



TSP10-PBE Feldbus Appendix

AHS Antriebstechnik GmbH
Fichtenweg 17
64319 Pfungstadt
Phone: +49 6157 9866110
Fax: +49 6157 9866112

Ausgabenhistorie:

Ausgabe	Bemerkungen
2015-03-05	Erstausgabe
2016-01-15	Ergänzungen des Kommando- und Statusworts ab Revision 64

© AHS Antriebstechnik GmbH, 2015

Ohne schriftliche Genehmigung der AHS Antriebstechnik GmbH darf kein Teil dieser Dokumentation in irgendeiner Form vervielfältigt, verarbeitet oder verbreitet werden.

Irrtum vorbehalten!

Inhaltliche Änderungen der Dokumentation und technische Änderungen der Produkte vorbehalten!

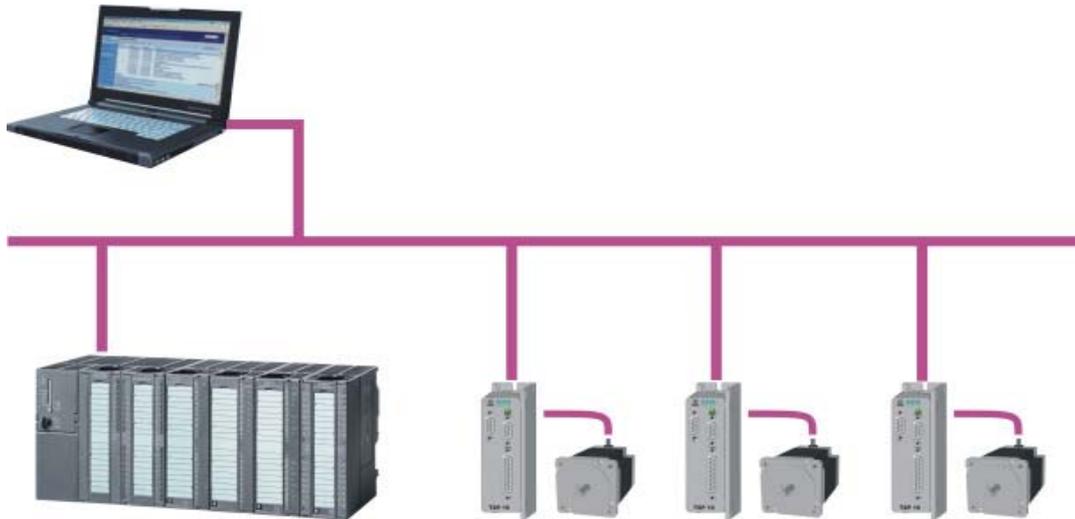
Die aktuelle Ausgabe dieser Betriebsanleitung steht im Internet unter www.ahs-antriebstechnik.de zur Verfügung.

Windows® ist ein eingetragenes Warenzeichen der Microsoft Corporation.

TSP10-PBE - Kompakte Schrittmotoransteuerungen

1 Profibus-Einstellungen

Die TSP10-PBE ist eine modulare Station mit 2 Modulen. Die Profibus-ID ist 0EE7h. Es gibt ein Ausgangsmodul und ein Eingangsmodul mit 6 konfigurierbaren Worten. Die Schrittmotoransteuerung kann im Drehzahl- oder Positionier-Mode betrieben werden. Der Profibus-Master kann Aktionen des Antriebs durch das Setzen von Steuer-Bits im Kommandowort starten.



Der aktuelle Status und die aktuelle Position können durch Lesen der Eingangsworte jederzeit von der Steuerung erfasst werden.

In die Ausgangsworte 5 und 6 wird die absolute Zielposition eingetragen, die die Schrittmotoransteuerung mit dem nächsten Bewegungsprofil erreichen soll. So kann eine exakte dezentrale Wegpositionierung vorgenommen werden ohne den Master zu belasten. Ein Beispielprojekt für eine S7-Steuerung kann per Email angefordert werden (info@ahs-antriebstechnik.de).

In diesem Appendix finden sie die Zusatzfunktionen der TSP10-PBE und die Unterschiede zum Basisgerät TSP10-BA. Die allgemeinen Funktionen des Geräts sind im Handbuch zur TSP10 (<http://www.ahs-antriebstechnik.de/downloads2.html>) beschrieben.

