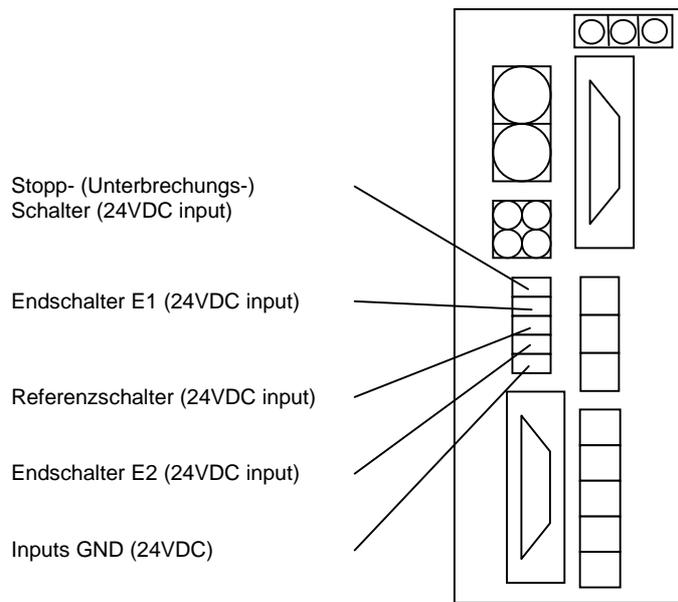
Schematischer Aufbau einer Antriebsachse für den STEPPER



Aufstellung der Ein- und Ausgänge:

- Endschalter oben (Öffner) -----> Eingang 2
- Endschalter unten (Öffner) -----> Eingang 4
- Referenzpunktkontakt (Schließer) -----> Eingang 3
- Stopp (Schließer oder Öffner) -----> Eingang 1

Belegung der Eingänge:



5. Diagnose-LED und Alarmzustand

Um die korrekte Funktion des Modulprozessors auf einem Blick sichtbar zu machen, wurde eine Diagnose-LED verwendet. Die Diagnose-LED blinkt im 5Hz-Takt.

! Der Alarmzustand !

Sobald der Stepper den Alarmzustand einschaltet, signalisiert die Diagnose-LED diesen Zustand durch statisches Leuchten. Der Alarmzustand wird durch folgende Ereignisse ausgelöst:

1. Bei der Referenzfahrt wurde der untere Endschalter aktiviert. (siehe Abschnitt 8).
2. Es liegt ein Prozessorfehler vor.

Während des Alarmzustandes werden **keine** Kommandos verarbeitet. Der Stepper blockiert sämtliche PBS-Kommando-Informationen. Die PBS-Input-Informationen d.h. Statusinformationen und Positionsmeldung werden aber weiterhin gemeldet. In den Statusinformationen ist dann auch die Meldung „Alarmzustand aktiv“ enthalten.

! Alarmzustand beenden !

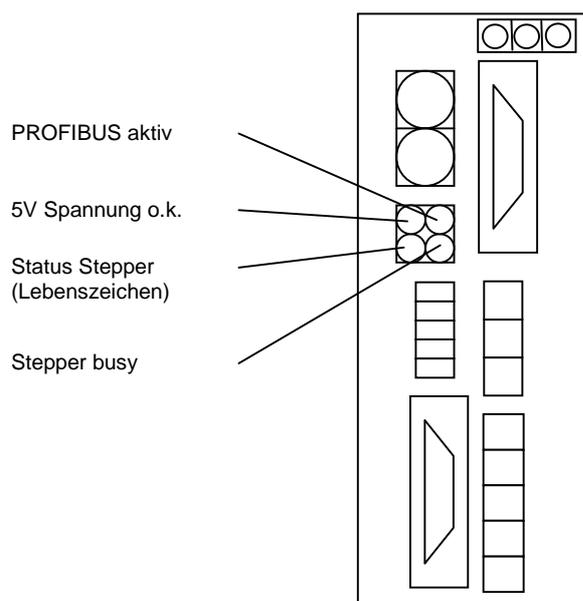
Beim Stepper wurde eine Möglichkeit geschaffen, den Alarmzustand auch ohne Modulabschaltung zu beenden. Dazu wurde ein **Pseudocode** für die PBS-Ausgangsbytes 1+2 (Geschwindigkeitscode) eingeführt. Es ist der Code **0xAA55**. Dieser Code kommt im normalen Betrieb nicht vor, da die max. Geschwindigkeit mit 0x1FF codiert wird. Liest der Stepper den Pseudocode, so prüft er **erneut**, ob die Bedingungen für den Alarmzustand noch vorhanden sind. Sind die Bedingungen nicht mehr vorhanden, schaltet er den Alarmzustand ab und kehrt **nach Beendigung** des Pseudocodes wieder in den Betriebszustand zurück. Die Diagnose-LED beginnt erneut zu blinken und zeigt somit den Betriebszustand an. Zu beachten ist hierbei, dass die Pseudocodeausgabe 0xAA55 auch tatsächlich **verlassen** wird, denn erst dann wird der Betriebszustand wieder eingenommen.

! Aktivieren/Deaktivieren der Steuerschalter !

Durch Eintragen des Wortes 8001h in den Geschwindigkeitscode werden die Steuerschalter aktiviert. Durch Eintragen von 8000h werden die Steuerschalter deaktiviert.

Standardeinstellung: Steuerschalter aktiviert.

Bedeutung der Status LED's:



6. Programmierung des PBS Schrittmotor-Controllers

Der PBS-Schrittmotor-Controller wird über den PROFIBUS DP mit acht Eingangsbytes und acht Ausgangsbytes gesteuert und kontrolliert. Die E/A-Bytes sind folgendermaßen belegt:

12 Ausgangsbytes:

Byte-Nr.	Bedeutung
1+2	Geschwindigkeitscode Soll-Geschwindigkeit, bei Normalfahrt , Soll-Endgeschwindigkeit, bei Beschleunigungsfahrt . Byte 1: High-Byte; Byte 2:Low-Byte
3+4	Startgeschwindigkeit für Beschleunigungsfahrten Byte 3: High-Byte; Byte 4:Low-Byte
5+6	Beschleunigungscode, bestehend aus Zeitbasis und Werteteilung Byte 5: Zeitbasis; Byte 6: Werteteilung
7+8	Kommandowort Byte7: High-Byte; Byte 8: Low- Byte
9+10	Zielpositionsvorgabe low-Wort Byte 9: High-Byte; Byte 10:Low-Byte
11+12	Zielpositionsvorgabe high-Wort Byte 11: High-Byte; Byte 12:Low-Byte

Die Zeitbasis im Beschleunigungscode gibt an, um das Wievielfache 1ms auf der Frequenzstufe während der Rampenzeit verweilt werden soll. Die Werteteilung gibt an, jeder wievielte Wert aus der Tabelle ab Seite 16 für die Generierung der Rampen benutzt werden soll. Eine detaillierte Beschreibung dieser beiden Parameter finden Sie auf Seite 14.

