



# PROFINET-Handbuch

***Wichtig!***  
***Vor Gebrauch sorgfältig lesen!***  
***Aufbewahren für späteres Nachschlagen!***

---

## Original PROFINET-Handbuch

### › Urheberrechte

© 2024 Metronix Meßgeräte und Elektronik GmbH. Alle Rechte vorbehalten.

Die Informationen und Angaben in diesem Dokument sind nach bestem Wissen zusammengestellt worden. Trotzdem können abweichende Angaben zwischen dem Dokument und dem Produkt nicht mit letzter Sicherheit ausgeschlossen werden. Für die Geräte und zugehörige Programme in der dem Kunden überlassenen Fassung gewährleistet Metronix den vertragsgemäßen Gebrauch in Übereinstimmung mit der Nutzerdokumentation. Im Falle erheblicher Abweichungen von der Nutzerdokumentation ist Metronix zur Nachbesserung berechtigt und, soweit diese nicht mit unangemessen Aufwand verbunden ist, auch verpflichtet. Eine eventuelle Gewährleistung erstreckt sich nicht auf Mängel, die durch Abweichen von den für das Gerät vorgesehenen und in der Nutzerdokumentation angegebenen Einsatzbedingungen verursacht werden.

Metronix übernimmt keine Gewähr dafür, dass die Produkte den Anforderungen und Zwecken des Erwerbers genügen oder mit anderen von ihm ausgewählten Produkten zusammenarbeiten. Metronix übernimmt keine Haftung für Folgeschäden, die im Zusammenwirken der Produkte mit anderen Produkten oder aufgrund unsachgemäßer Handhabung an Maschinen oder Anlagen entstehen.

Metronix behält sich das Recht vor, das Dokument oder das Produkt ohne vorherige Ankündigung zu ändern, zu ergänzen oder zu verbessern.

Dieses Dokument darf weder ganz noch teilweise ohne ausdrückliche Genehmigung des Urhebers in irgendeiner Form reproduziert oder in eine andere natürliche oder maschinenlesbare Sprache oder auf Datenträger übertragen werden, sei es elektronisch, mechanisch, optisch oder auf andere Weise.

### › Warenzeichen

Alle Produktnamen in diesem Dokument können eingetragene Warenzeichen sein. Alle Warenzeichen in diesem Dokument werden nur zur Identifikation des jeweiligen Produkts verwendet.

Metronix ServoCommander® ist ein eingetragenes Warenzeichen der Metronix Meßgeräte und Elektronik GmbH.

---

## › **Kontakt**daten

Metronix Meßgeräte und Elektronik GmbH  
Kocherstraße 3  
38120 Braunschweig  
Germany

Telefon: +49 (0)531 8668 0  
Telefax: +49 (0)531 8668 555  
E-mail: [vertrieb@metronix.de](mailto:vertrieb@metronix.de)  
<https://www.metronix.de>

## › **Revisions**information

Handbuchname	PROFINET-Handbuch
Dateiname	PN-HB_1p 0_DE.pdf
Version	1.0
Jahr	2024

# Inhaltsverzeichnis

<b>1 Zu diesem Handbuch</b>	<b>7</b>
1.1 Aufbau der Warnhinweise	7
1.2 Schreibweisen in diesem Handbuch	8
<b>2 Schnellstart</b>	<b>9</b>
2.1 Einleitung: PROFINET und PROFIdrive	9
2.2 Verkabelung und Steckerbelegung	10
2.2.1 Verkabelungshinweise	11
2.3 Status-LEDs (BL 4000-C)	11
2.4 PROFINET aktivieren	12
2.5 Einbinden des Servoreglers im TIA-Portal	14
2.5.1 Beispiel-Funktionsbaustein	15
2.5.1.1 Allgemeine Eingänge/Ausgänge	16
2.5.1.2 Eingänge/Ausgänge für alle Betriebsarten	17
2.5.1.3 Eingänge/Ausgänge für die Betriebsart Positionieren	18
2.5.1.4 Eingänge/Ausgänge für die Betriebsart Drehzahlregelung	20
2.5.1.5 Eingänge/Ausgänge für die Betriebsart Drehmomentregelung	21
2.5.1.6 Eingänge/Ausgänge für den Tippbetrieb (Jogging)	21
2.5.1.7 Eingänge/Ausgänge für den Parameterzugriff (PKW)	22
<b>3 Parameter</b>	<b>23</b>
3.1 Physikalische Einheiten	24
3.1.1 Übersicht	24
3.1.2 Parametrierung der physikalischen Einheiten	24
3.2 Übersicht	26
3.3 PNUs für alle Betriebsarten	29
3.3.1 PNU 1500: Operating Mode	29
3.3.2 PNU 1100: Position Actual Value	29
3.3.3 PNU 1101: Velocity Actual Value	29
3.3.4 PNU 1102: Current Actual Value	30
3.3.5 CAN-Objekt 6077h: torque_actual_value	30
3.3.6 PNU 1110: Sampling Positions	30
3.3.7 PNU 1141: Digital Inputs	31
3.3.8 PNU 1600: Last Error Code	31
3.4 PNUs zur Betriebsart Positionieren	32
3.4.1 PNU 1000: Position Set Number	32
3.4.2 PNU 1001: Position Data	33
3.4.3 PNU 1002: Start Set Number	34
3.4.4 PNU 1003: Position Profile Type	35
3.4.5 PNU 1004: Override Factor	35
3.4.6 PNU 1005: Software Position Limits	36
3.4.7 PNU 1006: Rotary Axis	36
3.4.8 PNU 1050: Homing Method	37
3.4.9 PNU 1051: Home Offset	38
3.4.10 Referenzfahrt-Abläufe	39

3.4.10.1	Methode -17 und -18: Anschlag	39
3.4.10.2	Methoden -1 und -2: Anschlag mit Nullimpulsauswertung	39
3.4.10.3	Methoden 17 und 18: Positiver und negativer Endschalter	40
3.4.10.4	Methoden 1 und 2: Positiver und negativer Endschalter mit Nullimpulsauswertung	40
3.4.10.5	Methoden 23 und 27: Referenzschalter	41
3.4.10.6	Methoden 7 und 11: Referenzschalter und Nullimpulsauswertung	42
3.4.10.7	Methoden -23 und -27: Referenzfahrt (pos/neg) auf den Referenzschalter	43
3.4.10.8	Methoden 32 und 33: Referenzfahrt auf den Nullimpuls	43
3.4.10.9	Methode 34: Referenzfahrt auf die aktuelle Position	43
3.4.11	PNU 1060: Thread Speed	44
3.4.12	PNU 1270: Position Control Parameters	44
3.4.13	PNU 1271: Position Window Data	44
3.4.14	PNU 1272: Following Error Data	45
3.4.15	PNU 1273: Position Error Data	45
3.5	PNUs zur Betriebsart Drehzahlregelung	46
3.5.1	PNU 1010: Target Velocity	46
3.5.2	PNU 1011: Accelerations for Velocity Control	46
3.5.3	CAN-Objekt 2415h: current_limitation	47
3.6	PNUs zur Betriebsart Drehmomentregelung	48
3.6.1	CAN-Objekt 6071h: target_torque	48
3.6.2	CAN-Objekt 2416h: speed_limitation	48
3.7	PNUs für den Tippbetrieb (Jogging)	49
3.7.1	PNU 1040: Symmetrical Jogging	49
3.7.2	PNU 1041: Jogging Positive	50
3.7.3	PNU 1042: Jogging Negative	51
3.8	Weitere PNUs	52
3.8.1	PNU 964: Device Identification	52
3.8.2	PNU 971: Transfer into a non-volatile memory	53
3.8.3	PNU 2000: PKW Access	54
3.8.4	PNU 2010: Placeholder	55
3.8.5	PNU 2011: Element 0	55
<b>4</b>	<b>Gerätesteuerung</b>	<b>56</b>
4.1	Verwendete Begriffe	56
4.2	PNU 967: Control word 1	57
4.3	PNU 968: Status word 1	60
4.4	Das Zustandsdiagramm (State Machine)	62
4.4.1	Zustandsdiagramm: Zustände	63
4.4.2	Zustandsdiagramm: Zustandsübergänge	64
4.5	Anzeige des aktuellen Steuer-/Statusworts (Control word 1/Status word 1)	66
4.6	Diagnose - Alarme	67
<b>5</b>	<b>Betriebsarten</b>	<b>68</b>
5.1	Übersicht	68
5.2	Betriebsart Drehmomentregelung	68
5.3	Betriebsart Drehzahlregelung	69

5.4 Betriebsart Positionieren .....	69
<b>6 Zyklische Kommunikation .....</b>	<b>71</b>
6.1 Telegrammeditor .....	71
6.1.1 Anzeige der aktuellen Telegrammdaten .....	76
6.2 Konfiguration der Telegramme im TIA-Portal .....	77
<b>7 Anhang .....</b>	<b>78</b>
7.1 Einstellung für den Beispiel-Funktionsbaustein .....	78

# 1 Zu diesem Handbuch

Dieses Handbuch beschreibt, wie die Servoregler der Gerätefamilie smartServo BL 4000 in ein PROFINET-Netzwerk einbezogen werden können. Es werden der physikalische Anschluss, die Aktivierung des Feldbus-Protokolls, die Einbindung in das Netzwerk und die Parameter zur Anpassung an die jeweilige Applikation beschrieben. Es richtet sich an Personen, die bereits mit der jeweiligen Servoregler-Familie vertraut sind und das entsprechende Produkthandbuch gelesen und verstanden haben.

Im Produkthandbuch sind Hinweise für den sachgemäßen und fachgerechten Transport, die Lagerung, die Montage, die Installation, die Projektierung und den einwandfreien und sicheren Betrieb des Servoreglers enthalten.

Das Produkthandbuch enthält Sicherheitshinweise, die unbedingt beachtet werden müssen. Die Produkthandbücher stehen auf unserer Homepage (<http://www.metronix.de>) zum Download zur Verfügung.

## 1.1 Aufbau der Warnhinweise

Warnhinweise sind folgendermaßen aufgebaut:

- Signalwort
- Art der Gefährdung
- Maßnahmen zur Abwehr der Gefährdung

### > Verwendete Signalwörter

#### **▲ GEFAHR**

Bezeichnet eine unmittelbar drohende Gefahr.

Wenn die Situation nicht gemieden wird, sind Tod oder schwerste Verletzungen die Folge.

#### **▲ WARNUNG**

Bezeichnet eine möglicherweise gefährliche Situation.

Wenn die Situation nicht gemieden wird, können Tod oder schwerste Verletzungen die Folge sein.

#### **▲ VORSICHT**

Bezeichnet eine möglicherweise gefährliche Situation.

Wenn die Situation nicht gemieden wird, können leichte oder geringfügige Verletzungen die Folge sein.

#### **ACHTUNG**

Bezeichnet eine Warnung vor Sachschäden.

### > Verwendete Warnzeichen gemäß ISO 7010

Warnzeichen	Erklärung
	Warnung vor lebensgefährlicher elektrischer Spannung.

## 1.2 Schreibweisen in diesem Handbuch

### > Aufbau der Hinweise

Hinweise in diesem Handbuch sind folgendermaßen aufgebaut:

- Signalwort "HINWEIS"
- Einleitender Satz
- Erklärungen, spezielle Hinweise und Tipps

### > Bedienelemente, Menüs

Bedienelemente, Menüs und Menüpfade werden in Orange geschrieben.

**Beispiel:** Doppelklick auf das gewünschte Gerät oder Anklicken der Schaltfläche **Verbindung neu aufbauen** stellt eine Online-Verbindung her.

### > PNUs, Bitkonstanten

Begriffe wie zum Beispiel Parameternamen (PNUs) werden in Blau geschrieben. Bitkonstanten werden durch eine andere Schriftart hervorgehoben.

**Beispiel:** Der erste Fall lässt sich erreichen, indem das Bit Change `immediately` im `Control word 1` beim Starten des zweiten Fahrauftrags gesetzt wird.

### > Zustände, Kommandos

Reglerzustände (siehe Abschnitt 4 *Gerätesteuerung* auf Seite 56) werden in einer anderen Schriftart gesetzt und großgeschrieben. Kommandos werden mit einem weißen Kasten hinterlegt.

**Beispiel:**

SWITCHING_ON_INHIBITED	Der Servoregler hat seinen Selbsttest abgeschlossen.		
4	<code>Disable Operation</code>	0111	Wegnahme der Reglerfreigabe

## 2 Schnellstart

Dieses Kapitel beschreibt, wie die Servoregler mit einer PROFINET -Steuerung verbunden und in Betrieb genommen werden, um eine schnelle Einrichtung zum Starten der Anwendungs-Entwicklung zu erhalten. Im Abschnitt 3 *Parameter* auf Seite 23 werden dann alle verfügbaren Parameter beschrieben, um den Servoregler an die jeweilige Applikation anzupassen. Das folgende Kapitel richtet sich an Anwender, die bereits eine Industriesteuerung besitzen.