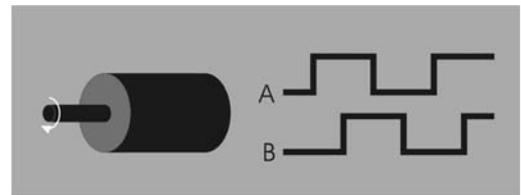
1.1 Steckerbelegung

Der Profibus wird an die zusätzliche 9-pol Sub-D-Buchse X5 links neben der Buchse für die serielle Schnittstelle X1 angeschlossen. Alle Signale dieser Schnittstelle sind optisch getrennt. Es werden Baudraten bis 12 Mbaud unterstützt.



Der Encoder wird über die 9-pol Sub-D-Buchse X6 unterhalb der Profibus-Buchse angeschlossen und hat folgende Belegung:

Signal	TSP10 X6 Pin	AE30 Aderfarbe	M21 Aderfarbe	DFS Aderfarbe
+5V	1	Rot	Rot	Rot
A	2	Violett	Grün	Weiß
B	3	Gelb	Orange	Rosa
Index	4	Grün	Weiß	Violett
GND	6	Schwarz	Schwarz	Blau
/A	7	Braun	Rot/Schwarz	Braun
/B	8	Orange	Weiß/Schwarz	Schwarz
/Index	9	Blau	Blau	Gelb



Die Eingänge für End-, Referenz- und Stopp-Schalter sind beim TSP10-PBE wie folgt festgelegt:

GND-DE	Bezugspotential	X2 pin 1
DE2	unterer Endschalter	X2 Pin 10
DE3	oberer Endschalter	X2 Pin 11
DE4	Referenzschalter	X2 Pin 12
DE5	Stopp-Schalter	X2 Pin 13

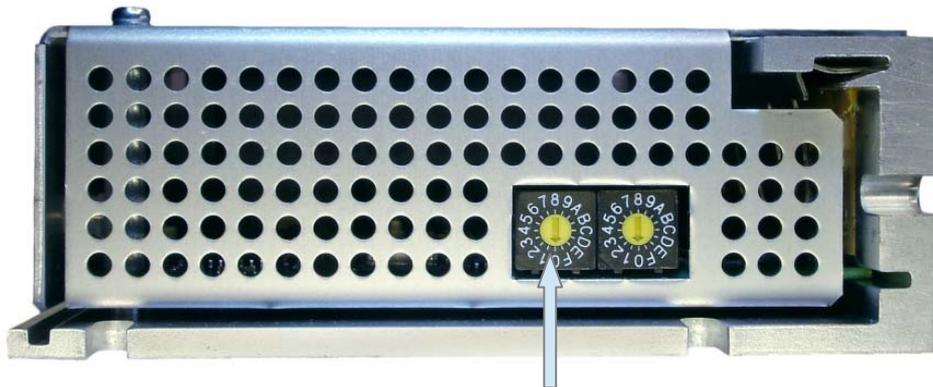


Bei Einstellung der Motordrehrichtung in "Gegendrehrichtung" wird die Funktion von DE2 und DE3 getauscht.

Die Eingänge sind optisch getrennt und für 24V oder 5V ausgelegt (siehe Typenschlüssel). Die Eingänge können über die Parameter-Daten (siehe 1.3) deaktiviert werden. Der Pegel aller Eingänge kann über ein Profibus-Wort ausgelesen werden.

1.2 Profibus-Adresse

Die beiden Drehschalter dienen beim TSP10-PBE der Einstellung der Profibus-Adresse. Der Motorstrom und der Mikroschrittfaktor werden über die Parameterdaten (siehe 1.3) vorgegeben.



Höherwertige Adresse (Faktor: 16)

Das Einstellen der Adresse erfolgt hexadezimal. Für eine Adresse bis 15 bleibt der linke Drehschalter auf Null. Für höhere Adressen wird die linke Drehschalterstellung mit 16 multipliziert und zum rechten Wert addiert.

1.3 Parameterdaten

Die Einstellungen für Motorstrom, Schrittfaktor (Mikroschrittfaktor), Endschalter, Referenzfahrt, Smoothing und Stillstandsstromreduzierung werden über die Profibus-User-Parameter-Daten vorgegeben. Vorgaben sind in der GSG-Datei festgelegt. Auszüge aus der GSG-Datei sind *kursiv* dargestellt.