8 Eingangsbytes:

Byte-Nr.	Bedeutung
1+2	Anzeige der aktuellen Geschwindigkeit Byte 1: High-Byte; Byte 2: Low-Byte
3+4	Statusbytes Byte 3: High-Byte; Byte 4:Low-Byte
5+6	Ist-Position-low-Wort Byte 5: High-Byte; Byte 6:Low-Byte
7+8	Ist-Position-high-Wort Byte 7: High-Byte; Byte 8:Low-Byte

6.1 Aufbau des Kommandowortes und Beschleunigungscode

Um den Schrittmotor zu einer Aktion zu veranlassen, müssen entsprechende Bits im **Kommandowort** von der Steuerung gesetzt werden. Es werden neun Bits zur Übergabe von Kommandos an den Schrittmotor-Controller übergeben. Die Bits des Kommandowortes haben folgende Bedeutung:

Bits	Byte	Bedeutung	Wertebereich
00	8	Reserve	
01		Counter Reset	0 = Counter aktiv 1 = Counter Reset (auf 8000 0000h)
02		Beschleunigung ein/aus	0 = Beschleunigung aus (Normalfahrt) 1 = Beschleunigung ein (Beschleunigungsfahrt)
03		Velocity mode	0 = position mode 1 = velocity mode
04		Motor Start/Stop	0 = Stop Motor/ Kommando initialisieren 1 = Start Motor
05		Fahrtrichtung	0 = vorwärts 1 = rückwärts
06		Enable Ausgang	0 = off 1 = on
07		Referenzfahrt	0 = Normalbetrieb 1 = Referenzfahrt
08	7	Vorzugsdrehrichtung	0 = Vorzugsdrehrichtung entsprechend Bit 05 1 = Vorzugsdrehrichtung entsprechend invertiertem Bit 05
09		Polarität Stoppschalter	0 = normal/Öffner 1 = negiert/Schließer
10...15		Reserve	

Bit	Beschreibung
1	Wenn dieses Bit auf 1 gesetzt wird, wird der Schrittzähler in den Eingangsbytes 5-8 auf 8000 0000h zurückgesetzt. Für den normalen Positionierbetrieb muss dieses Bit immer auf 0 gesetzt werden. Solange das Bit gesetzt ist, wirkt die Freigabe/Bit4 nicht
2	Wenn dieses Bit auf 1 gesetzt ist, wird der Beschleunigungscode des Stepper aktiviert, d.h. die angewählte Endgeschwindigkeit wird nicht sofort, sondern entsprechend der Vorgabe im Beschleunigungscode verzögernd erreicht. Ist dieses Bit auf 0 gesetzt, wird die angewählte Schrittgeschwindigkeit sofort ausgegeben.
3	Wenn dieses Bit auf 1 gesetzt ist, wird der Stepper im Velocity Mode als ungesteuerte Drehzahlvorgabe betrieben. Es werden keine Endschalter, Stoppschalter und Positionsvorgaben ausgewertet. Im normalen Positionierbetrieb ist dieses Bit auf 0 zu setzen
4	Mit diesem Bit kann ein laufender Fahrbefehl gestoppt werden, indem es auf 0 gesetzt wird. Mit dem Wechsel von 0 auf 1 wird der Motor freigegeben und der entsprechende Fahrbefehl gestartet. Nach Ablauf des Fahrbefehls muss dieses Bit wieder auf 0 gesetzt werden.
5	Mit diesem Bit wird die Fahrtrichtung, d.h. der direction Ausgang des Stepper gesteuert. Dabei ist Bit8 gemäß der Vorzugsdrehrichtung zu berücksichtigen
6	Dieses Bit wird direkt auf den Enable Ausgang des Stepper übertragen und kann zur Freischaltung einer Schrittmotorbaugruppe verwendet werden.
7	Wenn dieses Bit auf 1 gesetzt ist, wird der Normalbetrieb des Stepper deaktiviert und eine Referenzfahrt gestartet. Die Referenzfahrt ist im Kapitel 8 genau beschrieben.
8	Es wird festgelegt, ob Bit 5 invertiert betrachtet wird oder nicht. Bei Bit8=0 gilt für Bit5=0 rückwärts und Bit5=1 vorwärts. Umgekehrtes gilt bei Bit 8=1.
9	Mit diesem Bit wird die Polarität des Stoppeingangs festgelegt, 0= Öffner, 1= Schließer

Detallierter Aufbau des Beschleunigungs-codes

Bit15... Beschleunigungs-Zeitbasis D7	Bit07... Beschleunigungs-Werteteilung D7
Bit14... Beschleunigungs-Zeitbasis D6	Bit06... Beschleunigungs-Werteteilung D6
Bit13... Beschleunigungs-Zeitbasis D5	Bit05... Beschleunigungs-Werteteilung D5
Bit12... Beschleunigungs-Zeitbasis D4	Bit04... Beschleunigungs-Werteteilung D4
Bit11... Beschleunigungs-Zeitbasis D3	Bit03... Beschleunigungs-Werteteilung D3
Bit10... Beschleunigungs-Zeitbasis D2	Bit02... Beschleunigungs-Werteteilung D2
Bit09... Beschleunigungs-Zeitbasis D1	Bit01... Beschleunigungs-Werteteilung D1
Bit08... Beschleunigungs-Zeitbasis D0	Bit00... Beschleunigungs-Werteteilung D0

Alle Angaben der Anfangs- und Endgeschwindigkeit in den Beispielen beziehen sich auf die Codierung gemäß Tabelle ab Seite 16:

Beispiel1

- Vorgabe:
- Anfangsgeschwindigkeit: 0
 - Endgeschwindigkeit: 535
 - Beschl.-Zeitbasis: 1
 - Beschl.-Werteteilung: 10
- Berechnung:
- interne Indextabelle:
 - 0,10,20,30,40,50,60,70,80,90,100,110,120,130,140,150,160,170,180,
 - 190,200,210,220,230,240,250,260,270,280,290,300,310,320,330,
 - 340,350,360,370,380,390,400,410,420,430,440,450,460,470,480,
 - 490,500,510,520, 520,530,535
 - Gesamtzeit bis zum Erreichen der Endgeschwindigkeit: 54ms

Beispiel2

- Vorgabe:
- Anfangsgeschwindigkeit: 10
 - Endgeschwindigkeit: 50
 - Beschleunigung-Zeitbasis: 5
 - Beschleunigung-Werteteilung: 1
- Berechnung:
- interne Indextabelle:
 - 10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,
 - 31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50
 - Gesamtzeit bis zum Erreichen der Endgeschwindigkeit: 200ms

Beispiel3

- Vorgabe:
- Anfangsgeschwindigkeit: 10
 - Endgeschwindigkeit: 535
 - Beschleunigung-Zeitbasis: 1
 - Beschleunigung-Werteteilung: 52
- Berechnung:
- interne Indextabelle:
 - 10,63,116,169,222,275,328,381,434,487,535
 - Gesamtzeit bis zum Erreichen

6.2 Aufbau der Statuswortes

Um den derzeitigen Status des Schrittmotor-Controllers in der Steuerung einzulesen, werden die Statusbytes (**Eingangsbyte 3+4**) des Controllers gelesen. Die untenstehende Tabelle zeigt, welche Bits innerhalb der Statusbytes ausgewertet werden können, um den aktuellen Status des Controllers zu ermitteln.