### 2.1 Einleitung: PROFINET und PROFIdrive

#### › PROFINET IO

PROFINET IO (Input - Output) erlaubt die Anbindung von dezentralen Feldgeräten wie E/A, Antrieben, Ventilen, Messumformern oder Analysegeräten an ein zentrales Automatisierungsgerät, wie SPS, PC oder Prozessleitsystem. Die Datenübertragung basiert auf der Fast-Ethernet-Standardübertragung mit 100 Mbit/s. Drei aufeinander aufbauende Konformitätsklassen (CC-A, CC-B und CC-C) geben den Funktionsumfang und die Echtzeit-Eigenschaften von PROFINET IO an. Die Servoregler der Gerätefamilie smartServo BL 4000 erfüllen die **Conformance Class B (CC-B)**:

#### › PROFIdrive

Das „PROFIBUS profile for drive technology“, kurz PROFIdrive, ist ein Standard für Hersteller zur Implementierung von PROFIBUS-Schnittstellen bei Antrieben. PROFIdrive spezifiziert Konfiguration, Diagnose, Datenaustausch und Zustandsmaschinen mit einem Master. In den Geräten der Gerätefamilie smartServo BL 4000 ist ein Metronix-spezifisches Applikationsprofil implementiert, das sich an der PROFIdrive Version 3.1 anlehnt. So ist beispielsweise die Ansteuerung der Zustandsmaschine über Control word 1 und Status word 1 bzw. die Bedeutung der einzelnen Bits weitgehend übernommen worden. Ebenso ist das Konzept der Parameternummern (PNU) übernommen worden, um auf die Parameter des Antriebs zugreifen zu können. Darüber hinaus besteht zusätzlich die Möglichkeit auf alle Parameter des CANopen Objektverzeichnisses zuzugreifen. PROFIdrive-Telegramme wie z.B. MC-Servo werden nicht unterstützt.

#### › Metronix Applikationsprotokoll

Das Metronix-spezifische Applikationsprofil basiert auf einer zyklischen Kommunikation zwischen Steuerung und Regler, die in Abschnitt 6 *Zyklische Kommunikation* auf Seite 71 beschrieben wird. Die Steuerung über **Control word 1** und **Status word 1** ist im Abschnitt 4 *Gerätesteuerung* auf Seite 56 beschrieben. Eine Übersicht über alle Parameter findet sich im Abschnitt 3 *Parameter* auf Seite 23. Zur Vereinfachung der Feldbus-Anschaltung sind alle wichtigen Funktionen und Parameter in einem Funktionsbaustein zusammengefasst. Dieser ist in Abschnitt 2.5 *Einbinden des Servoreglers im TIA-Portal* auf Seite 14 beschrieben.

## 2.2 Verkabelung und Steckerbelegung

Die PROFINET-Schnittstelle ist bei den Servoreglern BL 4000-C immer integriert. Bei Servoreglern der Gerätereihe BL 4000-M bzw. BL 4000-D ist hingegen das PROFINET-Interface nur bei der Feldbusvariante PROFINET/EtherCAT vorhanden. Nähere Informationen hierzu finden sich im Abschnitt *Produktbeschreibung* im Produkthandbuch BL 4000-D und BL 4000-M.

### > BL 4000-C

Gemäß der PROFINET-Spezifikation sind zwei RJ45-Stecker für den Busanschluss vorhanden [X21].



Abbildung 1: Steckeransicht Feldbusschnittstelle BL 4000-C

Die beiden Anschlüsse RTE0 und RTE1 sind RJ 45-Buchsen, Cat. 6

Pin	Bezeichnung	Spezifikation
1	RX-	Empfängersignal -
2	RX+	Empfängersignal +
3	TX-	Sendesignal -
4	-	-
5	-	-
6	TX+	Sendesignal +
7	-	-
8	-	-

### > BL 4000-D und BL 4000-M (Feldbusvariante PROFINET/EtherCAT)

Bei diesen Geräten ist der PROFINET-Anschluss als M8-Steckverbinder gemäß IEC 61076-114 (4-polig, Buchse, D-codiert) ausgeführt. Beachten Sie, dass die Feldbusvariante CAN zwar die identischen Stecker verwendet, elektrisch aber nicht kompatibel ist. Die Feldbusvarianten dürfen nicht verwechselt und auch keinesfalls gleichzeitig im selben Netzwerk verwendet werden!

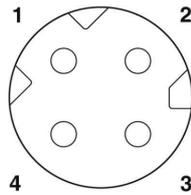


Abbildung 2: Steckerbelegung Feldbuschnittstelle BL 4000-D und BL 4000-M

Steckerbelegung **EtherCAT/PROFINET**:

Pin	Bezeichnung	Beschreibung	Farbe
1	TD+	Sendesignal +	Gelb
2	RD+	Empfängersignal +	Weiß
3	TD-	Sendesignal -	Orange
4	RD-	Empfängersignal -	Blau

Für die Verkabelung empfehlen wir die Verwendung der folgenden vorkonfektionierten Kabel oder vergleichbarer Produkte anderer Hersteller.

Konfektioniertes Netzkabel Phoenix Contact:

Rundstecker auf Rundstecker: NBC-M8MSD/ 1,0-93C/M8MSD - 1423707

Rundstecker auf RJ45: NBC-M8MSD/ 1,0-93C/R4AC - 1423711

Rundstecker auf Leitungsende: NBC-M8MSD/ 1,0-93C - 1423703

## 2.2.1 Verkabelungshinweise

In Automatisierungsanlagen sind ausschließlich für PROFINET zugelassene Kabel mit entsprechender Herstellererklärung zu verwenden. Die gemeinsame Verlegung von Energieleitungen (z.B. Motorleitungen) und Kommunikationsleitungen sollte vermieden werden, um den elektromagnetischen Einfluss der Energieleitungen auf die Kommunikationsleitungen zu minimieren. Die einzelnen PROFINET-Knoten werden in der Regel linienförmig miteinander verbunden.

## 2.3 Status-LEDs (BL 4000-C)

Zur einfachen Anzeige des PROFINET-Bus-Status ist die Servoreglerfamilie BL 4000-C mit zwei Feldbusstatus-LEDs ausgestattet. Das Verhalten der LEDs ist durch PROFINET vordefiniert.

Die obere LED (SF) zeigt Systemfehler an:

Blinkcode	Description
LED ist aus	Keine Systemfehler.
LED blinkt rot (1 Hz für 3 s)	PROFINET Geräteidentifikation

Blinkcode	Description
LED leuchtet rot	Einer der folgenden Fehler liegt vor: <ul style="list-style-type: none"> <li>• Watchdog timeout</li> <li>• Kanaldiagnose</li> <li>• Allgemeine oder erweiterte Diagnose</li> <li>• Systemfehler</li> </ul>

Die untere LED (BF) zeigt mögliche Busfehler an:

Blinkcode	Description
LED ist aus	Keine Busfehler.
LED leuchtet rot	Einer der folgenden Fehler liegt vor: <ul style="list-style-type: none"> <li>• Keine Konfiguration</li> <li>• Fehler am physikalischen Link</li> <li>• Kein physikalischer Link</li> </ul>
LED blinkt rot (2 Hz)	Es werden keine Daten übertragen.

## 2.4 PROFINET aktivieren

Die PROFINET-Feldbuskommunikation muss einmalig über das PROFINET-Fenster des Metronix ServoCommander® aktiviert werden ([Parameter/Feldbus/PROFINET/Betriebsparameter](#)).

The screenshot shows the 'PROFINET' configuration window. It includes the following sections:

- Aktivierung:** A checkbox for 'PROFINET aktiv' is currently unchecked.
- Gerätename / Name of station:** An empty text input field.
- Protokoll:** A dropdown menu set to 'Standardprotokoll'.
- PROFInergy:** A dropdown menu set to 'inaktiv'.
- IP-Konfiguration (PROFINET):** A section with input fields for IP-Adresse (192, 168, 0, 86), Subnetzmaske (255, 255, 255, 0), and Gateway (192, 168, 0, 86). To the right, a green box contains the same IP address and gateway values, along with three MAC addresses: 00-40-55-40-02-2A, 00-40-55-49-02-2A, and 00-40-55-50-02-2A.
- Parametrierschnittstelle:** A dropdown menu set to 'Automatisch' and a green box containing 'OnBoard-Ethernet'.
- Port-Nummer:** An input field containing the value '8802'.

At the bottom, there are three buttons: 'OK', 'Abbruch', and 'Hilfe'.

Abbildung 3: Einstellungen der Betriebsparameter unter PROFINET

### › **Aktivierung**

Die PROFINET Kommunikation wird über das Kontrollkästchen **PROFINET aktiv** aktiviert. Es ist zu beachten, dass die Aktivierung der PROFINET-Kommunikation erst nach einem **Save & Reset** wirksam ist. Die Deaktivierung der Kommunikation erfolgt dagegen unmittelbar.

### › **Gerätename, Protokoll und PROFIenergy**

Zur Konfiguration der Kommunikation auf Seiten des Servoreglers ist lediglich der **Gerätename** (Name of Station) erforderlich. Das Zuweisen der IP-Adresse erfolgt auf Grund des Gerätenamens mit dem DCP-Protokoll (Discovery and basic Configuration Protocol). Für die Zuweisung ist es notwendig, dass sich ein DCP fähiger Controller im Netzwerk befindet.

Das verwendete Applikationsprotokoll wird in der Auswahlliste **Protokoll** angezeigt. Zurzeit wird nur das Standardprotokoll unterstützt.

Mit der Auswahlliste **PROFIenergy** kann das standardisierte Energieeffizienzprofil PROFIenergy aktiviert oder deaktiviert werden.

### › **IP-Konfiguration (PROFINET)**

Dem Servoregler muss eine eindeutige IP-Adresse zugeordnet werden. Bei einer dynamischen Adressvergabe wird die IP-Adresse sowie die zugehörige Subnetzmaske und das Gateway über das DCP-Protokoll (anhand des Gerätenamens) vergeben. Eine zuvor zugeordnete statische IP-Adresse wird hierbei überschrieben.

### › **Parametrierschnittstelle**

Die Parametrierung bzw. Diagnose des Servoreglers kann mit dem Metronix ServoCommander<sup>®</sup> über das PROFINET-Netzwerk erfolgen. Dazu wird der PC/Laptop an das PROFINET-Netzwerk angeschlossen.

Bei Geräten der Gerätefamilie BL 4000 wird die Kommunikation über PROFINET automatisch aktiviert, da eine Kommunikation über [X18] nach Aktivierung der PROFINET-Kommunikation nicht möglich ist.

Im Eingabefeld **Port-Nummer** kann der zu verwendende Ethernet-Port angegeben werden.

### › **Save & Reset**

Die Einstellungen der Betriebsparameter werden erst mit dem Betätigen der Schaltfläche **Save & Reset** gültig. Dabei werden die Einstellungen im Parametersatz gespeichert und anschließend ein Reset des Servoreglers durchgeführt.

## 2.5 Einbinden des Servoreglers im TIA-Portal

Um den Servoregler schnell in das TIA-Portal zu integrieren, stehen ein Beispielprojekt inklusive eines Funktionsbausteins zur Verfügung. Für den Servoregler ist eine zum Funktionsbaustein passende DCO-Datei verfügbar.

Alle Beispielprojekte und Parameterdateien können von unserer Homepage (<https://www.metronix.de>) heruntergeladen werden.

### **ACHTUNG** Sachschäden durch Fehlfunktion

Wenn Sie den Beispiel-Funktionsbaustein oder das Beispielprojekt in Ihrer Applikation verwenden, müssen Sie abschließend prüfen, ob alle funktions- und sicherheitsrelevanten Anforderungen ihrer Applikation erfüllt werden.

#### › Öffnen des Beispielprojekts

Laden Sie zunächst das *Beispielprojekt für TIA-Portal* von unserer Homepage (<http://www.metronix.de>) herunter. Nach der Extraktion der ZIP-Datei können Sie die folgende Datei im TIA-Portal öffnen:

UnifiedFunctionBlock\TIAv7\_UnifiedFunctionBlock.apxx

Die Platzhalter xx geben dabei die Versionsnummer des TIA-Portals an, mit dem die Datei erstellt wurde.

Durch die Verwendung des Beispielprojekts werden automatisch die passende Gerätebeschreibungsdatei (GSDML) sowie der Beispiel-Funktionsbaustein installiert. Das Projekt ist für eine S7-1511 CPU ausgelegt. Wenn Sie eine andere CPU verwenden, müssen Sie die S7-1511 in der Topologieansicht durch Ihren Typ ersetzen. Ebenso wird im Beispiel ein BL 4102-C verwendet, den Sie unter Umständen durch den von Ihnen verwendeten Reglertyp tauschen müssen. Alle Regler der Reglerfamilie BL 4000 haben die identische PROFINET-Anschaltung.