Allgemein		Parameterzuordnung													
Modul Daten:															
Parameter	Value														
Endschalter	Endschalter sind angeschlossen														
Smoothing	ohne Smoothing														
Stillstandsstromreduzierung	nach 100 ms														
Motordrehrichtung	Standarddrehrichtung														
Stopp-Schalter	High-Signal zum Anhalten														
reduzierter Stillstandsstrom [%]	50														
Motorstrom [mA eff]	100														
Mikroschrittfaktor n*200 / Umdr.	20														
Referenzfahrt	Istposition														
DA1	Bereit														
DA2	Aktiviert														
DA3	Ziel erreicht														
DA4	Fehler														
Aktiviert	Aktiviert														
User Prm Daten:															
001	002	003	004	005	006	007	008	009	010	011	012	013	014	015	
00	00	00	00	8D	32	00	64	14	23	01	02	03	04	02	

In der folgenden Abbildung sieht man die Default-Parameter des Eingangsworte-Moduls.

Allgemein Parameterzuordnung Tag(s)		
Moduldaten		
Parameter	Value	
Encoder [Striche/Umdrehung]	0	
Eingangswort 1	Istgeschwindigkeit	
Eingangswort 2	Statuswort	
Eingangswort 3	Istposition Bit0-15	
Eingangswort 4	Istposition Bit16-31	
Eingangswort 5	Encoderwert Bit0-15	
Eingangswort 6	Logikpegel der Eing.	

1.3.1 Mikroschrittfaktor (Schrittweite)

Der Mikroschrittfaktor (Byte 9) kann zwischen 1 und 128 (200 und 25600 Schritten pro Umdrehungen) gewählt werden. Eine höhere Auflösung bietet einen gleichmäßigeren Betrieb bei kleinen Drehzahlen.

```
ExtUserPrmData = 1 "Mikroschrittfaktor n*200 / Umdr. "
Unsigned8 20 1-128
EndExtUserPrmData
```

Beispiel:

Ein Antrieb dreht eine Spindel, die pro Umdrehung 4 mm Vorschub erzeugt. Der Gesamtweg ist 700 mm - also 175 Umdrehungen. Bei einem gewählten Mikroschrittfaktor von $n = 20$ erhält man eine Mikroschrittauflösung von 4000 Schritten pro Umdrehung. Als Zielposition kann dann die Position in μm (Mikrometer) vorgeben werden. Die Endposition wäre somit 700.000.

1.3.2 Endschalter

Bei nicht verwendeten Endschaltern (X2 Pin 10 und 11) muss der Parameter (Byte 4 Bit 0) auf Null gesetzt werden, damit der Motor sich bewegen kann. Die Endschalter funktionieren als Öffner und verhindern im offenen Zustand die Bewegung des Motors.

```
PrmText = 2
Text(0) ="Keine Endschalter verwendet!"
Text(1) ="Endschalter sind angeschlossen"
EndPrmText
```

1.3.3 Smoothing

Ein weicherer Lauf kann durch Einschalten des Smoothing (Byte 4 Bit 1) erreicht werden.

```

PrmText           = 3
Text(0)           ="ohne Smoothing"
Text(1)           ="mit Smoothing"
EndPrmText
ExtUserPrmData    = 3 "Smoothing"
Bit (1) 0 0-1
Prm Text Ref      = 3

```

1.3.4 Motordrehrichtung

Der Motor dreht in der Standardeinstellung bei Blick auf die Welle im Uhrzeigersinn.

Wird als Einstellung „Gegendrehrichtung“ angewählt, werden die Endschaltefunktionen DE2 und DE3 getauscht.

```

PrmText           = 8
Text(0)           ="Standarddrehrichtung"
Text(1)           ="Gegendrehrichtung"
EndPrmText
ExtUserPrmData    = 8 "Motordrehrichtung"
Bit(6) 0 0-1
Prm_Text_Ref      = 8
EndExtUserPrmData

```

1.3.5 Stopp-Schalter

Die Polarität des Stopp-Eingangs kann konfiguriert werden.

```

PrmText           = 9
Text(0)           ="Low-Signal zum Anhalten"
Text(1)           ="High-Signal zum Anhalten"
EndPrmText
ExtUserPrmData    = 9 "Stopp-Schalter"
Bit(7) 1 0-1
Prm_Text_Ref      = 9
EndExtUserPrmData