Die Bits 0-7 = Eingangsbyte 3 und 8-15 = Eingangsbyte 4 haben folgende Bedeutung:

Bits	Byte	Bedeutung	Wertebereich
0	4	Status Stoppschalter	0 = Stoppschalter nicht aktiv 1 = Stoppschalter ausgelöst
1		Status unterer Endschalter	0 = Endschalter nicht aktiv 1 = Endschalter ausgelöst
2		Status Referenzpunktsch.	0 = Referenzpunktschalter ist offen 1 = Referenzpunktschalter ist geschlossen
3		Status oberer Endschalter	0 = Endschalter nicht aktiv 1 = Endschalter ausgelöst
4		Aktuelle Fahrtrichtung	0 = vorwärts (COUNT wird incr.) 1 = rückwärts (COUNT wird decr.)
5		Status Endgeschwindigkeit	0 = Endgeschwindigkeit noch nicht erreicht 1 = Endgeschwindigkeit erreicht
6		nicht belegt	
7		Busy Anzeige	0 = Step-Ausgang steht (Busy) 1 = Step-Ausgang läuft
8	3	Status Endschalter	0 = kein Endschalter erreicht 1 = ein Endschalter erreicht
9		logischer Pegel oberer Endschalter	
10		logischer Pegel unterer Endschalter	
11		Notstopp/Alarm	0 = Alles ok 1 = Alarm bzw. Notstop aktiv
12		Zählerstatus	0 = Daten sind ungültig 1 = Daten vom Zähler sind gültig
13		nicht belegt	
14		Modul-Error	0 = Modul ok 1 = Modul-Error
15		nicht belegt	

6.3 Vorgabe des Geschwindigkeitscodes

In den **ersten beiden Ausgangsbytes** ist der Geschwindigkeitscode enthalten. Er hat einen **Wertevorrat** von **000h...32Bh**. Zur Interpretation des Geschwindigkeitscodes benutzt der Stepper eine intern abgelegte Tabelle. Sie enthält 811 Integer-Werte, die die direkten Timerkonstanten für den Step-Takt verschlüsseln. Von einem Geschwindigkeitswert zum nächsten beträgt die Differenz 15Hz unterhalb 10kHz, oberhalb 10 kHz beträgt die Differenz 30Hz-Schritte. Bedingt durch die Struktur des Timers im Prozessor verläuft diese Abhängigkeit nicht völlig linear. Die genauen Werte können der folgenden Tabelle entnommen werden. Für spezielle Applikationen ist es möglich, eine neue Code-Tabelle zu erstellen und im Prozessor abzulegen. Die Tabelle gilt auch für die Startgeschwindigkeit.

Code	Frequenz(Hz)
0	30
1	45
2	60
3	75
4	90
5	105
6	120
7	135
8	150
9	165
A	180
B	195
C	210
D	225
E	240
F	255
10	270
11	285
12	300
13	315
14	330
15	345
16	360
17	375
18	390
19	405
1A	420
1B	435
1C	450
1D	465
1E	480
1F	495
20	510
21	525
22	540
23	555
24	570
25	585
26	600
27	615
28	630
29	645
2A	660
2B	675
2C	690
2D	705
2E	720
2F	735
30	750
31	765
32	780
33	795
34	810

Code	Frequenz(Hz)
35	825
36	840
37	855
38	870
39	885
3A	900
3B	915
3C	930
3D	945
3E	960
3F	975
40	990
41	1005
42	1020
43	1035
44	1050
45	1065
46	1080
47	1095
48	1110
49	1125
4A	1140
4B	1155
4C	1170
4D	1185
4E	1200
4F	1215
50	1230
51	1245
52	1260
53	1275
54	1290
55	1305
56	1320
57	1335
58	1350
59	1365
5A	1380
5B	1395
5C	1410
5D	1425
5E	1440
5F	1455
60	1470
61	1485
62	1500
63	1515
64	1530
65	1545
66	1560
67	1575
68	1590
69	1605

Code	Frequenz(Hz)
6A	1620
6B	1635
6C	1650
6D	1665
6E	1680
6F	1695
70	1710
71	1725
72	1740
73	1755
74	1770
75	1785
76	1800
77	1815
78	1830
79	1845
7A	1860
7B	1875
7C	1890
7D	1905
7E	1920
7F	1935
80	1950
81	1965
82	1980
83	1995
84	2010
85	2025
86	2040
87	2055
88	2070
89	2085
8A	2100
8B	2115
8C	2130
8D	2145
8E	2160
8F	2175
90	2190
91	2205
92	2220
93	2235
94	2250
95	2265
96	2280
97	2295
98	2310
99	2325
9A	2340
9B	2355
9C	2370
9D	2385
9E	2400

Code	Frequenz(Hz)
9F	2415
A0	2430
A1	2445
A2	2460
A3	2475
A4	2490
A5	2505
A6	2521
A7	2536
A8	2551
A9	2566
AA	2581
AB	2596
AC	2612
AD	2628
AE	2643
AF	2658
B0	2673
B1	2688
B2	2703
B3	2718
B4	2733
B5	2748
B6	2763
B7	2779
B8	2794
B9	2810
BA	2826
BB	2842
BC	2857
BD	2872
BE	2887
BF	2902
C0	2918
C1	2933
C2	2948
C3	2964
C4	2979
C5	2994
C6	3009
C7	3024
C8	3039
C9	3054
CA	3069
CB	3084
CC	3099
CD	3115
CE	3131
CF	3146
D0	3161
D1	3176
D2	3191
D3	3206

Code	Frequenz(Hz)
D4	3222
D5	3237
D6	3252
D7	3267
D8	3282
D9	3298
DA	3313
DB	3329
DC	3344
DD	3359
DE	3374
DF	3389
E0	3405
E1	3420
E2	3436
E3	3452
E4	3468
E5	3484
E6	3500
E7	3516
E8	3531
E9	3546
EA	3562
EB	3577
EC	3592
ED	3607
EE	3623
EF	3638
F0	3654
F1	3669
F2	3685
F3	3701
F4	3717
F5	3733
F6	3750
F7	3766
F8	3783
F9	3799
FA	3814
FB	3831
FC	3846
FD	3861
FE	3878
FF	3893
100	3908
101	3924
102	3939
103	3955
104	3970
105	3986
106	4002
107	4018
108	4034