#### › Laden der Parameterdatei

Damit der Regler und die Steuerung zyklisch Daten austauschen können, müssen auf Reglerseite drei Voraussetzungen erfüllt sein:

- Der Regler muss einen passenden Gerätenamen haben.
- Die zyklischen Telegramme müssen passend eingestellt sein.
- Die PROFINET-Kommunikation muss aktiviert werden.

Passend zum Beispielprojekt ist die reglerseitige Konfiguration in der folgenden Parameterdatei hinterlegt, die Sie mit dem Metronix ServoCommander® in den Servoregler laden:

ProfiNET\_UnifiedFunctionBlock.DCO

Diese enthält nur die PROFINET-spezifischen Parametrierungen, verändert aber nicht die Motor- bzw. Geber-spezifischen Parameter des Servoreglers (Motorstrom, Winkelgeber, etc.).

## 2.5.1 Beispiel-Funktionsbaustein

Der Zugriff auf den Regler ist in einem Funktionsbaustein gekapselt, den sie im TIA-Portal unter **Programmbausteine, Main** finden.

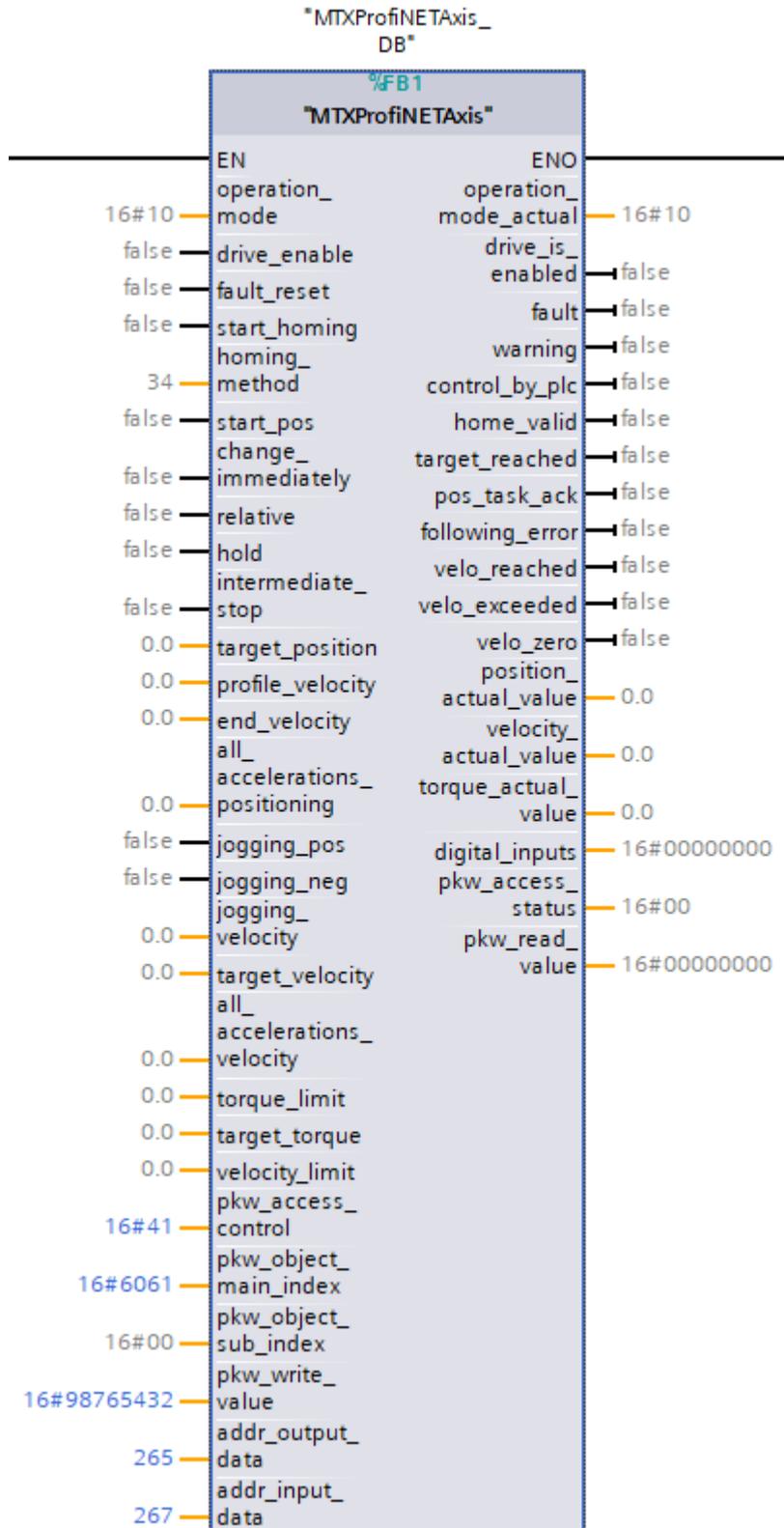


Abbildung 4: Beispiel-Funktionsbaustein

Der Funktionsbaustein kann für die Betriebsarten Positionieren, Drehzahlregelung und Drehmomentregelung genutzt werden. Zusätzlich dazu ist eine Momentenbegrenzung bzw. eine Drehzahlbegrenzung verwendbar. Der Baustein besitzt Eingänge, um den Regler freizugeben, die Betriebsart zu wechseln und Sollwerte vorzugeben. Über die Ausgänge können der aktuelle Zustand (Freigegeben, Fehler, etc.), die Betriebsart und die Istwerte des Reglers ausgelesen werden. Werden darüber hinaus weitere Parameter benötigt, finden Sie diese im Abschnitt 3 *Parameter* auf Seite 23

**⚠ VORSICHT Verletzungsgefahr durch falsch parametrieren Servoregler**

Ein falsch parametrierter Servoregler kann unkontrollierte Drehbewegungen und dadurch Personenschäden oder Sachschäden verursachen.

Stellen Sie vor dem allerersten Einschalten der Endstufe sicher, dass der Servoregler die von Ihnen gewünschten Parameter enthält.

### 2.5.1.1 Allgemeine Eingänge/Ausgänge

Eingang	Bedeutung
addr_output_data	Hardwarekennung der <b>Ausgangsdaten</b> des verbundenen Servoreglers. Über diese ist der FB mit der jeweiligen Achse verknüpft. Die Hardwarekennung wird im TIA-Portal unter den Systemkonstanten des jeweiligen Reglers angezeigt. Sie ist im Beispielprojekt bereits korrekt eingestellt.
addr_input_data	Hardwarekennung der <b>Eingangsdaten</b> des verbundenen Servoreglers.

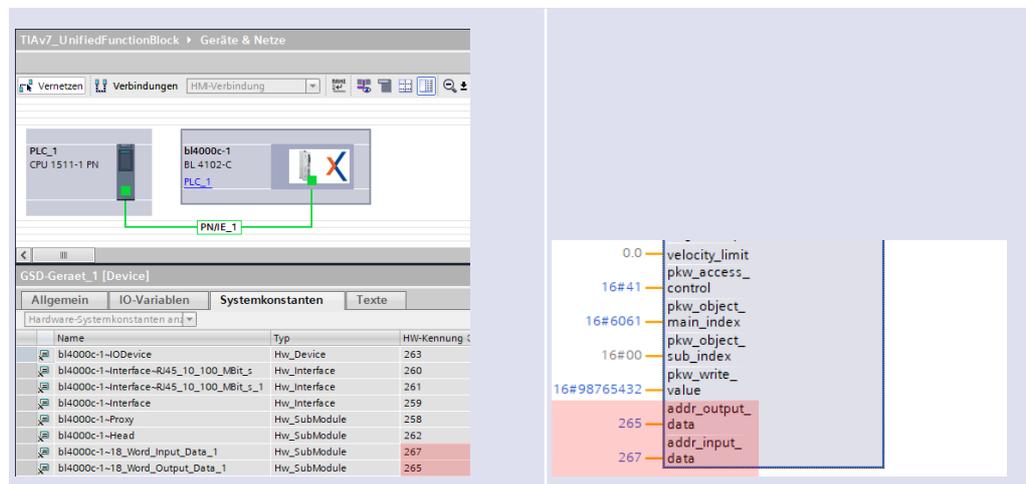


Abbildung 5: Ermittlung der HW-Kennung

Ausgang	Bedeutung
Control by PLC	Dieser Ausgang bildet Bit 9 des <b>Status word 1</b> (Kontrolle durch die SPS) ab. Das entsprechende Bit im <b>Control word 1</b> wird durch den Funktionsbaustein automatisch gesetzt.

## 2.5.1.2 Eingänge/Ausgänge für alle Betriebsarten

Die folgenden Eingänge und Ausgänge können unabhängig von der eingestellten Betriebsart verwendet werden.

Eingang	Bedeutung												
operation_mode	<p>Dieser Eingang legt die Betriebsart des Servoreglers fest. Folgende Betriebsarten können eingestellt werden:</p> <table border="1"> <thead> <tr> <th>Name</th> <th>Wert</th> <th>Bedeutung</th> </tr> </thead> <tbody> <tr> <td>OPMODE_POS</td> <td>10<sub>h</sub></td> <td>Positionieren</td> </tr> <tr> <td>OPMODE_VELO</td> <td>08<sub>h</sub></td> <td>Drehzahlregelung</td> </tr> <tr> <td>OPMODE_TORQUE</td> <td>04<sub>h</sub></td> <td>Drehmomentregelung</td> </tr> </tbody> </table> <p>Siehe auch Abschnitt 5 <i>Betriebsarten</i> auf Seite 68.</p>	Name	Wert	Bedeutung	OPMODE_POS	10 <sub>h</sub>	Positionieren	OPMODE_VELO	08 <sub>h</sub>	Drehzahlregelung	OPMODE_TORQUE	04 <sub>h</sub>	Drehmomentregelung
Name	Wert	Bedeutung											
OPMODE_POS	10 <sub>h</sub>	Positionieren											
OPMODE_VELO	08 <sub>h</sub>	Drehzahlregelung											
OPMODE_TORQUE	04 <sub>h</sub>	Drehmomentregelung											
drive_enable	<p>Gibt den Regler frei, d.h. die Endstufe wird eingeschaltet und der Motor wird gemäß der eingestellten Betriebsart (<b>operation_mode</b>) geregelt. Stellen Sie daher vorher unbedingt sicher, dass der Antrieb richtig parametrisiert ist und ein entsprechender Sollwert gleich Null ist. Der Eingang bedient die State Machine des Servoreglers, siehe auch Abschnitt 4.4.1 <i>Zustandsdiagramm: Zustände</i> auf Seite 63.</p>												
fault_reset	<p>Bei einer steigenden Flanke an diesem Eingang werden anstehende Fehlermeldungen quittiert. Der Eingang entspricht Bit 7 von <i>PNU 967: Control word 1</i>.</p>												

Ausgang	Bedeutung
operation_mode_actual	<p>Aktuelle Betriebsart des Reglers, die über <b>operation_mode</b> vorgegeben wurde. Der Wechsel der Betriebsart erfordert mehrere Zyklen. Erst wenn die eingestellte Betriebsart über dieses Objekt gelesen werden kann, ist der Wechsel abgeschlossen. Siehe Abschnitt 5 <i>Betriebsarten</i> auf Seite 68.</p>
drive_is_enabled	<p>Die Reglerfreigabe des Servoreglers ist aktiv.</p>
fault	<p>Bei gesetztem Ausgang liegt ein Fehler vor. Die Reglerfreigabe wird weggenommen. Der Ausgang wird zurückgesetzt, nachdem der Fehler quittiert wurde.</p>
warning	<p>Bei gesetztem Ausgang liegt eine Warnung im Antrieb vor. Im Gegensatz zu einem Fehler führt eine Warnung nicht zum Abschalten des Reglers. Der Ausgang wird automatisch zurückgesetzt, wenn der Grund für die Warnung verschwunden ist.</p>
digital_inputs	<p>Dieser Ausgang liefert den Zustand der digitalen Eingänge zurück. Er entspricht Parameter 1141.0, siehe Abschnitt 3.3.7 <i>PNU 1141: Digital Inputs</i> auf Seite 31</p>
velo_zero	<p>Bei gesetztem Ausgang steht der Antrieb. Der Ausgang entspricht Bit 13 von <i>PNU 968: Status word 1</i></p>

Ausgang	Bedeutung
velo_reached	Bei gesetztem Ausgang ist die Istdrehzahl im parametrisiertem Toleranzfenster der Soll Drehzahl (Vergleichsdrehzahl). Der Ausgang entspricht Bit 8 von <i>PNU 968: Status word 1</i>
velo_exceeded	Bei gesetztem Ausgang ist die Istdrehzahl größer als die freie Vergleichsdrehzahl. Der Ausgang entspricht Bit 10 von <i>PNU 968: Status word 1</i>
position_actual_value	Dieser Ausgang liefert die aktuelle Position zurück. Er entspricht Parameter 1100.0, siehe Abschnitt 3.3.2 <i>PNU 1100: Position Actual Value</i> auf Seite 29.
velocity_actual_value	Dieser Ausgang liefert die aktuelle Geschwindigkeit zurück. Er entspricht Parameter 1101.0, siehe Abschnitt 3.3.3 <i>PNU 1101: Velocity Actual Value</i> auf Seite 29.
current_actual_value	Dieser Ausgang liefert den aktuellen Strom zurück. Er entspricht Parameter 1102.0, siehe Abschnitt 3.3.4 <i>PNU 1102: Current Actual Value</i> auf Seite 30.