```

1.3.6 Motorstrom

Der effektive Motorstrom kann zwischen 100 und 7000 mA (Byte 6 und 7) eingestellt werden. Der parametrisierte Wert darf niemals – auch nicht kurzzeitig - höher als der zulässige Motorstrom sein.

```

ExtUserPrmData    = 6 "Motorstrom [mA eff]"
Unsigned16 100 100-7000
EndExtUserPrmData

```

1.3.7 Stillstandsstromreduzierung

Die Wartezeit bis zur Reduzierung des Stroms kann in 8 Stufen (Byte 4 Bit 2-5) parametrisiert werden. Der Motorstrom wird in Prozent des eingestellten Wertes (Byte 5) vorgegeben, so dass die Erwärmung des Motors verringert wird.

```

PrmText           = 4
Text(0)           ="Keine Stromreduzierung"
Text(1)           ="nach 25 ms"
Text(2)           ="nach 50 ms"
Text(3)           ="nach 100 ms"
Text(4)           ="nach 250 ms"
Text(5)           ="nach 500 ms"
Text(6)           ="nach 1 Sekunde"
Text(7)           ="nach 2 Sekunde"
EndPrmText
ExtUserPrmData    = 4 "Stillstandsstromreduzierung"
BitArea (2-5) 3 0-7
Prm Text Ref      = 4
EndExtUserPrmData
ExtUserPrmData    = 5 "reduzierter Stillstandsstrom [%]"
Unsigned8 50 0-100
EndExtUserPrmData

```

1.3.8 Referenzfahrt

Die Referenzfahrtmethode kann über Byte 10 vorgegeben werden.

```

PrmText           = 5
Text(35)          ="Istposition"
Text(17)          ="Unterer Endschalter"
Text(18)          ="Oberer Endschalter"
Text(24)          ="Unterer Referenzsch. (Fahrtr. +)"
Text(29)          ="Unterer Referenzsch. (Fahrtr. -)"
Text(25)          ="Oberer Referenzsch. (Fahrtr. +)"
Text(28)          ="Oberer Referenzsch. (Fahrtr. -)"
Text(250)         ="Unterer mechanischer Anschlag"
Text(251)         ="Oberer mechanischer Anschlag"
EndPrmText
ExtUserPrmData    = 7 "Referenzfahrt"
Unsigned8 35 17-251
Prm Text Ref      = 5
EndExtUserPrmData

```

1.3.9 Ausgänge

Jeder Ausgang kann mit einer internen Funktion belegt oder direkt über das Profibus-Kommandowort gesteuert werden.

```

PrmText           = 12
Text(0)           ="nicht belegt"
Text(1)           ="Bereit"
Text(2)           ="Aktiviert"
Text(3)           ="Ziel erreicht"
Text(4)           ="Fehler"
Text(100)         ="Profibus Kommandowort"
EndPrmText
ExtUserPrmData    = 24 "DA1"
Unsigned8 1 0-100
Prm_Text_Ref     = 12
EndExtUserPrmData
ExtUserPrmData    = 25 "DA2"
Unsigned8 2 0-100
Prm_Text_Ref     = 12
EndExtUserPrmData
ExtUserPrmData    = 26 "DA3"
Unsigned8 3 0-100
Prm_Text_Ref     = 12
EndExtUserPrmData
ExtUserPrmData    = 27 "DA4"
Unsigned8 4 0-100
Prm_Text_Ref     = 12
EndExtUserPrmData
ExtUserPrmData    = 28 "Aktiviert"
Unsigned8 2 0-100
Prm_Text_Ref     = 12
EndExtUserPrmData

```

1.3.10 Encoder

```

ExtUserPrmData    = 10 "Encoder [Striche/Umdrehung]"
Signed16 0 -10000-10000
EndExtUserPrmData

```

Die Einstellung dieses Wertes hat folgende Bedeutung

0: der Impulszählerwert wird unverändert weitergegeben (Rohdaten)

1: der Zählerwert kann mit „Counter Reset“ (Kommandowort Bit1) zurückgesetzt werden

500-10000: der Zählerwert wird auf den eingestellten Mikroschrittfaktor umgerechnet und kann mit „Counter Reset“ zurückgesetzt werden

1.3.11 Eingangswort 1-6

Die Belegung der Eingangsworte kann über die Parameter des Moduls verändert werden.

```

PrmText           = 10
Text(0)           ="nicht belegt"
Text(1)           ="Istgeschwindigkeit"
Text(2)           ="Statuswort"
Text(3)           ="Istposition Bit0-15"
Text(4)           ="Istposition Bit16-31"
Text(5)           ="Encoderwert Bit0-15"
Text(6)           ="Encoderwert Bit16-31"
Text(7)           ="Encodergeschwindigkeit"
Text(8)           ="Logikpegel der Eing."
EndPrmText
ExtUserPrmData    = 12 "Eingangswort 1"
Unsigned8 1 0-8
Prm_Text_Ref      = 10
EndExtUserPrmData