Code	Frequenz(Hz)								
109	4051	14C	5145	18F	6309	1D2	7481	215	8982
10A	4067	14D	5163	190	6329	1D3	7500	216	9009
10B	4084	14E	5181	191	6349	1D4	7518	217	9036
10C	4101	14F	5199	192	6369	1D5	7537	218	9063
10D	4118	150	5217	193	6389	1D6	7556	219	9090
10E	4135	151	5235	194	6410	1D7	7575	21A	9118
10F	4152	152	5253	195	6430	1D8	7594	21B	9146
110	4169	153	5272	196	6451	1D9	7614	21C	9174
111	4184	154	5291	197	6472	1DA	7633	21D	9202
112	4201	155	5309	198	6493	1DB	7653	21E	9230
113	4216	156	5328	199	6514	1DC	7672	21F	9259
114	4231	157	5347	19A	6535	1DD	7692	220	9287
115	4246	158	5366	19B	6550	1DE	7712	221	9302
116	4261	159	5381	19C	6571	1DF	7731	222	9331
117	4276	15A	5400	19D	6586	1E0	7751	223	9360
118	4291	15B	5415	19E	6607	1E1	7772	224	9375
119	4307	15C	5434	19F	6622	1E2	7792	225	9404
11A	4322	15D	5449	1A0	6637	1E3	7812	226	9419
11B	4338	15E	5464	1A1	6659	1E4	7832	227	9448
11C	4354	15F	5479	1A2	6674	1E5	7853	228	9463
11D	4369	160	5494	1A3	6696	1E6	7874	229	9478
11E	4385	161	5509	1A4	6711	1E7	7894	22A	9493
11F	4402	162	5524	1A5	6726	1E8	7915	22B	9508
120	4418	163	5540	1A6	6741	1E9	7936	22C	9523
121	4434	164	5555	1A7	6756	1EA	7957	22D	9538
122	4451	165	5571	1A8	6772	1EB	7978	22E	9554
123	4467	166	5586	1A9	6787	1EC	8000	22F	9569
124	4484	167	5602	1AA	6802	1ED	8021	230	9584
125	4501	168	5617	1AB	6818	1EE	8042	231	9600
126	4518	169	5633	1AC	6833	1EF	8064	232	9615
127	4535	16A	5649	1AD	6849	1F0	8086	233	9630
128	4552	16B	5665	1AE	6864	1F1	8108	234	9646
129	4569	16C	5681	1AF	6880	1F2	8130	235	9661
12A	4587	16D	5698	1B0	6896	1F3	8152	236	9677
12B	4604	16E	5714	1B1	6912	1F4	8174	237	9693
12C	4622	16F	5730	1B2	6928	1F5	8196	238	9708
12D	4640	170	5747	1B3	6944	1F6	8219	239	9724
12E	4658	171	5763	1B4	6960	1F7	8241	23A	9740
12F	4676	172	5780	1B5	6976	1F8	8264	23B	9756
130	4691	173	5797	1B6	6993	1F9	8287	23C	9771
131	4709	174	5813	1B7	7009	1FA	8310	23D	9787
132	4724	175	5830	1B8	7025	1FB	8333	23E	9803
133	4739	176	5847	1B9	7042	1FC	8356	23F	9819
134	4754	177	5865	1BA	7058	1FD	8379	240	9836
135	4769	178	5882	1BB	7075	1FE	8403	241	9852
136	4784	179	5899	1BC	7092	1FF	8426	242	9868
137	4800	17A	5917	1BD	7109	200	8450	243	9884
138	4815	17B	5934	1BE	7125	201	8474	244	9900
139	4830	17C	5952	1BF	7142	202	8498	245	9917
13A	4846	17D	5970	1C0	7159	203	8522	246	9933
13B	4862	17E	5988	1C1	7177	204	8547	247	9950
13C	4878	17F	6006	1C2	7194	205	8571	248	9966
13D	4893	180	6024	1C3	7211	206	8595	249	9983
13E	4909	181	6042	1C4	7228	207	8620	24A	10000
13F	4926	182	6060	1C5	7246	208	8645	24B	10016
140	4942	183	6079	1C6	7263	209	8670	24C	10050
141	4958	184	6097	1C7	7281	20A	8695	24D	10084
142	4975	185	6116	1C8	7299	20B	8720	24E	10118
143	4991	186	6134	1C9	7317	20C	8746	24F	10152
144	5008	187	6153	1CA	7334	20D	8771	250	10186
145	5025	188	6172	1CB	7352	20E	8797	251	10221
146	5042	189	6191	1CC	7371	20F	8823	252	10256
147	5059	18A	6211	1CD	7389	210	8849	253	10291
148	5076	18B	6230	1CE	7407	211	8875	254	10327
149	5093	18C	6250	1CF	7425	212	8902	255	10362
14A	5110	18D	6269	1D0	7444	213	8928	256	10398
14B	5128	18E	6289	1D1	7462	214	8955	257	10434

Code	Frequenz(Hz)
258	10471
259	10507
25A	10544
25B	10582
25C	10619
25D	10657
25E	10695
25F	10733
260	10771
261	10810
262	10849
263	10889
264	10928
265	10968
266	11009
267	11049
268	11090
269	11131
26A	11173
26B	11214
26C	11257
26D	11299
26E	11342
26F	11385
270	11428
271	11472
272	11516
273	11560
274	11605
275	11650
276	11695
277	11741
278	11787
279	11834
27A	11881
27B	11928
27C	11976
27D	12024
27E	12072
27F	12121
280	12170
281	12219
282	12269

Code	Frequenz(Hz)
283	12320
284	12371
285	12422
286	12474
287	12526
288	12578
289	12631
28A	12684
28B	12738
28C	12793
28D	12847
28E	12903
28F	12958
290	13015
291	13071
292	13129
293	13186
294	13245
295	13303
296	13363
297	13422
298	13482
299	13542
29A	13603
29B	13667
29C	13733
29D	13800
29E	13868
29F	13936
2A0	14005
2A1	14075
2A2	14146
2A3	14218
2A4	14291
2A5	14365
2A6	14440
2A7	14516
2A8	14593
2A9	14671
2AA	14750
2AB	14830
2AC	14911
2AD	15000

Code	Frequenz(Hz)
2AE	14117
2AF	14150
2B0	14184
2B1	14218
2B2	14251
2B3	14285
2B4	14319
2B5	14354
2B6	14388
2B7	14423
2B8	14457
2B9	14492
2BA	14527
2BB	14563
2BC	14598
2BD	14634
2BE	14669
2BF	14705
2C0	14742
2C1	14778
2C2	14814
2C3	14851
2C4	14888
2C5	14925
2C6	14962
2C7	15000
2C8	15037
2C9	15075
2CA	15113
2CB	15151
2CC	15189
2CD	15228
2CE	15267
2CF	15306
2D0	15345
2D1	15384
2D2	15424
2D3	15463
2D4	15503
2D5	15544
2D6	15584
2D7	15625
2D8	15665

Code	Frequenz(Hz)
2D9	15706
2DA	15748
2DB	15789
2DC	15831
2DD	15873
2DE	15915
2DF	15957
2E0	16000
2E1	16042
2E2	16085
2E3	16129
2E4	16172
2E5	16216
2E6	16260
2E7	16304
2E8	16348
2E9	16393
2EA	16438
2EB	16483
2EC	16528
2ED	16574
2EE	16620
2EF	16666
2F0	16713
2F1	16759
2F2	16806
2F3	16853
2F4	16901
2F5	16949
2F6	16997
2F7	17045
2F8	17094
2F9	17142
2FA	17191
2FB	17241
2FC	17291
2FD	17341
2FE	17391
2FF	17441
300	17492
301	17543
302	17595
303	17647

Code	Frequenz(Hz)
304	17699
305	17751
306	17804
307	17857
308	17910
309	17964
30A	18018
30B	18072
30C	18126
30D	18181
30E	18237
30F	18292
310	18348
311	18404
312	18461
313	18518
314	18575
315	18633
316	18691
317	18750
318	18808
319	18867
31A	18927
31B	18987
31C	19047
31D	19108
31E	19169
31F	19230
320	19292
321	19354
322	19417
323	19480
324	19543
325	19607
326	19672
327	19736
328	19801
329	19867
32A	19933
32B	20000

6.4 Positionierung

Der Stepper verwendet die **absolute** Positionierung. Diese übersichtlichere Positionierungsart wurde möglich, da der Wertevorrat des Zählers nunmehr eine unsigned long Variable ist. Das bedeutet, es stehen jetzt **0xFFFFFFFF Zählschritte** als Wertebereich zur Verfügung. Absolute Positionierung bedeutet, dass die Zählerstellung 0 der Position 0 entspricht und die Zählerstellung 0xFFFFFFFF der Position 0xFFFFFFFF.

Erst wenn eine dieser beiden Positionen um einen Step überschritten wird, spricht der untere bzw. obere Softwareendschalter an und schaltet den Antrieb ab. Eine Ausnahme stellt der **Velocity-Mode** dar, bei dem weder Hardware- noch Software-Endschalter ansprechen. Das Erreichen eines Software-Endschalters im **Normalmode** zeigt einen Fehler an, denn die tatsächlichen Endpositionen einer realen Anwendung sind durch die Hardware-Endschalter markiert.