### 2.5.1.3 Eingänge/Ausgänge für die Betriebsart Positionieren

Die folgenden Eingänge und Ausgänge sind in der Betriebsart Positionieren verwendbar. Für weitere Details siehe Abschnitt 5.4 *Betriebsart Positionieren* auf Seite 69.

Eingang	Bedeutung
start_homing	Startet die über <i>PNU 1050: Homing Method</i> festgelegte Referenzfahrt. Voraussetzung ist eine aktive Reglerfreigabe, d.h. der Ausgang <i>drive_enable_ok</i> muss gesetzt sein. Ein Rücksetzen des Eingangs <i>start_homing</i> während der Referenzfahrt bricht diese ohne Fehler ab. Der Eingang entspricht Bit 11 von <i>PNU 967: Control word 1</i> .
start_pos	Eine steigende Flanke signalisiert, dass ein neuer Fahrauftrag übernommen werden soll. Eine fallende Flanke hat keine Auswirkungen. Während einer Referenzfahrt hat dieser Eingang keine Auswirkung. Der Eingang entspricht Bit 6 von <i>PNU 967: Control word 1</i> .
relative	Ist dieser Eingang bei einer steigenden Flanke am Eingang <i>start_pos</i> gesetzt, wird die Positionierung relativ zum aktuellen Lagesollwert ausgeführt. Ist dieser Eingang bei einer steigenden Flanke an <i>start_pos</i> nicht gesetzt, erfolgt eine absolute Positionierung. Der Eingang entspricht Bit 12 von <i>PNU 967: Control word 1</i> .

Eingang	Bedeutung
<b>change_immediatly</b>	Ist dieser Eingang bei einer steigenden Flanke am Eingang <b>start_pos</b> gesetzt, so wird eine laufende Positionierung sofort abgebrochen und durch den neuen Fahrauftrag ersetzt. Ist dieser Eingang bei einer steigenden Flanke an <b>start_pos</b> nicht gesetzt, wird der neue Fahrauftrag an das Ende einer laufenden Positionierung angehängt. In diesem Fall wird der Ausgang <b>target_reached</b> am Ende der laufenden Positionierung nicht gesetzt, sondern erst am Ende der angehängten Positionierung. Der Eingang entspricht Bit 13 von <i>PNU 967: Control word 1</i> .
<b>intermediate_stop</b>	Ist dieser Eingang nicht gesetzt, wird eine gestartete Positionierung abgefahren. Wird der Eingang während einer laufenden Positionierung gesetzt, so wird der Antrieb angehalten und verbleibt in Lageregelung. Die aktuelle Positionierung ist nicht beendet. Sie wird fortgesetzt, wenn der Eingang <b>intermediate_stop</b> zurückgesetzt wird. Während einer Referenzfahrt hat dieser Eingang keine Auswirkung. Der Eingang entspricht dem invertierten Bit 5 von <i>PNU 967: Control word 1</i> .
<b>hold</b>	Ist dieser Eingang gesetzt, wird die laufende Positionierung abgebrochen. Gebremst wird hierbei mit der für diese Positionierung gültigen Bremsbeschleunigung. Nach der Beendigung des Vorgangs wird der Ausgang <b>target_reached</b> nicht gesetzt. Das Rücksetzen des Eingangs hat keine Auswirkung. Während einer Referenzfahrt hat dieser Eingang keine Auswirkung. Wird der Eingang gesetzt, werden Bit 4, 8 und 9 von <i>PNU 967: Control word 1</i> gelöscht.
<b>homing_method</b>	Dieser Eingang legt die Referenzfahrt-Methode fest. Er entspricht Parameter 1050.0, siehe Abschnitt 3.4.8 <i>PNU 1050: Homing Method</i> auf Seite 37.
<b>target_position</b>	Dieser Eingang legt die Zielposition fest. Er entspricht Parameter 1001.0, siehe Abschnitt 3.4.2 <i>PNU 1001: Position Data</i> auf Seite 33.
<b>profile_velocity</b>	Dieser Eingang legt die Profilgeschwindigkeit fest. Entspricht Parameter 1001.1, siehe Abschnitt 3.4.2 <i>PNU 1001: Position Data</i> auf Seite 33
<b>end_velocity</b>	Dieser Eingang legt die Endgeschwindigkeit fest. Er entspricht Parameter 1001.2, siehe Abschnitt 3.4.2 <i>PNU 1001: Position Data</i> auf Seite 33
<b>all_accelerations_positioning</b>	Dieser Eingang legt gleichzeitig die Beschleunigung und die Bremsbeschleunigung für den Verfahrensvorgang fest. Er entspricht Parameter 1001.5, siehe Abschnitt 3.4.2 <i>PNU 1001: Position Data</i> auf Seite 33

Ausgang	Bedeutung
home_valid	Dieser Ausgang ist gesetzt, wenn eine gültige Referenzposition vorliegt. Der Ausgang ist während einer laufenden Referenzfahrt nicht gesetzt. Der Ausgang entspricht Bit 11 von <i>PNU 968: Status word 1</i>
target_reached	Dieser Ausgang wird gesetzt, wenn die aktuelle Position nach abgeschlossener Positionierung im Zielfenster steht. Der Ausgang entspricht Bit 10 von <i>PNU 968: Status word 1</i>
following_error	Dieser Ausgang wird gesetzt, wenn ein Schleppfehler vorliegt. Der Ausgang entspricht dem invertierten Bit 10 von <i>PNU 968: Status word 1</i>
pos_task_ack	Dieser Ausgang ist gelöscht, wenn der Regler einen neuen Fahrauftrag verarbeiten kann. Der Ausgang entspricht dem invertierten Bit 12 von <i>PNU 968: Status word 1</i>

### 2.5.1.4 Eingänge/Ausgänge für die Betriebsart Drehzahlregelung

Die folgenden Eingänge und Ausgänge sind in der Betriebsart Drehzahlregelung verwendbar. Siehe auch Abschnitt 5.3 *Betriebsart Drehzahlregelung* auf Seite 69.

Eingang	Bedeutung
target_velocity	Dieser Eingang legt die Sollgeschwindigkeit fest. Er entspricht Parameter 1010.2, siehe Abschnitt 3.5.1 <i>PNU 1010: Target Velocity</i> auf Seite 46
current_limit	Dieser Eingang legt den maximalen Strom in mA fest, der in der Betriebsart Drehzahlregelung verwendet wird. Siehe Abschnitt 3.5.3 <i>CAN-Objekt 2415h: current_limitation</i> auf Seite 47.

Ausgang	Bedeutung
---	Es sind keine Ausgänge vorhanden, die nur in dieser Betriebsart verwendet werden.

### 2.5.1.5 Eingänge/Ausgänge für die Betriebsart Drehmomentregelung

Die folgenden Eingänge und Ausgänge sind in der Betriebsart Drehmomentregelung verwendbar. Siehe auch Abschnitt 5.2 *Betriebsart Drehmomentregelung* auf Seite 68.

Eingang	Bedeutung
target_torque	Dieser Eingang legt den Sollstrom fest. Er entspricht dem CAN-Objekt 6071 <sub>h_00h</sub> ( <i>target_torque</i> ) und wird in Promille des Nennstroms vorgegeben.
velocity_limit	Über diesen Eingang kann die maximale Drehzahl begrenzt werden. Siehe Abschnitt 3.6.2 <i>CAN-Objekt 2416h: speed_limitation</i> auf Seite 48.

Ausgang	Bedeutung
---	Es sind keine Ausgänge vorhanden, die nur in dieser Betriebsart verwendet werden.

### 2.5.1.6 Eingänge/Ausgänge für den Tippbetrieb (Jogging)

In diesem Abschnitt werden die Eingänge und Ausgänge für den Tippbetrieb beschrieben.

Eingang	Bedeutung
jog_pos	Der Antrieb bewegt sich in positiver Richtung, solange dieser Eingang gesetzt ist. Die Geschwindigkeit wird dabei über den Eingang <i>jogging_velocity</i> vorgegeben. Zum Beschleunigen und Bremsen wird die im Positionssatz TIPP 0 eingestellte Beschleunigung/ Bremsbeschleunigung verwendet. Während einer Referenzfahrt hat dieser Eingang keine Auswirkung. Der Eingang entspricht Bit 8 von <i>PNU 967: Control word 1</i> .
jog_neg	Der Antrieb bewegt sich in negativer Richtung, solange dieser Eingang gesetzt ist. Die Geschwindigkeit wird dabei über den Eingang <i>jogging_velocity</i> vorgegeben. Zum Beschleunigen und Bremsen wird die im Positionssatz TIPP 0 eingestellte Beschleunigung/ Bremsbeschleunigung verwendet. Während einer Referenzfahrt hat dieser Eingang keine Auswirkung. Der Eingang entspricht Bit 9 von <i>PNU 967: Control word 1</i> .
jogging_velocity	Dieser Eingang legt die Geschwindigkeit des Tippvorgangs fest. Er entspricht dem Parameter 1040.0, siehe Abschnitt 3.7.1 <i>PNU 1040: Symmetrical Jogging</i> auf Seite 49.

Ausgang	Bedeutung
---	Es sind keine Ausgänge vorhanden, die nur in dieser Betriebsart verwendet werden.

### 2.5.1.7 Eingänge/Ausgänge für den Parameterzugriff (PKW)

Der Parameterzugriff ermöglicht das asynchrone Lesen/Schreiben jeder beliebigen PNU bzw. jeden beliebigen CAN-Objekts, ohne dass dieses zyklisch übertragen werden muss. Die Ein- und Ausgänge bilden den Zugriff auf die PNU 2000, siehe Abschnitt 3.8.3 *PNU 2000: PKW Access* auf Seite 54 ab.

Eingang	Bedeutung
pkw_object_main_index	Dieser Eingang legt den Hauptindex der PNU oder des CAN-Objekts fest, das gelesen oder geschrieben werden soll. Er entspricht Byte 1 und 2 des Parameters 2000.0.
pkw_object_sub_index	Dieser Eingang legt den Subindex der PNU oder des CAN-Objekts fest, das gelesen oder geschrieben werden soll. Er entspricht Byte 3 des Parameters 2000.0.
pkw_write_value	Dieser Eingang legt den Wert fest, der in die PNU oder das CAN-Objekts geschrieben werden soll. Er entspricht Byte 4 bis 7 des Parameters 2000.0.
pkw_access_control	Über diesen Eingang wird das Lesen und Schreiben gesteuert. Er entspricht Byte 0 des Parameters 2000.0.

Ausgang	Bedeutung
pkw_read_value	Dieser Ausgang liefert den Wert, der aus der PNU oder aus dem CAN-Objekts gelesen wurde. Er entspricht Byte 4 bis 7 des Parameters 2000.0.
pkw_access_status	Über diesen Ausgang spiegelt der Regler den Wert aus <b>pkw_access_control</b> , um anzuzeigen, dass das Lesen und Schreiben durchgeführt wurde. Er entspricht Byte 0 des Parameters 2000.0.

# 3 Parameter

Dieser Abschnitt beschreibt die unter PROFINET verfügbaren Parameter (PNU), mit denen der Antrieb an die jeweilige Applikation angepasst werden kann.

## › Darstellung der Parameter

Alle Parameter des Antriebs werden in einer einheitlichen Darstellungsweise beschrieben. Handelt es sich beim Parameter um einen einfachen Datentyp, wird dieser folgendermaßen dargestellt:

PNU	<b>Index (dezimal)</b>		
Name	<b>Name des Parameters</b>		
Info	Einheit	rw	Datentyp
Value	Wertebereich	Defaultwert	

Handelt es sich beim Parameter um einen zusammengesetzten Datentyp (ARRAY/RECORD), wird dieser folgendermaßen dargestellt:

PNU	<b>Index (dezimal)</b>		
Name	<b>Name der Parametergruppe</b>		
Type	Object Code		Max
Sub-Index	<b>Subindex (dezimal)</b>		
Name	<b>Name des Parameters</b>		
Info	Einheit	rw	Datentyp
Value	Wertebereich	Defaultwert	

Die einzelnen Felder haben folgende Bedeutung:

Feld	Bedeutung
PNU	Der Hauptindex des beschriebenen Parameters.
Subindex (hexadezimal)	Der Subindex des beschriebenen Parameters. Wenn dieser nicht angegeben ist, ist der Subindex Null.
Name der Parametergruppe	Klartext-Name der Parametergruppe.
Name des Parameters	Klartext-Name des Parameters.
Object Code	Angabe, ob es sich um einen einfachen oder zusammengesetzten Datentyp handelt: <ul style="list-style-type: none"> <li>• VAR: Einfacher Datentyp</li> <li>• ARRAY: Gruppe von Parametern, die alle den identischen Datentyp haben.</li> <li>• RECORD: Gruppe von Parametern, die unterschiedliche Datentypen haben.</li> </ul>
Max	Höchster Subindex der Gruppe.
Datentyp	Datentyp des Parameters oder des ARRAYS.
Einheit	Physikalische Einheit des Parameters.
Zugriff	Angabe ob der Parameter gelesen (ro), geschrieben (wo) oder gelesen und geschrieben (rw) werden darf.