```

1.3.12 Ausgangswort 1-6

Die vorgegebene Standard-Belegung der Ausgangsworte kann über die Parameter des Moduls verändert werden.

```

; Modul 1 mit 6 parametrierbaren Ausgangsworten
PrmText           = 11
Text(0)           ="nicht belegt"
Text(1)           ="Geschwindigkeit"
Text(2)           ="Startgeschwindigkeit"
Text(3)           ="Beschleunigungszeit [ms]"
Text(4)           ="Kommandowort"
Text(5)           ="Zielposition Bit0-15"
Text(6)           ="Zielposition Bit16-31"
EndPrmText
ExtUserPrmData    = 18 "Ausgangswort 1"
Unsigned8 1 0-8
Prm_Text_Ref      = 11
EndExtUserPrmData
ExtUserPrmData    = 19 "Ausgangswort 2"
Unsigned8 2 0-8
Prm_Text_Ref      = 11
EndExtUserPrmData
ExtUserPrmData    = 20 "Ausgangswort 3"
Unsigned8 3 0-8
Prm_Text_Ref      = 11
EndExtUserPrmData
ExtUserPrmData    = 21 "Ausgangswort 4"
Unsigned8 4 0-8
Prm_Text_Ref      = 11
EndExtUserPrmData
ExtUserPrmData    = 22 "Ausgangswort 5"
Unsigned8 5 0-8
Prm_Text_Ref      = 11
EndExtUserPrmData
ExtUserPrmData    = 23 "Ausgangswort 6"
Unsigned8 6 0-8
Prm_Text_Ref      = 11
EndExtUserPrmData

```

1.4 Diagnosedaten

In der anwenderspezifischen Diagnose (EXT_DIAG) wird als Statusinformation die Revisionsnummer (letzte Zahl) der Firmware als 16-Bit-Wert übertragen.

Beispiel: Firmware 2.1.60 Anwenderdiagnose:
 Byte 1: 3 (Länge)
 Byte 2: 0 (High-Byte)
 Byte 3: 60 (Low-Byte)

2 Modul 1 (Ausgangsworte)

In der Tabelle sind die einzelnen Ausgangsworte für die Steuerung der TSP10-PBE aufgeführt. Grün markiert sind die Teile, die nur für die Positionierung verwendet werden. Werte, die im Geschwindigkeitsmode verwendet werden, sind orange eingefärbt.

Modul 1																	
6 Ausgangsworte																	
1	Geschwindigkeit	Höchstgeschwindigkeit in UPM															
2	Startgeschwindigkeit	Anfangsgeschwindigkeit der Beschleunigungsrampe in Upm															
3	Beschleunigungszeit	Zeit der Beschleunigung in ms															
4	Kommandowort				A	4	3	2	1	H	E	D	S	V	a	R	I
5	Zielposition (32-Bit Integer)	Low-Wort															
6		High-Wort															

Die Byte-Reihenfolge wird in Big-Endian erwartet. Das heißt, das höherwertige Byte muss zuerst gesendet werden.

2.1 Kommandowort

Bit		Bedeutung	Beschreibung
0	I	Sync Encoder	↑ = Istposition auf Encoderposition setzen
1	R	Counter Reset	1 = Istposition auf 0 setzen
2	a	Beschleunigung	0 = keine Rampe (Sprung) 1 = lineare Geschwindigkeitsänderung
3	V	Velocity Mode	0 = Positionierung 1 = Geschwindigkeits-Mode
4	S	Motor-Start	0 = Motor anhalten 1 = Positionierung starten
5	D	Direction	0 = vorwärts 1 = rückwärts
6	E	Enable	0 = Motor ist stromlos 1 = Haltemoment bzw. Drehmoment aktiv
7	H	Homing	0 = Normalbetrieb 1 = Referenzfahrt
8	1	DA1	Logischer Pegel des Ausgangs, falls parametrier
9	2	DA2	Logischer Pegel des Ausgangs, falls parametrier
10	3	DA3	Logischer Pegel des Ausgangs, falls parametrier
11	4	DA4	Logischer Pegel des Ausgangs, falls parametrier
12	A	Aktiviert	Logischer Pegel des Ausgangs, falls parametrier

Im Positionier-Mode wird immer eine Rampe verwendet und die Drehrichtung ergibt sich aus der Zielposition und der Istposition.