Zur Orientierung bei der absoluten Positionierung dient grundsätzlich der Referenzpunkt. Der **Referenzpunkt** wird durch die Position **0x80000000** codiert. Dieser Wert wurde festgelegt, um für den Bereich links und rechts vom Referenzpunkt aus, genügend Wertevorrat zur Positionierung zu haben. Je nachdem welches Zahlenformat der Anwender benutzt, kann der Positionswert leicht umgerechnet werden. Der verwendete unsigned long Wert stellt für die Konvertierung in unterschiedliche kundenspezifische Zahlenformate eine günstige Basis dar. Nach diesem Prinzip wird auch bei analogen Eingabegeräten in der PROFIBUS DP Welt und auch bei anderen Feldbussen verfahren.

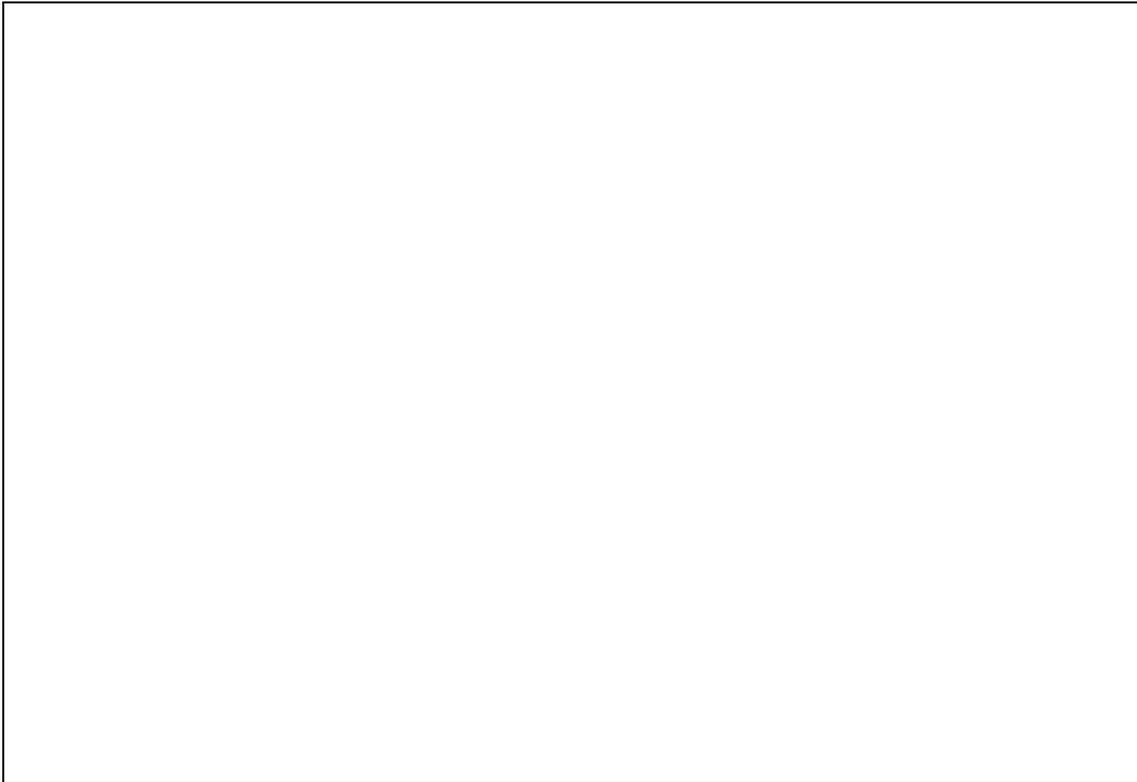
Die **Ist-Position** des Stepper wird in den **PBS-Eingangsbytes 5-8** dargestellt. Die **Soll-Position** wird in **PBS-Ausgangsbytes 9-12** eingetragen. Wie schon erläutert, wird der Zähler nach Erreichen des Referenzpunktes bei der Referenzfahrt auf 0x80000000 gesetzt. Da **nach dem Einschalten** des Gerätes die Position undefiniert ist, wird der Zähler in diesem Zustand ebenfalls auf **0x80000000 gesetzt**, um dem Anwender die Möglichkeit zu eröffnen, in beide Richtungen zu fahren, ohne sofort auf einen Software-Endschalter zu stoßen. Bei Zählerstellung 0 nach Reset und Rückwärtsfahrt wäre z.B. sofort der untere Software-Endschalter erreicht. Aus den gleichen Gründen wird auch der Befehl „**Counter Reset**“ so ausgeführt, dass der Zähler den Positionswert 0x80000000 erhält.

Durch die Abfrage des aktuellen Zählerstandes kann die Steuerung die momentane Position des Antriebs ermitteln und innerhalb des Anwenderprogrammes verwerten.

Zur Positionierung ist die Abfrage des aktuellen Zählerstandes durch die Steuerung nicht erforderlich, da hierzu die Zielpositionsvorgabe in den Ausgangsbytes 9-12 vorgesehen ist. Das Modul vergleicht bei jedem Step die Istposition mit der Zielposition und beendet den Stepvorgang, sobald beide übereinstimmen. Die Zielposition wird nicht um einen einzigen Step überfahren. Hierdurch werden lauffzeitbedingte Fehler bei der Positionierung vermieden.

Hat der Stepper durch Erreichen der Zielposition die Stepgenerierung beendet, wird das Anwenderprogramm eine neue Zielposition vorwählen. Bevor der Stepper nun **erneut** losläuft, **muss** jedoch das **Motor-Start-Bit** des **Kommandobytes**, das evtl. durch den vorhergehenden Lauf noch aktiv ist, **zuerst inaktiv** und dann **wieder aktiv** gesetzt werden. Diese zusätzliche Sicherheit ist nützlich, da der Stepper sonst bei Eintragungen im niederwertigen Teil der Zielposition sofort starten würde, obwohl der höherwertige Positionswert noch nicht geändert wurde. Die Zielposition läßt sich auch **während** des Laufes ändern. Sollte diese Funktion im Anwenderprogramm genutzt werden, ist darauf zu achten, dass die Zielposition **zwischen zwei PBS-Datenzyklen** angepasst wird, um diese konsistent zu übertragen, denn der Antrieb läuft während der Änderung weiter.

Zu beachten ist weiterhin, dass der Stepper bei inaktivem Beschleunigungsmodus den Fahrtrichtungsausgang selbstständig schaltet. Das entsprechende Kommandobit ist daher unwirksam. Die vom Stepper gewählte Fahrtrichtung resultiert stets aus dem Vergleich von Ist- und Zielposition. Das Kommandobit Fahrtrichtung ist nur im Velocity-Mode wirksam, da in diesem Mode kein Positionsvergleich stattfindet.



6.5 Nutzung der Positionsvorgabe

Die **Stoppbedingungen** werden während der Fahrt ständig überprüft. Die Hardware-Bedingungen (Stopp-, End-, Notstoppschalter) haben natürlich eine höhere Priorität, d.h. wenn der vorgegebene Positionswert noch nicht erreicht ist und ein Schalter betätigt wird, wird der Motor sofort gestoppt.

Die Positionsvorgabe wird während der **Referenzfahrt** nicht ausgewertet, da die Referenzfahrt vom STEPPER eigenständig gesteuert wird.