Feld	Bedeutung
Wertebereich	Bereich, in dem zulässige Werte für diesen Parameter liegen.
Defaultwert	Wert, der im Auslieferungszustand wirksam ist.

## 3.1 Physikalische Einheiten

### 3.1.1 Übersicht

Die Skalierung der Parameter, die über den Bus übertragen werden, kann für die physikalischen Größen Position, Geschwindigkeit und Beschleunigung angepasst werden. Bei der Verwendung des Beispiel-Funktionsbausteins ist dies nicht nötig. Dieser basiert auf der Standard-Einstellung der physikalischen Einheiten. Standardmäßig sind folgende Einheiten eingestellt:

Größe	Bezeichnung	Einheit	Erklärung
Position/Lage	Phys. Einheit Lage	Tausendstel Umdrehung	1000 = 1 Umdrehung
Geschwindigkeit	Phys. Einheit Geschwindigkeit	min <sup>-1</sup>	Umdrehung pro Minute
Beschleunigung	Phys. Einheit Beschleunigung	(min <sup>-1</sup> )/s	Drehzahlerhöhung in Umdrehung pro Minute pro Sekunde

### 3.1.2 Parametrierung der physikalischen Einheiten

Die physikalischen Einheiten können über folgenden Menüpunkt des Parametrierprogramms Metronix ServoCommander® eingestellt werden:

[Parameter/Feldbus/PROFINET/Anzeigeeinheiten](#)

Die Parameter für die physikalischen Einheiten sollten einmalig eingestellt und während einer laufenden Applikation nicht geändert werden. Sie können für Position/Lage, Geschwindigkeit und Beschleunigung unabhängig voneinander eingestellt werden. Der Getriebefaktor und eine Vorschubkonstante werden als separate Parameter angegeben.

Aus den eingestellten physikalischen Einheiten werden Umrechnungsfaktoren berechnet, die aus Zähler und Nenner bestehen. Zähler und Nenner dürfen jeweils nicht größer als 32 Bit werden. Kommt es bei der Eingabe der Faktoren hier zu einem Überlauf, wird der Wert nicht angenommen. In diesem Fall müssen die Faktoren bzw. die physikalischen Einheiten korrigiert werden.

Abbildung 6: Einstellung der physikalischen Einheiten für PROFINET

Es ist zu beachten, dass einige Größen nicht immer sinnvoll genutzt werden können. In einem rein rotatorischen System wird z.B. keine Vorschubkonstante benötigt. Darüber hinaus verfügt die Vorschubkonstante über eine physikalische Einheit. Ist diese nicht passend parametrierbar, dann wird die Vorschubkonstante nicht berücksichtigt. Der Wert der Vorschubkonstante wird für die jeweilige physikalische Einheit ignoriert, wenn die Vorschubkonstante eine translatorische Einheit besitzt und für die physikalische Größe eine rotatorische Einheit ausgewählt ist.

## Beispiel

Die folgenden Fälle verdeutlichen die Berücksichtigung der Vorschubkonstante:

1. Lage in **Umdrehungen**, Vorschubkonstante in **mm/Umdrehung**:  
=> Die Vorschubkonstante wird ignoriert.
2. Lage in **mm**, Vorschubkonstante ohne Einheit:  
=> Die Vorschubkonstante wird wie ein Getriebefaktor berücksichtigt.
3. Lage in **mm**, Vorschubkonstante in **µm/Umdrehung**:  
=> Die Vorschubkonstante wird mit dem Faktor 1000 berücksichtigt.

Falls durch die Einstellung der physikalischen Einheiten der interne oder der von außen eingegebene Wert durch die Umrechnung nicht mehr darstellbar ist, wird ein Fehler ausgelöst. In diesem Fall ist die Einstellung der physikalischen Einheiten zu prüfen.

Bei der Parametrierung der Anzeigeeinheiten können Übergangszustände auftreten, die zu einem Überlauf der physikalischen Einheiten führen. In diesem Fall wird der Fehler 22-4 generiert. Ob die Parametrierung tatsächlich ungültig ist, lässt sich in diesem Fall nur durch Speichern und anschließendem Reset feststellen. Liegt danach kein Fehler 22-4 mehr vor, dann sind die Einstellungen gültig.

Getriebefaktor und Vorschubkonstante können nur positive Werte annehmen. Falls die Orientierung der Applikation gedreht werden soll, so kann dies über den Getriebefaktor des Winkelgebers in der Parametriersoftware Metronix ServoCommander® erreicht werden.

## 3.2 Übersicht

Die folgende Tabelle liefert eine Übersicht über die aktuell implementierten PNUs:

PNU	Sub-index	Beschreibung / Link	Typ	Zugriff
1000	0	<i>Position Set Number</i>	UINT16	rw
1001	-	<i>Position Data</i>		
	0	<i>Target Position (Zielposition)</i>	INT32	rw
	1	<i>Profile Velocity (Fahrgeschwindigkeit)</i>	INT32	rw
	2	<i>End Velocity (Endgeschwindigkeit)</i>	INT32	rw
	3	<i>Acceleration Positioning (Beschleunigungsrampe)</i>	UINT32	rw
	4	<i>Deceleration Positioning (Bremsrampe)</i>	UINT32	rw
	5	<i>All Accelerations Positioning (Beschleunigungs- und Bremsrampe)</i>	UINT32	rw
1002	0	<i>Start Set Number</i>	UINT8	rw
1003	0	<i>Position Profile Type</i>	UINT16	rw
1004	0	<i>Override Factor</i>	UINT16	rw
1005	-	<i>Software Position Limits</i>		
	0	<i>Lower software position limit switch (Unterer Software Endschalter)</i>	INT32	rw
	1	<i>Upper software position limit switch (Oberer Software Endschalter)</i>	INT32	rw
1006	-	<i>Rotary Axis</i>		
	0	<i>Rotary Axis Mode (Modus Rundachse) Rotary Axis Mode (Modus Rundachse)</i>	UINT8	rw
	1	<i>Lower rotary axis limit (Untere Rundachsgrenze)</i>	INT32	rw
	2	<i>Upper rotary axis limit (Obere Rundachsgrenze)</i>	INT32	rw
1010	0	<i>Target Velocity</i>	INT32	rw
1011	-	<i>Accelerations for Velocity Control</i>		
	0	<i>Acceleration Velocity Control (Beschleunigungsrampe Drehzahlregelung)</i>	UINT32	rw
	1	<i>Deceleration Velocity Control (Bremsrampe Drehzahlregelung)</i>	UINT32	rw
	2	<i>All Accelerations Velocity Control (Beschleunigung und Bremsbeschleunigung für Drehzahlregelung)</i>	UINT32	rw
1040	-	<i>Symmetrical Jogging</i>		
	0	<i>Symmetrical Jogging Velocity (Tippgeschwindigkeiten, symmetrisch)</i>	INT32	rw
	1	<i>Symmetrical Jogging Accelerations (Tippbeschleunigungen, symmetrisch)</i>	UINT32	rw

PNU	Sub-index	Beschreibung / Link	Typ	Zugriff
1041	-	<i>Jogging Positive</i>		
	0	<i>Jogging Velocity Positive</i>	INT32	rw
	1	<i>Jogging Acceleration Positive (Beschleunigung für Tippen positiv)</i>	UINT32	rw
	2	<i>Jogging Deceleration Positive (Bremsbeschleunigung für Tippen positiv)</i>	UINT32	rw
	3	<i>Symmetrical Jogging Accelerations Positive (Beschleunigungen für Tippen in positiver Richtung, symmetrisch)</i>	UINT32	rw
1042	-	<i>Jogging Negative</i>		
	0	<i>Jogging Velocity Negative (Tippgeschwindigkeit negativ)</i>	INT32	rw
	1	<i>Jogging Acceleration Negative (Beschleunigung für Tippen negativ)</i>	UINT32	rw
	2	<i>Jogging Deceleration Negative (Bremsbeschleunigung für Tippen negativ)</i>	UINT32	rw
	3	<i>Symmetrical Jogging Accelerations Negative (Beschleunigungen für Tippen in negativer Richtung, symmetrisch)</i>	UINT32	rw
1050	0	<i>PNU 1050: Homing Method</i>	INT8	rw
1051	0	<i>Home Offset</i>	INT32	rw
1060	0	<i>Thread Speed (Einrichtdrehzahl)</i>	INT32	rw
1100	0	<i>Position Actual Value</i>	INT32	ro
1101	0	<i>Velocity Actual Value</i>	INT32	ro
1102	0	<i>Current Actual Value</i>	INT32	ro
1110	-	<i>Sampling Positions</i>		
	0	<i>Sampling Position Rising Edge (Gespeicherte Istposition auf steigender Flanke)</i>	INT32	ro
	1	<i>Sampling Position Falling Edge (Gespeicherte Istposition auf fallender Flanke)</i>	INT32	ro
1141	0	<i>Digital Inputs</i>	UINT32	ro
1270	-	<i>Position Control Parameters</i>		
	2	<i>Position error tolerance window (Totbereich Lageregler)</i>	UINT32	rw
1271	-	<i>Position Window Data</i>		
	0	<i>Target Window (Fenster für "Ziel erreicht")</i>	UINT32	rw
1272	-	<i>Following Error Data</i>		
	0	<i>Following Error Window (Schleppfehler-Fenster)</i>	UINT32	rw
1273	-	<i>Position Error Data</i>		
	0	<i>Position Error Limit (Abschaltgrenze Schleppfehler)</i>	UINT32	rw

PNU	Sub-index	Beschreibung / Link	Typ	Zugriff
1500	0	<i>Operating Mode</i>	UINT8	ro
1600	0	<i>Last Error Code</i>	UINT16	ro
2000	0	<i>PKW Access</i>	2xUINT32	rw
2010	-	<i>Placeholder</i>		
	0	<i>8 Bit</i>	UINT8	rw
	1	<i>16 Bit</i>	UINT16	rw
	2	<i>32 Bit</i>	UINT32	rw
2011	0	<i>Placeholder</i>	UINT32	rw

Vom Beispiel-Funktionsbaustein werden zusätzlich folgende CAN-Objekte verwendet:

Main index	Sub-index	Beschreibung	Typ	Zugriff
6071 <sub>h</sub>	00 <sub>h</sub>	<i>target_torque</i>	INT16	rw
6077 <sub>h</sub>	00 <sub>h</sub>	<i>torque_actual_value</i>	UINT16	ro
2415 <sub>h</sub>	-	<i>current_limitation</i>		
	01 <sub>h</sub>	<i>limit_current</i>	INT32	rw
2416 <sub>h</sub>	-	<i>speed_limitation</i>		
	01 <sub>h</sub>	<i>limit_speed</i>	INT32	rw

In den folgenden Abschnitten werden die Parameter - gruppiert nach ihrem Verwendungszweck - ausführlich beschrieben.

## 3.3 PNUs für alle Betriebsarten

In diesem Abschnitt sind Parameternummern aufgelistet, die in aller Regel unabhängig von der Betriebsart verwendet werden können, wie beispielsweise Istwerte.

### 3.3.1 PNU 1500: Operating Mode

Dieser herstellerspezifische Parameter erlaubt das Setzen/Lesen der Betriebsart. Siehe hierzu auch Abschnitt 5 *Betriebsarten* auf Seite 68.

PNU	<b>1500</b>		
Name	<b>Operating Mode</b>		
Info	--	rw	UINT8
Value	4 <sub>h</sub> , 8 <sub>h</sub> , 10 <sub>h</sub>	--	

Wert	Bedeutung
4 <sub>h</sub>	Drehmomentregelung
8 <sub>h</sub>	Drehzahlregelung
10 <sub>h</sub>	Positionieren

### 3.3.2 PNU 1100: Position Actual Value

Über diesen Parameter wird der Lageistwert zurückgegeben. Dieser ist in der für PROFINET eingestellten physikalischen Einheit skaliert. Bei der Berechnung des Lageistwertes kann es zu Fehlern kommen, da die interne Lage des Servoreglers einen größeren darstellbaren Wertebereich besitzt, als übertragen werden kann. Dies hängt aber von den eingestellten physikalischen Einheiten sowie dem Getriebefaktor und der Vorschubkonstante ab. Setzen Sie sich ggf. mit dem technischen Support in Verbindung.