2.2 Geschwindigkeit und Startgeschwindigkeit

Im ersten Ausgangswort wird die Geschwindigkeit in Umdrehungen pro Minute eingetragen. Soll der Motor beim Start zunächst auf eine Anfangsgeschwindigkeit springen, kann der Wert in Wort 2 vorgegeben werden.

2.3 Beschleunigung

In Wort 3 wird die Beschleunigungszeit vorgegeben. Die Beschleunigung ergibt sich aus der eingestellten Geschwindigkeit dividiert durch die Zeit für die Beschleunigung. Im Velocity-Mode wird für die Berechnung der Beschleunigung 100 Upm als Geschwindigkeit verwendet. Das Produkt aus Zeitbasis und Beschleunigungszeit ergibt die Zeit für die Geschwindigkeitsrampe in Millisekunden. Ist die eingestellte Beschleunigungszeit Null wird die Beschleunigung auf 1 Sekunde gesetzt.

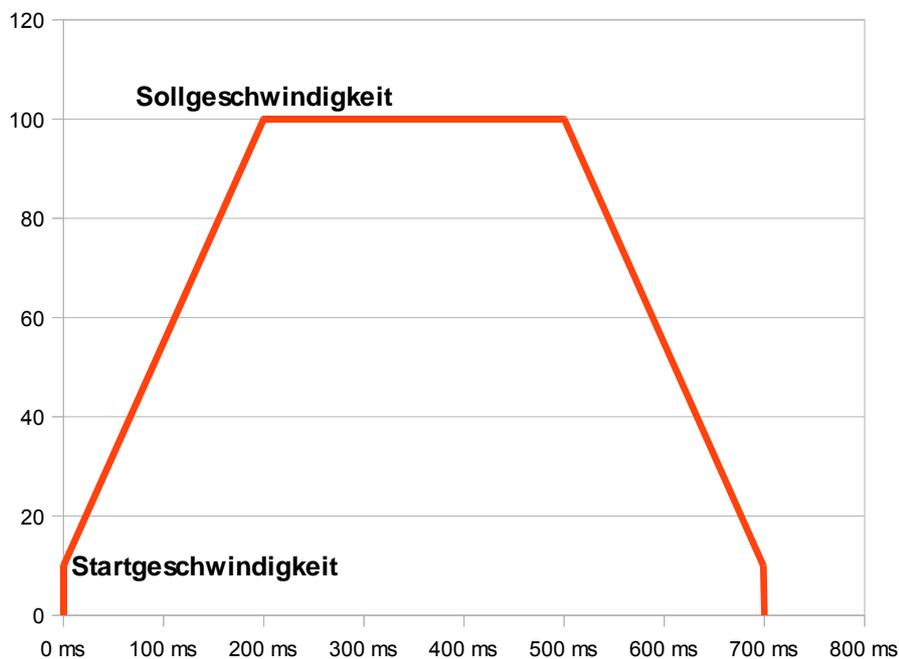


Abbildung 5: Geschwindigkeitsprofil einer Positionierung

Beschleunigungszeit = 200 ms

2.4 Zielposition

In den Ausgangsworten 5 und 6 wird die gewünschte Zielposition für den Fahrauftrag vorgegeben. Um bei einer großen Wegstrecke auch mit einer hohen Auflösung und der absoluten Positionierung arbeiten zu können wird die Zielposition als 32Bit-Integer-Wert angegeben.

Zur Orientierung bei der absoluten Positionierung dient grundsätzlich der Referenzpunkt.

Die Istposition des Steppers wird bei Standardparametrierung in den Eingangsworten 3 und 4 dargestellt. Der Zähler wird nach Erreichen des Referenzpunktes bei der Referenzfahrt auf 0 gestellt und der Zählerstatus im Statuswort auf referenziert gesetzt. Da nach dem Einschalten des Gerätes die Position undefiniert ist, wird der Zähler in diesem Zustand auch auf 0 gesetzt, um dem Anwender die Möglichkeit zu eröffnen, in beide Richtungen zu fahren. Durch die Abfrage des aktuellen Zählerstandes kann die Steuerung die momentane Position des Antriebs ermitteln und innerhalb des Anwenderprogramms verwerten.