Auch im **Velocity-Mode** wird die Positionsvorgabe nicht ausgewertet.

Die Positionsvorgabe kann während der Fahrt im Normalmode geändert werden. Hierbei ist jedoch zu beachten, dass die Positionsvorgabe konsistent geändert werden muss, da der Stepper den übergebenen Positionswert sofort in den Positionsvergleich übernimmt. Die geforderte Datenkonsistenz erreicht man, indem die Positionsbytes zwischen zwei PBS Datenzyklen geändert werden. Falls dies aufgrund der Hardwaregegebenheiten des verwendeten Masters nicht möglich ist, sollte auf eine Positionsänderung während der Fahrt verzichtet werden. Ein nicht konsistent übertragener Positionswert kann den Stepper zu einem ungewolltem Stopp oder bei ungünstigen Positionswerten sogar zur Änderung der Fahrtrichtung veranlassen.

Da der Stepper nach dem Erreichen der Zielposition stoppt, kann dann die nächste Zielposition ohne Konsistenzprobleme angewählt werden. Gleiches gilt für Endschalter-Stopps oder Stopps durch Abschalten der Motorfreigabe. In diesen Fällen kann die Zielpositionsvorgabe auch über mehrere PBS-Datenzyklen verteilt eingegeben werden, wenn das Motor-Start-Kommando erst nach der vollständigen Zielpositionsvorgabe für die nächste Fahrt erneut aktiviert wird.

Bei Endschalter- oder Alarm Stopps oder auch bei Stopps durch Erreichen der Zielposition, erfolgt der nächste **Motor-Start erst**, wenn das **Motor-Start-Bit** zuerst **rückgesetzt** und dann **wieder** gesetzt wird. Somit ist die Datenkonsistenz nach einem Stopp vollständig gewährleistet.

7. Kommandobeispiele

Kommandobeispiel 1: mittlere Fahrt zurück

Nach Ausgabe der im Folgenden beschriebenen Kommandos fährt der Stepper von der Ist-Position 8000 0000h mit der Taktfrequenz 495Hz zur Zielposition 7FFF0BB8h und bleibt dort stehen, Startfrequenz soll 0Hz sein. Der Zähler steht nach dem Einschalten des Gerätes oder nach Counter-Reset oder nach erfolgreicher Referenzfahrt auf der Istposition 8000 0000 h. Zu beachten ist, dass bei den Einstellungen der PBS-OUT-Bytes, das Motor-Start-Bit zuletzt freigegeben wird.

Tempo	01Fh -> 495Hz
Pos.-Istwert	8000 0000 h
Pos.Vorgabe	7FFF 0BB8 h
Stop	Öffner (nicht aktiv)
oberer Endsch.	geschlossen (nicht aktiv)
unterer Endsch.	geschlossen (nicht aktiv)
Richtung	Rückwärtsfahrt
Betriebsart	Normalbetrieb
Freigabe Motor	Ja (Motor ist freigegeben)

Daraus ergibt sich folgende Belegung der PBS-OUT Bytes:

Geschwindigkeits -Code	PBS-OUT Bytes 1+2	001Fh	0000 0000 0001 1111
Startgeschwindigkeit-Code	PBS-OUT Bytes 3+4	0000h	0000 0000 0000 0000
Beschleunigungs-Code	PBS-OUT Bytes 5+6	0000h	0000 0000 0000 0000
Pos.vorgabe , low-Wort	PBS-OUT Bytes 9+10	0BB8h	0000 1011 1011 1000
Pos.vorgabe , high-Wort	PBS-OUT Bytes 11+12	7FFFh	0111 1111 1111 1111

Nach Auftrags-Ende zeigen die PBS-IN Bytes folgende Belegung:

Aktuelle Geschwindigkeit	PBS-IN Bytes 1+2	001Fh	0000 0000 0001 1111
Statusbytes	PBS-IN Bytes 3+4	0010h	0000 0000 0111 0000
Low-Wort Positionswert	PBS-IN Bytes 5+6	0BB8h	0000 1011 1011 1000
High-Wort Positionswert	PBS-IN Bytes 7+8	7FFFh	0111 1111 1111 1111

Kommandoispiel 2: Beschleunigungsfahrt vorwärts

Nach Ausgabe der im Folgenden beschriebenen Kommandos fährt der Stepper von der Ist-Position 8000 0000h mit der Startgeschwindigkeit 0 im Beschleunigungsmodus vorwärts. Die Beschleunigungszeit beträgt 147ms, die Endgeschwindigkeit ist 16kHz. . Bei Erreichen der Zielposition bleibt der Antrieb stehen. Der Zähler steht nach dem Einschalten des Gerätes oder nach Counter-Reset oder nach erfolgreicher Referenzfahrt auf der Istposition 8000 0000 h. Zu beachten ist, dass bei den Einstellungen der OUT-Bytes, das Motor-Start-Bit zuletzt freigegeben wird.

Endgeschwindigkeit	02E0h -> 16000Hz
Pos.-Istwert	8000 0000 h
Pos.Vorgabe	8001 0000 h
Stopp	Öffner (nicht aktiv)
oberer Endsch.	geschlossen (nicht aktiv)
unterer Endsch.	geschlossen (nicht aktiv)
Richtung	Vorwärtsfahrt
Betriebsart	Beschleunigungsfahrt
Beschleunigungscode	0105 -> Zeitbasis 1ms mit jedem fünften Wert aus der Tabelle
Freigabe Motor	Ja (Motor ist freigegeben)

Daraus ergibt sich folgende Belegung der PBS-OUT Bytes:

Geschwindigkeits -Code	PBS-OUT Bytes 1+2	02E0h	0000 0010 1110 0000
Startgeschwindigkeit	PBS-OUT Bytes 3+4	0000h	0000 0000 0000 0000
Beschl. Code	PBS-OUT Bytes 5+6	0105h	0000 0001 0000 0101
Kommandowort	PBS-OUT Bytes 7+8	0054h	0000 0000 0101 0100
Zielposition low	PBS-OUT Bytes 9+10	0000h	0000 0000 0000 0000
Zielposition high	PBS-OUT Bytes 11+12	8001h	1000 0000 0000 0001

Nach Auftrags-Ende zeigen die PBS-IN Bytes folgende Belegung:

Aktuelle Geschw	PBS-IN Bytes 1+2	0000h	0000 0000 0000 0000
Statusbytes	PBS-IN Bytes 3+4	0000h	0000 0000 0000 0000
Low-Wort Positionswert	PBS-IN Bytes 5+6	0000h	0000 0000 0000 0000
High-Wort Positionswert	PBS-IN Bytes 7+8	8001h	1000 0000 0000 0001

Kommandoispiel 3: Taktausgabe im Velocity-Mode

Der Velocity-Mode stellt eine Besonderheit in der Funktion des STEPPER dar. In diesem Modus arbeitet das Modul als Drehzahlsteuerung, die über den PROFIBUS DP entsprechend der vorgegebenen Geschwindigkeitskurve verändert werden kann. Hardware-Endschalter, Software-Endschalter, Stoppschalter und Zielposition werden in diesem Mode nicht ausgewertet. Der Notstopp-Schalter (beide Endschalter gleichzeitig aktiv) wird jedoch ausgewertet und führt bei Aktivierung zum sofortigen Stopp des Antriebes. Beschleunigungsfahrten sind möglich, der aktuelle Zählerstand wird gemeldet. Zu beachten ist, dass bei den Einstellungen der OUT Bytes, das Motor-Start-Bit zuletzt freigegeben wird.