PNU	<b>1100</b>		
Name	<b>Position Actual Value</b>		
Info	Physikalische Einheit Lage	ro	INT32
Value	--	--	

### 3.3.3 PNU 1101: Velocity Actual Value

Über diesen Parameter wird der Drehzahlwert zurückgegeben. Dieser ist in der für PROFINET eingestellten physikalischen Einheit skaliert.

PNU	<b>1101</b>		
Name	<b>Velocity Actual Value</b>		
Info	Physikalische Einheit Geschwindigkeit	ro	INT32
Value	--	--	

### 3.3.4 PNU 1102: Current Actual Value

Über diesen Parameter wird der Wirkstrom-Istwert gelesen. Dieser wird bezogen auf den Motornennstrom zurückgegeben.

PNU	<b>1102</b>		
Name	<b>Current Actual Value</b>		
Info	Promille bezogen auf den Motornennstrom	ro	INT32
Value	--	--	

### 3.3.5 CAN-Objekt 6077<sub>h</sub>: torque\_actual\_value

Über dieses Objekt kann der Drehmomenten-Istwert des Motors in Tausendstel des Nennmomentes ausgelesen werden.

Index	<b>6077<sub>h</sub></b>		
Name	<b>torque_actual_value</b>		
Info	‰ (1000 = motor_rated torque)	ro	PDO INT16
Value	--	--	

### 3.3.6 PNU 1110: Sampling Positions

Diese Parameternummern liefern die Positionen, die auf der steigenden bzw. fallenden Flanke des so genannten Sample-Eingangs gespeichert wurden. Über das Parametrierprogramm Metronix ServoCommander® kann im Fenster „Parameter - IOs - Digitale Eingänge“ der gewünschte Eingang DIN8 oder DIN9 für diesen Zweck parametrieren werden.

Index	<b>1110</b>		
Name	<b>Sampling Positions</b>		
Type	RECORD	1	
Sub-Index	<b>00</b>		
Name	<b>Sampling Position Rising Edge (Gespeicherte Istposition auf steigender Flanke)</b>		
Info	Physikalische Einheit Lage	ro	INT32
Value	--	--	
Sub-Index	<b>01</b>		
Name	<b>Sampling Position Falling Edge (Gespeicherte Istposition auf fallender Flanke)</b>		
Info	Physikalische Einheit Lage	ro	INT32
Value	--	--	

### 3.3.7 PNU 1141: Digital Inputs

Über diesen Parameter wird der Zustand der digitalen Eingänge gelesen. Die verfügbaren digitalen Eingänge hängen von der Parametrierung des Servoreglers ab.

PNU	<b>1141</b>		
Name	<b>Digital Inputs</b>		
Info		ro	UINT32
Value	--	--	

Bit	Bedeutung
0	Reserviert (= 0)
1	DIN 0
2	DIN 1
3	DIN 2
4	DIN 3
5	DIN 4
6	DIN 5 (digitale Reglerfreigabe)
7	DIN 6 (Endschalter 0 links = negative Drehrichtung)
8	DIN 7 (Endschalter 1 rechts = positive Drehrichtung)
9	DIN 8 (Default: Start-Eingang)
10	DIN 9 (Default: Sample-Eingang)
11...31	Reserviert

### 3.3.8 PNU 1600: Last Error Code

Unter dieser Parameternummer wird der zuletzt ausgelöste Fehler ausgegeben. Wenn kein Fehler vorliegt, liefert die PNU 0. Antriebsfehler werden auch über die Diagnoseschnittstelle an die Steuerung gemeldet, siehe Abschnitt 4.6 *Diagnose - Alarmer* auf Seite 67.

PNU	<b>1600</b>		
Name	<b>Last Error Code</b>		
Info	--	ro	UINT16
Value	siehe Tabelle	0	

Bit	Bedeutung
0...3	Unterfehlernummer (0...9)
4...15	Hauptfehlernummer (1...96)

## 3.4 PNUs zur Betriebsart Positionieren

In diesem Abschnitt werden die Parameter beschrieben, die für die Betriebsart Positionieren benötigt werden.

### 3.4.1 PNU 1000: Position Set Number

Über diesen Parameter kann der Positionsdatensatz ausgewählt werden, in den die über PROFINET übertragenen Daten eingetragen werden. Über diesen Parameter besteht Zugriff auf alle Positionsdatensätze des Servoreglers. Der Positionsdatensatz für PROFINET ist speicherbar und kann auch über das Parametrierprogramm Metronix ServoCommander® parametrierbar werden. Dadurch können Parameter fest vorgegeben werden, die in einer Applikation während des Betriebs nicht geändert werden müssen. Beispielsweise können die Beschleunigungen einmalig eingetragen werden und müssen dann nicht übertragen werden.

Auf die speziellen Positionsdatensätze für Referenzfahrt oder Tippen kann über diesen Parameter ebenfalls zugegriffen werden. Aufgrund der speziellen Struktur der Datensätze empfiehlt sich hier aber die Parametrierung über das Parametrierprogramm ServoCommander®.

PNU	<b>1000</b>		
Name	<b>Position Set Number</b>		
Info	--	rw	UINT16
Value	0...267	266 (PROFINET)	

Wert	Bedeutung
0...255	Standard-Positionsdatensätze
256	Referenzfahrt Phase 0
257	Referenzfahrt Phase 1
258	Referenzfahrt Phase 2
259	Tippen positiv
260	Tippen negativ
261...265	Reserviert
266	Positionsdatensatz PROFINET
267	Reserviert

### 3.4.2 PNU 1001: Position Data

Unter dieser Parameternummer können Parameter des ausgewählten Positionssatzes (PNU 1000) angesprochen werden. Es sind die folgenden Parameter verfügbar:

- Zielposition
- Fahrgeschwindigkeit
- Endgeschwindigkeit
- Beschleunigung und Bremsbeschleunigung, jeweils einzeln oder als Kombination für beide Beschleunigungen

Die Daten werden in der aktuell eingestellten physikalischen Einheit parametrierung (siehe Abschnitt 3.1 *Physikalische Einheiten* auf Seite 24).

Unter dieser PNU können auch Parameter z.B. für das Tippen parametrierung werden. Hierzu ist zuerst die Positionssatznummer entsprechend einzustellen, dann kann beispielsweise die Geschwindigkeit beim Tippen über die Fahrgeschwindigkeit eingestellt werden.

Index	<b>1001</b>		
Name	<b>Position Data</b>		
Type	RECORD		5

Sub-Index	<b>00</b>		
Name	<b>Target Position (Zielposition)</b>		
Info	Physikalische Einheit Lage	rw	INT32
Value	--	0	

Sub-Index	<b>01</b>		
Name	<b>Profile Velocity (Fahrgeschwindigkeit)</b>		
Info	Physik. Einheit Geschwindigkeit	rw	INT32
Value	--	1000 U/min	

Sub-Index	<b>02</b>		
Name	<b>End Velocity (Endgeschwindigkeit)</b>		
Info	Physik. Einheit Geschwindigkeit	rw	INT32
Value	--	0	

Sub-Index	<b>03</b>		
Name	<b>Acceleration Positioning (Beschleunigungsrampe)</b>		
Info	Physik. Einheit Beschleunigung	rw	UINT32
Value	--	10.000 (U/min)/s	

Sub-Index	<b>04</b>		
Name	<b>Deceleration Positioning (Bremsrampe)</b>		
Info	Physik. Einheit Beschleunigung	rw	UINT32
Value	--	10.000 (U/min)/s	

Der Parameter **All Accelerations Positioning** erlaubt den gleichzeitigen Zugriff auf Beschleunigungs- und Bremsrampe. Falls beide Parameter den gleichen Wert haben sollen, muss nur ein Datenwert übertragen werden. Intern wird dieser dann auf beide Beschleunigungen geschrieben. Es ist zu beachten, dass beim Lesen immer nur der aktuelle Wert der Beschleunigungsrampe gelesen wird. Der Anwender muss selbst sicherstellen, dass das Lesen eines Wertes ausreicht. Dies kann beispielsweise durch einmaliges Lesen und anschließendes Zurückschreiben dieses Wertes erreicht werden.

Sub-Index	<b>05</b>		
Name	<b>All Accelerations Positioning (Beschleunigungs- und Bremsrampe)</b>		
Info	Physik. Einheit Beschleunigung	rw	UINT32
Value	--	10.000 (U/min)/s	

### 3.4.3 PNU 1002: Start Set Number

Über diesen Parameter kann der Positionsdatensatz ausgewählt werden, der bei einem Startbefehl zur Positionierung über das **Control word 1** gestartet wird. Der Servoregler verfügt über 256 speicherbare Standard-Positionsdatensätze (0...255). Um die **Start Set Number** auf 8 Bit zu begrenzen, wird der PROFINET-Positionsdatensatz unter dem letzten Index angesprochen. Daher kann der Standard-Positionsdatensatz 255 über den Bus selbst nicht gestartet werden.

PNU	<b>1002</b>		
Name	Start Set Number		
Info	--	rw	UINT8
Value	0...255	255 (PROFINET)	

Wert	Bedeutung
0...254	Standard-Positionsdatensätze
255	Positionsdatensatz PROFINET

### 3.4.4 PNU 1003: Position Profile Type

Über diesen Parameter kann die Ruckbegrenzung der Positionsdatensätze zwischen 0 und einer automatischen Bestimmung umgeschaltet werden. Bei der automatischen Bestimmung wird die Filterzeit für die Ruckbegrenzung beim Aufruf des Positionssatzes in Abhängigkeit von Beschleunigung und Fahrgeschwindigkeit stets neu bestimmt. Dadurch wird die ruckfreie Zeit bei Änderung dieser Parameter ebenfalls aktualisiert und muss nicht in der Steuerung berechnet werden. Der „Aufruf“ des Positionsdatensatzes unterscheidet sich von dem „Start“ des Positionsdatensatzes. Der eigentliche Start des Positionsdatensatzes kann durch die entsprechende Option z.B. bis zum Ende einer aktuell laufenden Positionierung verzögert werden. Der „Aufruf“ bezieht sich hier auf den Zeitpunkt, zu dem das Start-Kommando über den Feldbus übertragen wird.

PNU	<b>1003</b>		
Name	<b>Position Profile Type</b>		
Info	--	rw	UINT16
Value	0...1	0	

Wert	Bedeutung
0	Ruckfreie Filterzeit für Positionsdatensatz = 0. Dies wird einmalig nach Reset bzw. beim Schreiben dieses Parameters ausgeführt. Wenn die ruckfreie Filterzeit anschließend verändert wird (z.B. über das Parametrierprogramm), dann bleibt die Filterzeit bis zum erneuten Schreiben dieses Parameters oder einem Reset wie eingetragen wirksam.
1	Ruckfreie Filterzeit für Positionsdatensatz beim Aufruf automatisch bestimmen.

### 3.4.5 PNU 1004: Override Factor

Mit diesem Parameter kann die Fahrgeschwindigkeit einer Positionierung jederzeit verändert werden. Durch Änderung des Override Faktors beispielsweise auf 50 % wird die Fahrgeschwindigkeit einer laufenden Positionierung auf die Hälfte reduziert. Dieser Wert ist nach Reset stets 100 % und kann nicht durch Speichern des Parametersatzes dauerhaft gesichert werden.

Die Änderung wirkt sich nicht auf die Beschleunigung aus. Diese bleibt unverändert. Änderungen des Overrides während einer Bremsphase haben damit keine Auswirkung mehr auf die laufende Positionierung.

PNU	<b>1004</b>		
Name	<b>Override Factor</b>		
Info	‰ (1000 = 100 %)	rw	UINT16
Value	0...2000	1000	

### 3.4.6 PNU 1005: Software Position Limits

Diese Parameternummer schreibt und liest die Software-Positionsgrenzen. Diese haben die Funktion von Software-Endschaltern. Sie sind nur in der Betriebsart Positionieren wirksam. Wenn die Zielposition hinter den Software-Endschaltern liegt, dann wird die Positionierung nicht gestartet. Bei entsprechender Parametrierung wird in diesem Fall eine Meldung ausgelöst.

Index	<b>1005</b>		
Name	<b>Software Position Limits</b>		
Type	ARRAY		1
Sub-Index	<b>00</b>		
Name	<b>Lower software position limit switch (Unterer Software Endschalter)</b>		
Info	Physikalische Einheit Lage	rw	INT32
Value	--		-2147483648
Sub-Index	<b>01</b>		
Name	<b>Upper software position limit switch (Oberer Software Endschalter)</b>		
Info	Physikalische Einheit Lage	rw	INT32
Value	--		2147483647

### 3.4.7 PNU 1006: Rotary Axis

Mit diesen Parameternummern werden der Rundachsmodus und dessen Grenzen parametrierbar. Bei aktiver Rundachse werden Lagesoll- und -istwert auf die Rundachsgrenzen limitiert. Die obere und untere Grenze "fallen aufeinander". Beispielsweise sind für einen Rundachsbereich von 1 Umdrehung als untere Grenze 0.0 U und als obere Grenze 1.0 U einzustellen.