Zur Positionierung ist die Abfrage des aktuellen Zählerstandes durch die Steuerung nicht erforderlich, im Statuswort kann dazu die Busy-Meldung verwendet werden. Das Modul vergleicht bei jedem Schritt die Istposition mit der Zielposition und beendet den Fahrauftrag, sobald beide übereinstimmen.

Hat der Stepper die Zielposition erreicht, wird das Anwenderprogramm eine neue Zielposition vorwählen. Bevor der Stepper nun erneut losläuft, muss jedoch das Motor-Start-Bit des Kommandowortes, das evtl. durch den vorhergehenden Lauf noch aktiv ist, zuerst inaktiv und dann wieder aktiv gesetzt werden.

Zu beachten ist weiterhin, dass der Stepper im Positionierbetrieb die Fahrtrichtung selbst vorgibt. Das entsprechende Kommandobit ist daher unwirksam. Die vom Stepper gewählte Fahrtrichtung resultiert stets aus dem Vergleich von Ist- und Zielposition.

Die Stoppbedingungen (Stopp- und Endschalter) werden während der Fahrt ständig überprüft. Sie haben natürlich eine höhere Priorität, d.h. wenn der vorgegebene Positionswert noch nicht erreicht ist und der entsprechende Schalter betätigt wird, wird der Motor sofort gestoppt.

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.

2.5 Referenzfahrt

Die Referenzfahrt stellt eine Besonderheit bei den Kommandos an den Schrittmotor-Controller dar. Die Referenzfahrt wird immer durch Setzen des Kommandobits 7 gestartet.

Es gibt verschiedene Möglichkeiten die Referenzfahrt durchzuführen. In den Parameterdaten sind die Referenzfahrtarten aufgelistet. Die Beschleunigung und die Geschwindigkeit der Referenzfahrt werden wie in Kapitel 2.2 und 2.3 beschrieben vorgegeben.

Nach der erfolgreichen Referenzfahrt steht der Antrieb genau auf dem Referenzpunkt und der Positionszähler steht auf Null. Im Statuswort wird signalisiert, dass die Daten des Positionszählers gültig sind. Eine weitere Referenzfahrt wird nicht ausgeführt bevor mit "Counter Reset" das Statusbit wieder gelöscht wird.

Jetzt ist der Controller bereit zur Annahme von absoluten Positionskommandos. Die Referenzfahrt kann jederzeit von der Steuerung gestartet werden. Unterbrochen wird die Referenzfahrt durch den Not-Stopp, durch den Stopp-Schalter und bei Erreichen des zweiten Endschalters, da dann 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 das Statuswort von der Steuerung erkannt und entsprechende Schritte eingeleitet werden.

Beispiel: **oberer Referenzschalter (Fahrtrichtung pos.)**, Schlitten zwischen Referenzpunkt und oberem Endschalter:

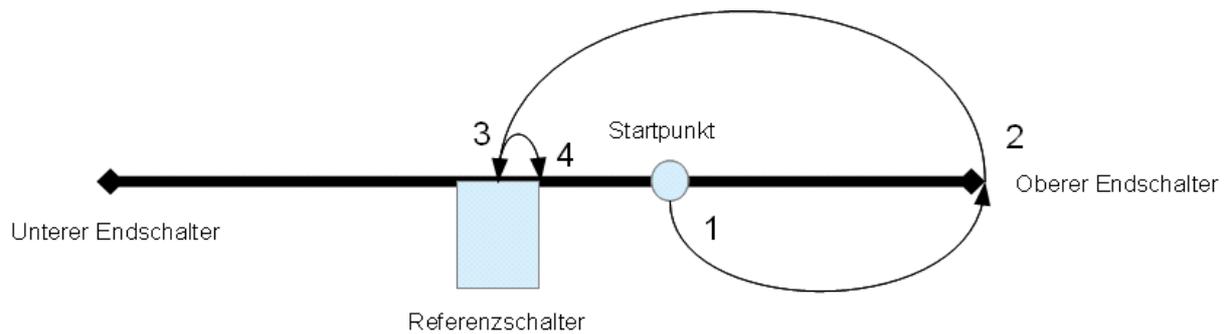
Ausgangsworte:

Geschwindigkeit	0x64
Kommandowort	0x00C0
Zielposition	beliebig

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

Aktuelle Geschwindigkeit	0x64
Statuswort	0x1000

1	Start: Positionszähler inkrementiert, Geschwindigkeit wie vorgewählt.
2	Oberer Endschalter spricht an. Fahrtrichtung wechselt, Positionszähler dekrementiert. Geschwindigkeit wie vorgewählt. Referenzpunkt befindet sich unterhalb der Anfangsposition.
3	Referenzschalter spricht an. Fahrtrichtung wechselt. Positionszählerwert steigt. Referenzschalter von oben getroffen, zurück, bis Referenzschalter-Kante erreicht ist.
4	Referenzpunkt wird durch Verlassen und erneutes Auslösen des Referenzschalters verifiziert. Positionszähler auf Null setzen. Motor steht auf dem Referenzpunkt.



3 Modul 2 (Eingangsworte)

Modul 2																			
6 Eingangsworte																			
1	Istgeschwindigkeit	Schrittgeschwindigkeit in Upm																	
2	Statuswort	E	U	C	A					B	I	V	D	↑	R	↓	S		
3	Istposition (32-Bit Integer)	Low-Wort																	
4		High-Wort																	
5	Encoderposition	Zählerwert Low-Wort																	
	Encoderposition	Zählerwert High-Wort																	
	Encodergeschwindigkeit	Encodergeschwindigkeit in Upm																	
6	Logikpegel der Eingänge DE1 – DE10 und Freigabe							F	1	0	9	8	7	6	5	4	3	2	1

Die Nummerierung entspricht der Standard-Parametrierung.

Die Byte-Reihenfolge wird in Big-Endian erwartet. Das heißt, das höherwertige Byte muss zuerst gesendet werden.

3.1 Statuswort

Um den Status des Schrittmotor-Controllers in der Steuerung überwachen zu können, wird das Statuswort des Controllers gelesen. Die untenstehende Tabelle zeigt, welche Anzeigen innerhalb des Statusworts ausgewertet werden können.

Bit		Bedeutung	Beschreibung
0	S	Stopp-Schalter	0 = Stopp-Schalter nicht aktiv 1 = Stopp-Schalter ausgelöst
1	↓	Endschalter unten	1 = unterer Endschalter aktiv
2	R	Referenzschalter	1 = Referenzschalter aktiv
3	↑	Endschalter oben	1 = oberer Endschalter aktiv
4	D	Fahrrichtung	0 = vorwärts (steigende Istposition) 1 = rückwärts
5	V	Endgeschwindigkeit	0 = nicht erreicht 1 = erreicht
6	I	In Position	0 = Abweichung des Encoderwerts 1 = Encoderwert im Bereich der Istposition
7	B	Busy	0 = Fahrauftrag beendet 1 = Fahrauftrag aktiv
8			
9			
10			
11	A	Notstopp/Alarm	1 = Alarm bzw. Notstopp aktiv
12	C	Zählerstatus	1 = Zählerstand ist referenziert
13	U	Busspannung	Zwischenkreisspannung ok
14	E	Fehler	1 = Fehler

3.2 Istposition und Istgeschwindigkeit

Die Werte für die aktuelle Position und Geschwindigkeit haben das gleiche Format wie die Vorgabewerte in den Ausgangsworten.

3.3 Notstopp

Die Schrittmotoransteuerung bietet die Möglichkeit, einen Notstopp für den Gefahren- oder Fehlerfall zu aktivieren. Im Falle des Notstopps wird der Antrieb sofort gestoppt und der Antrieb 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 DE2 und DE3 bei Betätigung des Notstopp-Schalters gleichzeitig vom 24V-Pegel getrennt werden.

3.4 Alarmzustand

Der Alarmzustand wird durch folgende Ereignisse ausgelöst:

1. Bei der Referenzfahrt auf Referenzschalter wurde der zweite Endschalter aktiviert.
2. Überstrom
3. Übertemperatur

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

Es wurde eine Möglichkeit geschaffen, den Alarmzustand auch ohne Abschaltung zu beenden. Dazu wurde ein **Pseudocode** für das Ausgangswort Geschwindigkeit eingeführt. Es ist der Code **0xAA55**. Dieser Code kommt im normalen Betrieb nicht vor, da die max. Geschwindigkeit mit 3000 vorgegeben ist. 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. Zu beachten ist hierbei, dass die Pseudocodeausgabe 0xAA55 auch **tatsächlich verlassen** wird, denn erst dann wird der normale Betriebszustand wieder eingenommen.