Taktfrequenz	3Fh -> 975Hz
Pos.-Istwert	8000 0000 h
Pos.Vorgabe	xxxx xxxx h (nicht bewertet)
Stopp	Öffner (nicht aktiv)
oberer Endsch.	x (nicht bewertet)
unterer Endsch.	x (nicht bewertet)
Richtung	Vorwärtsfahrt
Betriebsart	Velocity-Mode, unbeschleunigt
Freigabe Motor	Ja (Motor ist freigegeben)

Daraus ergibt sich folgende Belegung der PBS-OUT Bytes:

Geschwindigkeits -Code	PBS-OUT Bytes 1+2	003Fh	0000 0000 0011 1111
Startgeschwindigkeit	PBS-OUT Bytes 3+4	xxxxh	xxxx xxxx xxxx xxxx
Beschleunigungscode	PBS-OUT Bytes 5+6	xxxxh	xxxx xxxx xxxx xxxx
Kommandowort	PBS-OUT Bytes 7+8	0048h	0000 0000 0100 1000
Pos.vorgabe , low-Wort	PBS-OUT Bytes 5+6	xxxxh	xxxx xxxx xxxx xxxx
Pos.vorgabe , high-Wort	PBS-OUT Bytes 7+8	xxxxh	xxxx xxxx xxxx xxxx

Nach Auftrags-Ende zeigen die PBS-IN Bytes folgende Belegung:

Aktuelle Geschw	PBS-IN Bytes 1+2	003Fh	0000 0000 0011 1111
Statusbytes	PBS-IN Bytes 3+4	00A0h	0000 0000 1010 0000
Low-Wort Positionswert	PBS-IN Bytes 5+6	xxxxh	xxxx xxxx xxxx xxxx
High-Wort Positionswert	PBS-IN Bytes 7+8	xxxxh	xxxx xxxx xxxx xxxx

Beispiel: Software-Stopp bei Vorwärtsfahrt

Tempo	03
Stopp	Schließer (nicht aktiv)
oberer Endsch.	geschlossen (nicht aktiv)
unterer Endsch.	geschlossen (nicht aktiv)
Richtung	Vorwärtsfahrt
Betriebsart	Normalbetrieb
Freigabe Motor	Nein (Motor soll stoppen)

Daraus ergibt sich folgende Belegung der PBS-OUT Bytes:

Geschwindigkeits-Code	PBS-OUT Bytes 1+2	0003h	0000 0000 0000 0011
Kmd.-Wort	PBS-OUT Bytes 7+8	0040h	0000 0000 0100 0000
Pos.vorgabe , low-Wort	PBS-OUT Bytes 5+6	1000h	0001 0000 0000 0000
Pos.vorgabe , high-Wort	PBS-OUT Bytes 7+8	8FFFh	1000 1111 1111 1111

Nach Ausgabe des Befehls stoppt der Motor und es ergibt sich folgende Belegung der PBS-IN Bytes:

Aktuelle Geschw.	PBS-IN Bytes 1+2	0000h	0000 0000 0000 0000
Statusbytes	PBS-IN Bytes 3+4	0000h	0000 0000 0000 0000
Pos.-Istwert, low-Wort	PBS-IN Bytes 5+6	xxxxh	xxxx xxxx xxxx xxxx
Pos.-Istwert, high-Wort	PBS-IN Bytes 7+8	xxxxh	xxxx xxxx xxxx xxxx

Weitere Erkennungsmöglichkeiten des Motor-Stopp:

- Stoppschalter schließt und Stopp: Statusbytes1+2 = 0x1001
- oberer Endschalter öffnet und Stopp: Statusbytes1+2 = 0x1308
- Zählerüberlauf bei Fahrt vor und Stopp: Statusbytes1+2 = 0x1300

Beim Ansprechen eines Hardware-Endschalters oder eines Software-Endschalters, kann die Endschalterposition grundsätzlich durch Fahrt in Gegenrichtung verlassen werden.

8. Aufruf der Referenzfahrt

Die Referenzfahrt stellt eine Besonderheit bei den Kommandos an den Schrittmotor-Controller dar. Bei der Ausführung des Kommandos ist die Geschwindigkeit auf den Wert 0x20 fest eingestellt. Die Referenzfahrt wird immer durch Setzen des Kommandobits 7 (PC) gestartet.

Es gibt zwei Möglichkeiten die Referenzfahrt durchzuführen:

- Referenzfahrt **rückwärts** beginnend
- Referenzfahrt **vorwärts** beginnend

Dazu muss das Fahrtrichtungs-Bit (Bit 5) im Kommandobyte entsprechend gesetzt werden.

Es wird garantiert, dass beide Referenzfahrten am gleichen Eckpunkt des Endschalters enden. Wird während der beiden Referenzfahrten das Fahrtrichtungsbit umgeschaltet, wird der **Alarmzustand** eingenommen.

Wird im Bereich zwischen oberem und unterem Endschalter der Referenzschalter nicht gefunden, geht das Modul in den Alarmzustand, denn dann liegt ein **Hardwarefehler** vor. Beachten Sie, dass - bis zum erfolgreichen Abschluss der Referenzfahrt - die Ausführung weiterer Kommandos verweigert wird.

Nach der erfolgreichen Referenzfahrt steht der Antrieb genau auf dem Referenzpunkt und der Positionszähler steht auf dem Wert 0x80000000. Jetzt ist der Controller bereit zur Annahme von Kommandos und nimmt den normalen Betrieb auf. Die Referenzfahrt kann jederzeit von der Steuerung gestartet werden. Unterbrochen wird die Referenzfahrt nur durch den Not-Stopp, durch den Stopp-Schalter und bei Erreichen des unteren Endschalters, da dann davon ausgegangen werden kann, dass der Referenzschalter nicht gefunden wurde. Bei Not-Stopp und Erreichen des unteren Endschalters wird der Alarm-Zustand eingenommen, bei Ansprechen des Stopp-Schalters während der Referenzfahrt wird lediglich der Antrieb gestoppt und somit die Referenzfahrt abgebrochen. Alle Zustände können über die Statusbytes von der Steuerung erkannt und entsprechende Schritte eingeleitet werden.

Nach der Referenzfahrt steht der Antrieb genau am Referenzpunkt, der Positionszähler steht auf 0x80000000 und der Antrieb ist bereit für Fahrt vor oder rück, dies ist die Grundstellung. In den Statusbytes wird signalisiert, dass die Daten des Positionszählers ab jetzt gültig sind.

Zum genauen Verständnis der Schaltvorgänge während der Referenzfahrt soll folgende Übersicht dienen:

Abkürzungen:

GC = Geschwindigkeitscode

ES = Endschalter

RS = Referenzschalter

1. Referenzfahrt vorwärts, Schlitten zwischen Referenzpunkt und oberem Endschalter:

1.1	Positionszähler inkrementiert, GC wie vorgewählt. Referenzfahrt vorwärts beginnt.		
1.2	Unterer ES spricht an. Beim Start darf der untere ES aktiv sein.		
1.3	Oberer ES spricht an. Fahrtrichtung wechselt, Positionszähler dekrementiert. GC wie vorgewählt. Referenzpunkt befindet sich unterhalb der Anfangsposition.		
1.4	a) RS spricht an. Fahrtrichtung wechselt. Positionszähler inkrement., GC = 5 RS von oben getroffen, langsam zurück, bis RS wieder frei ist.	b) Unterer ES spricht an. Alarmzustand ! RS existiert nicht.	c) Oberer ES spricht an. Darf nach dem Richtungswechsel aktiv sein.
1.5	RS wechselt auf inaktiv. Fahrtrichtung wechselt, Positionszähler dekrementiert, GC = 5, langsames Heranfahren an RS.		
1.6	a) RS spricht an, Positionszähler auf 8000 0000h statisch, Referenzpunkt erreicht.	b) Unterer ES spricht an. Alarmzustand ! RS klemmt.	c) Oberer ES spricht an. Es läuft eine dekrementierende Fahrt.