Dieser Modus hat nur Einfluss auf die Sollwertgenerierung in der Betriebsart Positionierung. Dabei verhindert z.B. der Modus "Feste Drehrichtung positiv" nicht Bewegungen in negative Richtung. Der Lageregler liefert weiterhin Sollwerte in negative Richtung. Die Sollwertgenerierung in anderen Betriebsarten ist nicht betroffen.

Index	<b>1006</b>		
Name	<b>Rotary Axis</b>		
Type	RECORD		2
Sub-Index	<b>00</b>		
Name	<b>Rotary Axis Mode (Modus Rundachse)</b>		
Info	--	rw	UINT8
Value	--	0	

Wert	Bedeutung
0	Aus
1	Kürzester Weg
2	Drehrichtung aus Positionssatz
3	Feste Drehrichtung „Positiv“
4	Feste Drehrichtung „Negativ“

Sub-Index	<b>01</b>		
Name	<b>Lower rotary axis limit (Untere Rundachsgrenze)</b>		
Info	Physikalische Einheit Lage	rw	INT32
Value	--	-2147483648	

Sub-Index	<b>02</b>		
Name	<b>Upper rotary axis limit (Obere Rundachsgrenze)</b>		
Info	Physikalische Einheit Lage	rw	INT32
Value	--	2147483647	

### 3.4.8 PNU 1050: Homing Method

Für eine Referenzfahrt werden eine Reihe unterschiedlicher Methoden bereitgestellt. Unter dieser Parameternummer kann die für die Applikation benötigte Variante ausgewählt werden. Es gibt vier mögliche Referenzfahrt-Signale: den negativen und positiven Endschalter, den Referenzschalter und den (periodischen) Nullimpuls des Winkelgebers.

Außerdem kann der Servoregler sich ganz ohne zusätzliches Signal auf den negativen oder positiven Anschlag referenzieren. Wenn über das Objekt [Homing Method](#) eine Methode zum Referenzieren bestimmt wird, so werden hiermit folgende Einstellungen gemacht:

- Die Referenzquelle (neg./pos. Endschalter, der Referenzschalter, neg. / pos. Anschlag)
- Die Richtung und der Ablauf der Referenzfahrt
- Die Art der Auswertung des Nullimpulses vom verwendeten Winkelgeber

Eine detaillierte Beschreibung der Referenzfahrt-Methoden findet sich im Abschnitt 3.4.10 *Referenzfahrt-Abläufe* auf Seite 39.

	Richtung	Ziel	Bezugspunkt für Null
-18	positiv	Anschlag	Anschlag
-17	negativ	Anschlag	Anschlag
-2	positiv	Anschlag	Nullimpuls
-1	negativ	Anschlag	Nullimpuls
1	negativ	Endschalter	Nullimpuls
2	positiv	Endschalter	Nullimpuls
7	positiv	Referenzschalter	Nullimpuls
11	negativ	Referenzschalter	Nullimpuls
17	negativ	Endschalter	Endschalter
18	positiv	Endschalter	Endschalter
23	positiv	Referenzschalter	Referenzschalter
27	negativ	Referenzschalter	Referenzschalter
32	negativ	Nullimpuls	Nullimpuls
33	positiv	Nullimpuls	Nullimpuls
34		Keine Fahrt	Aktuelle Ist-Position

### 3.4.9 PNU 1051: Home Offset

Unter dieser Parameternummer wird der Abstand der Referenzposition (Nullposition) zu dem Bezugspunkt einer Referenzfahrt angegeben. Positive Werte verschieben den Nullpunkt in positiver Richtung vom Bezugspunkt ausgehend. Der Parameter entspricht dem Parameter Offset Startposition im Metronix ServoCommander®.

PNU	<b>1050</b>		
Name	<b>Home Offset</b>		
Info	Physikalische Einheit Lage	rw	INT32
Value		0	

### 3.4.10 Referenzfahrt-Abläufe

#### 3.4.10.1 Methode -17 und -18: Anschlag

Bei dieser Methode bewegt sich der Antrieb in positiver (-18) oder negativer (-17) Richtung, bis er den Anschlag erreicht. Im Normalfall wird eine Erhöhung des  $i^2t$ -Werts um 50 % als Kriterium verwendet, um den Anschlag zu erkennen. Alternativ kann ein Vergleichsmoment angegeben werden, bei dem der Anschlag als erkannt gilt (siehe Abschnitt *Registerkarte: Momente* im jeweiligen Produkthandbuch). Der Anschlag muss mechanisch so dimensioniert sein, dass er bei dem parametrisierten Maximalstrom keinen Schaden nimmt. Die Nullposition bezieht sich direkt auf den Anschlag. Da in diesem Fall die Nullposition direkt auf dem Anschlag liegen würde, sollte der Parameter **Offset Startposition** verwendet werden, um die Nullposition geeignet zu verschieben.

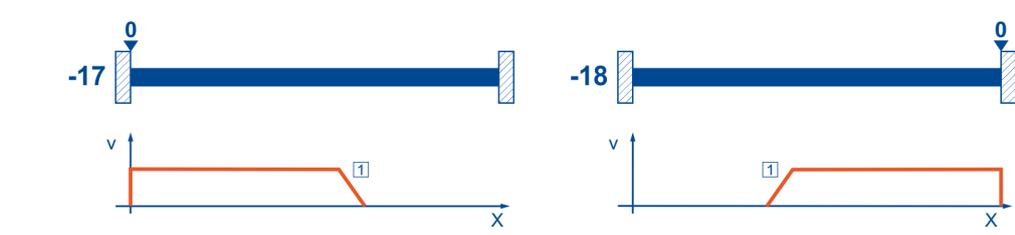


Abbildung 7: Referenzfahrt auf den Anschlag

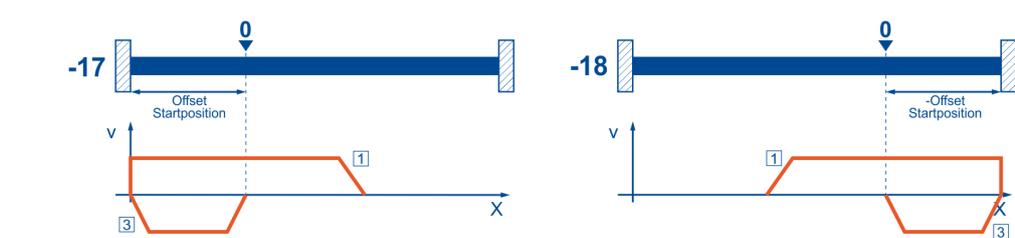


Abbildung 8: Verwendung von "Offset Startposition"

#### 3.4.10.2 Methoden -1 und -2: Anschlag mit Nullimpulsauswertung

Diese Methoden entsprechen den Methoden -17 und -18, die Nullposition bezieht sich allerdings zusätzlich auf den ersten Nullimpuls des Winkelgebers in negativer (-2) bzw. positiver (-1) Richtung vom Anschlag.

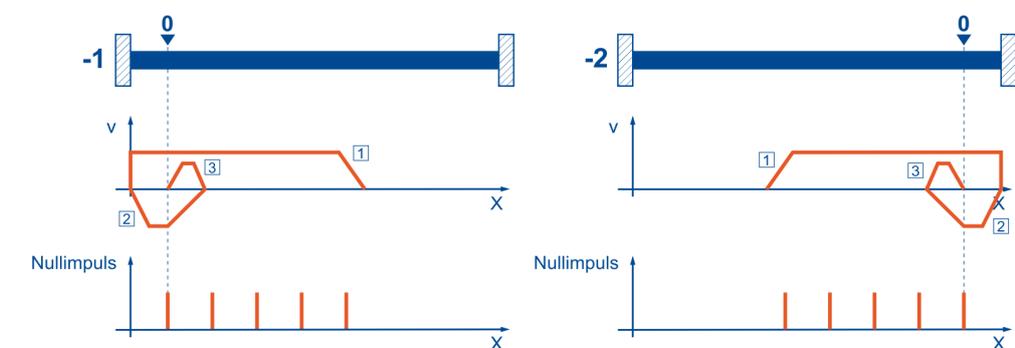


Abbildung 9: Referenzfahrt auf Anschlag mit Auswertung des Nullimpulses

### 3.4.10.3 Methoden 17 und 18: Positiver und negativer Endschalter

Bei diesen Methoden bewegt sich der Antrieb zunächst mit Suchgeschwindigkeit in positiver (18) bzw. negativer (17) Richtung, bis er den Endschalter erreicht. Danach fährt der Antrieb in Kriechgeschwindigkeit zurück und sucht die genaue Position des Endschalters. Die Nullposition bezieht sich auf die fallende Flanke vom Endschalter.

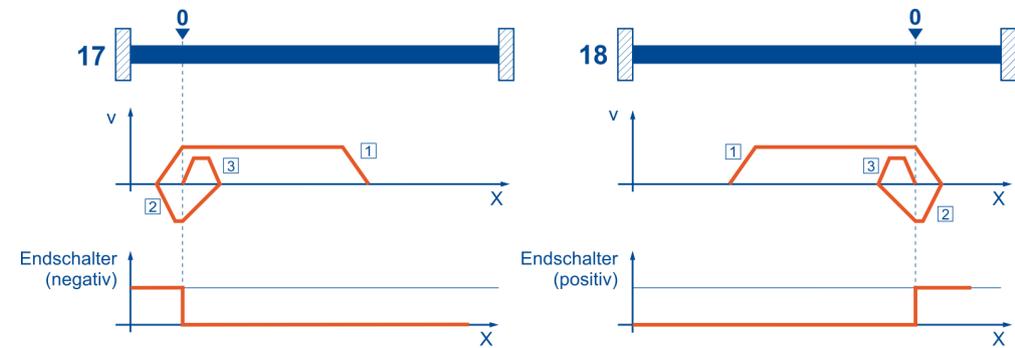


Abbildung 10: Referenzfahrt auf den Endschalter

### 3.4.10.4 Methoden 1 und 2: Positiver und negativer Endschalter mit Nullimpulsauswertung

Wie bei der vorhergehenden Methode wird zunächst auch der Endschalter gesucht. Zusätzlich bezieht sich die Nullposition allerdings auf den ersten Nullimpuls (Index pulse) des Winkelgebers in negativer (1) bzw. positiver (2) Richtung vom Endschalter.

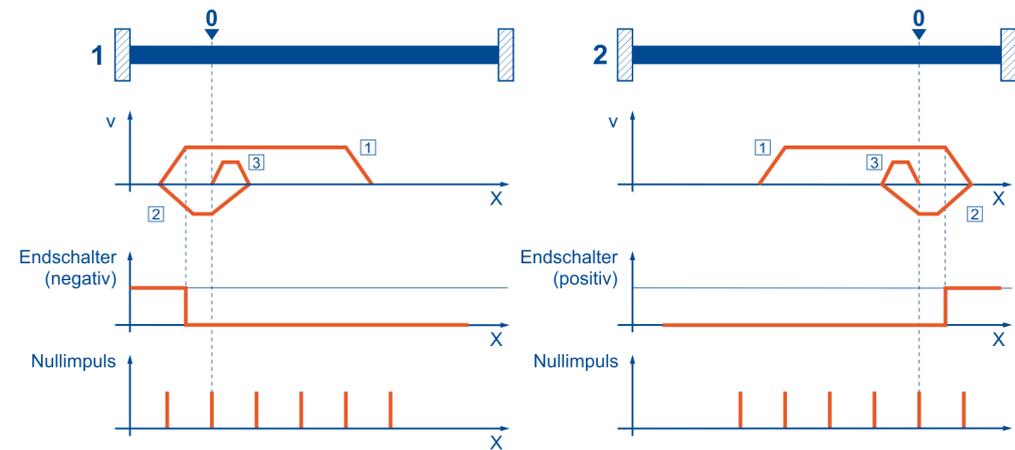


Abbildung 11: Referenzfahrt auf den Endschalter mit Auswertung Nullimpuls

### 3.4.10.5 Methoden 23 und 27: Referenzschalter

Diese beiden Methoden nutzen einen Referenzschalter, der nur über einen Teil der Strecke aktiv ist. Diese Referenzmethode bietet sich besonders für Rundachsen-Applikationen an, wo der Referenzschalter einmal pro Umdrehung aktiviert wird. Bei dieser Methode bewegt sich der Antrieb zunächst mit Suchgeschwindigkeit in positiver (23) bzw. negativer (27) Richtung, bis er den Referenzschalter erreicht. Danach fährt der Antrieb in Kriechgeschwindigkeit zurück und sucht die genaue Position des Referenzschalters. Die Nullposition bezieht sich auf die fallende Flanke vom Referenzschalter. Falls der Antrieb sich zunächst vom Referenzschalter weg bewegt, bewirkt der jeweilige Endschalter eine Drehrichtungsumkehr, so dass auch in diesem Fall der Referenzschalter gefunden wird.