2. **Referenzfahrt vorwärts**, Schlitten zwischen unterem Endschalter und Referenzschalter, Erläuterungen analog 1.

2.1	Start, Positionszähler inkrementiert, GC wie vorgewählt.		
2.2	RS spricht an, keine Veränderung		
2.3	RS wechselt auf inaktiv, Fahrtrichtung wechselt, Positionszähler dekrementiert, GC=5		
2.4	a) RS spricht an, Positionszähler auf 8000 0000h, Referenzpunkt erreicht	b) Unterer ES spricht an. Alarmzustand !	c) Oberer ES spricht an, keine Veränderung.

3. **Referenzfahrt vorwärts**, Richtung auf rückwärts umschalten während der Fahrt.

3.1	Start, Positionszähler inkrementiert, GC wie vorgewählt, Fahrt beginnt.
3.2	Fahrtrichtung auf rückwärts umschalten, Alarmzustand !, während der Fahrt darf die Fahrtrichtung nicht umgeschaltet werden.

4. **Referenzfahrt rückwärts**, Schlitten zwischen unterem Endschalter und Referenzschalter, Erläuterungen analog 1.

4.1	Positionszähler dekrementiert, GC wie vorgewählt. Referenzfahrt rückwärts beginnt.		
4.2	Oberer ES spricht an, keine Veränderung		
4.3	Unterer ES spricht an. Fahrtrichtung wechselt, Positionszähler inkrementiert. GC wie vorgewählt.		
4.4	RS spricht an, keine Veränderung		
4.5	RS wechselt auf inaktiv, Fahrtrichtung wechselt, Positionszähler dekrement., GC = 5		
4.6	a) RS spricht an. Positionszähler auf 8000 0000h statisch, Referenzpunkt erreicht.	b) Oberer ES spricht an, keine Veränderung.	c) Unterer ES spricht an, Alarmzustand !

3. **Referenzfahrt rückwärts**, Schlitten zwischen Referenzpunkt und oberem Endschalter:

5.1	Positionszähler dekrementiert, GC wie vorgewählt. Referenzfahrt rückwärts beginnt.		
5.2	a) RS spricht an. Fahrtrichtung wechselt. Positionszähler inkrement., GC = 5	b) Oberer ES spricht an, Alarmzustand !	c) Unterer ES spricht an, keine Veränderung.
5.3	RS wechselt auf inaktiv. Fahrtrichtung wechselt, Positionszähler dekrementiert, GC = 5, langsames Heranfahren an RS.		
5.4	a) RS spricht an, Positionszähler auf 8000 0000h statisch, Referenzpunkt erreicht.	b) Unterer ES spricht an, Alarmzustand !	c) Oberer ES spricht an, keine Veränderung.

6. Referenzfahrt rückwärts, Richtung auf vorwärts umschalten während der Fahrt.

6.1	Start, Positionszähler dekrementiert, GC wie vorgewählt, Fahrt beginnt.
6.2	Fahrtrichtung auf vorwärts umschalten, Alarmzustand !, während der Fahrt darf die Fahrtrichtung nicht umgeschaltet werden.

Fehlerzustände:

- Sobald der Referenzschalter das 1. mal erreicht wurde, wird bei allen darauf folgenden Vorwärtsfahrten der obere Endschalter als Referenzfehler bewertet und bei allen darauf folgenden Rückwärtsfahrten der untere Endschalter als Referenzfehler bewertet.
- Sobald der obere Endschalter das Umschalten in Rückwärtsfahrt bewirkt hat, wird der untere Endschalter als Referenzfehler bewertet.

Sonstiges:

- Bei der Referenzfahrt werden die Software-Endschalter ausgeblendet. Die Referenzfahrt kann also über den gesamten Wertebereich erfolgen.
- Bei der Referenzfahrt wird kein Positionsvergleich mit der Zielposition durchgeführt.
- Der EN-Ausgang bei der Referenzfahrt wird so geschaltet, wie er im Kommandobyte gewählt wurde.
- Das Statusbit „Endgeschwindigkeit erreicht“ wird während der Referenzfahrt gesetzt, solange mit der Geschwindigkeit des Geschwindigkeitscodes gefahren wird.
- Eine **neue Referenzfahrt** wird nur gestartet, wenn vorher das Kommando „**Counter Reset**“ gegeben wurde oder das Gerät neu eingeschalten wurde.

Beispiel für Referenzfahrt vorwärts beginnend:

Belegung der PBS-OUT Bytes:

Geschwindigkeits-Code	PBS-OUT Bytes 1+2	0005h	0000 0000 0000 0101
Kommandowort	PBS-OUT Bytes 7+8	00C0h	0000 0000 1010 0000
Pos.vorgabe , low-Wort	PBS-OUT Bytes 5+6	xxxxh	xxxx xxxx xxxx xxxx
Pos.vorgabe , high-Wort	PBS-OUT Bytes 7+8	xxxxh	xxxx xxxx xxxx xxxx

Nach Ausgabe des Befehls ergibt sich folgende Belegung der PBS-IN Bytes:

Aktuelle Geschw	PBS-IN Bytes 1+2	0005h	0000 0000 0000 0101
Statusbytes	PBS-IN Bytes 3+4	1000h	0001 0000 0100 0000
Pos.-Istwert, low-Wort	PBS-IN Bytes 5+6	0000h	0000 0000 0000 0000
Pos.-Istwert, high-Wort	PBS-IN Bytes 7+8	8000h	1000 0000 0000 0000

9. Die Notstopp-Funktion

Der STEPPER bietet die Möglichkeit, einen manuellen Notstopp für den Gefahren- oder Fehlerfall zu aktivieren. Im Falle des Notstopps wird der Antrieb sofort gestoppt und der STEPPER nimmt keinerlei Kommandos mehr entgegen. Der PROFIBUS DP wird hierdurch jedoch nicht unterbrochen, so dass der Betrieb der restlichen PROFIBUS DP-Teilnehmer weiterlaufen kann.

Die Notstopp-Funktion besitzt keinen speziellen Eingang, sondern wird durch die gleichzeitige Aktivierung der beiden Endschalter ausgelöst. Da die Endschalter grundsätzlich als Öffner ausgelegt sind, um Fehlfunktionen bei Drahtbruch auszuschließen, ist die Verdrahtung der Notstopp-Funktion so auszulegen, dass die Eingänge E2 und E4 bei Betätigung des Notstopp-Schalters gleichzeitig vom 24V-Pegel getrennt werden.

Hinweis:

Beim Start des STEPPER muss gewährleistet sein, dass die Eingänge der Endschalter bereits mit einem 24V-Pegel versorgt sind, da andernfalls die Notstopp-Funktion ungewollt aktiviert wird und der STEPPER keine Bewegung ausführen kann.

Anhang A: Gehäuseabmessungen