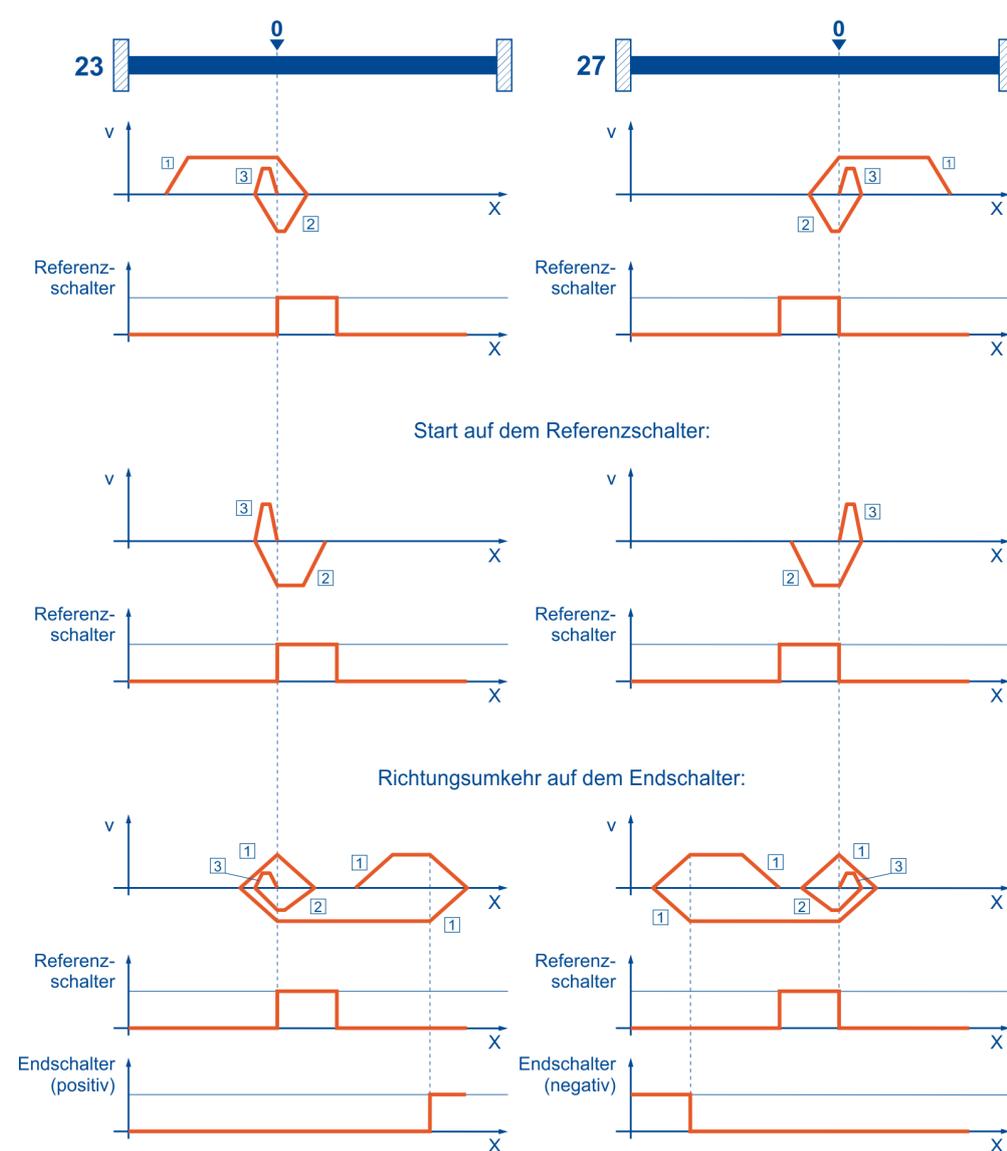


Abbildung 12: Referenzfahrt auf den Referenzschalter

### 3.4.10.6 Methoden 7 und 11: Referenzschalter und Nullimpulsauswertung

Die Methoden 7 und 11 benutzen wie die Methoden 23 und 27 den Referenzschalter, zusätzlich wird allerdings die Nullposition auf den ersten Nullimpuls in negativer oder positiver Richtung vom Referenzschalter bezogen.

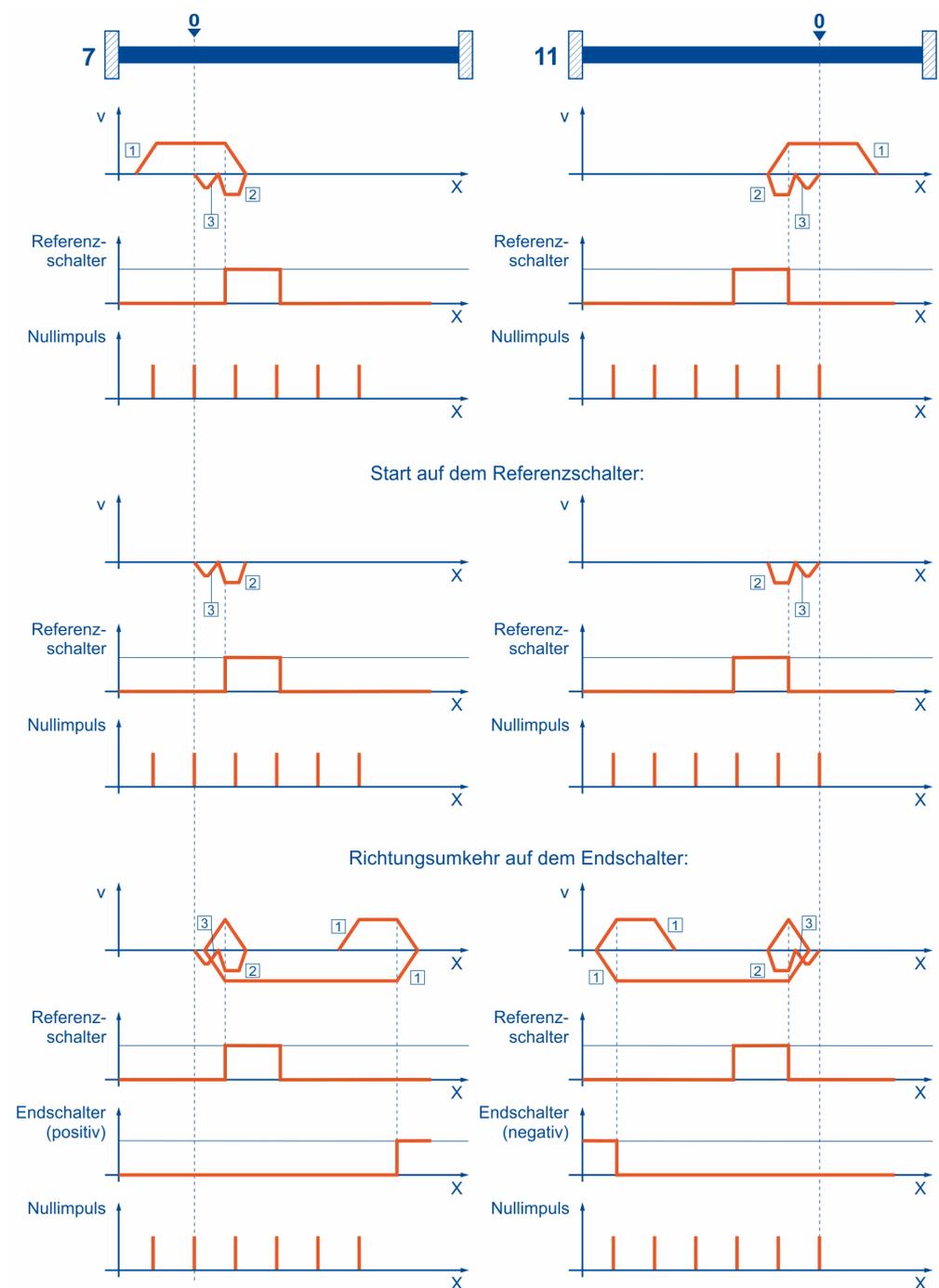


Abbildung 13: Referenzfahrt auf den Referenzschalter mit Nullimpulsauswertung

### 3.4.10.7 Methoden -23 und -27: Referenzfahrt (pos/neg) auf den Referenzschalter

Diese Methoden ähneln den Methoden 23 und 27. Allerdings wird hier zuerst das jeweilige Ende des Bewegungsbereiches gesucht, z.B. der Endanschlag oder ein Endschalter. Erst dann wird der Referenzschalter gesucht. Dadurch können an dem gleichen Eingang für den Referenzschalter mehrere Schalter angeschlossen sein. Während der Referenzfahrt wird dann der „letzte“ Schalter in Suchrichtung als Referenzschalter verwendet. Bei der Methode -23 bewegt sich der Antrieb zunächst in positiver und bei Methode -27 in negativer Richtung. Die Nullposition bezieht sich auf die fallende Flanke vom Referenzschalter.

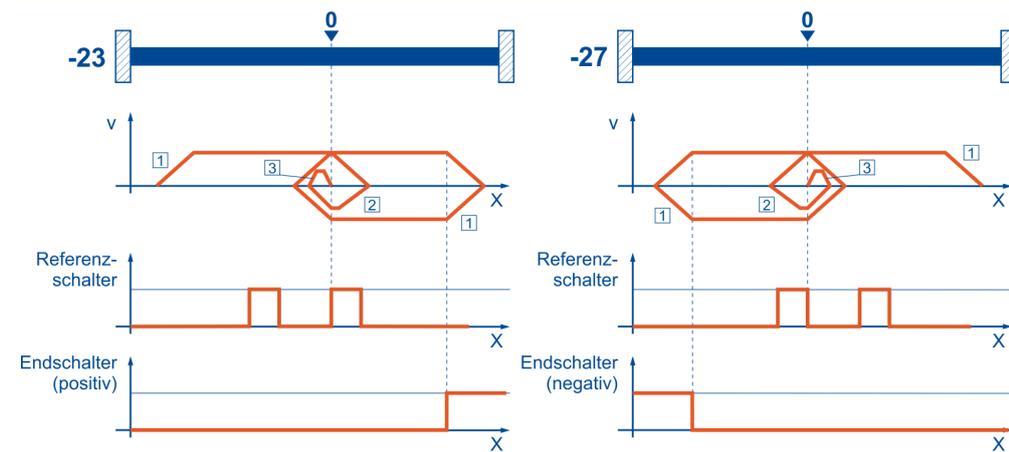


Abbildung 14: Referenzschalter bei positiver und negativer Anfangsbewegung

### 3.4.10.8 Methoden 32 und 33: Referenzfahrt auf den Nullimpuls

Bei den Methoden 32 und 33 ist die Richtung der Referenzfahrt negativ bzw. positiv. Die Nullposition bezieht sich auf den ersten Nullimpuls vom Winkelgeber in Suchrichtung.

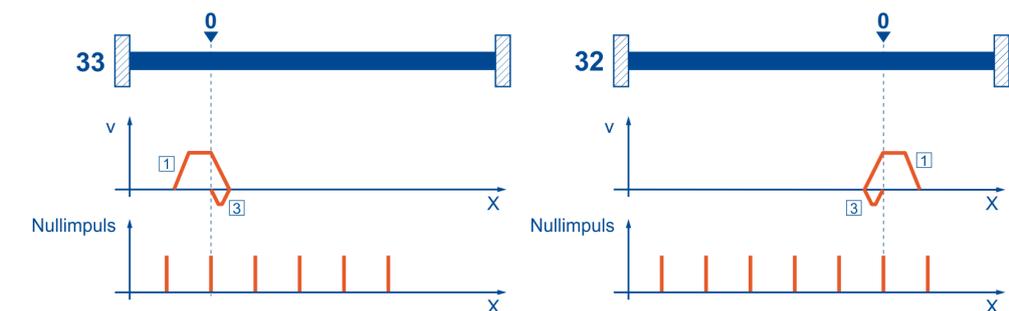


Abbildung 15: Nullimpuls bei negativer (32) und positiver (33) Anfangsbewegung

### 3.4.10.9 Methode 34: Referenzfahrt auf die aktuelle Position

Bei der Methode 34 wird die Nullposition auf die aktuelle Position bezogen, d.h., die aktuelle Position des Antriebs wird zu Null gesetzt.

### 3.4.11 PNU 1060: Thread Speed

Mit diesem Parameter kann die Einrichtdrehzahl verändert werden. Der Einrichtbetrieb kann durch einen entsprechenden digitalen Eingang aktiviert bzw. deaktiviert werden. Bei aktivem Einrichtbetrieb gelten sowohl in der Betriebsart Drehzahlregelung als auch Positionieren reduzierte Drehzahlgrenzwerte. Es wird die Einrichtdrehzahl direkt in Geschwindigkeitseinheiten geschrieben. Über das Parametrierprogramm Metronix ServoCommander® wird diese hingegen in prozentualer Abhängigkeit von der Grenzdrehzahl eingestellt. Beide verändern dieselbe interne Größe.

PNU	<b>1060</b>		
Name	<b>Thread Speed (Einrichtdrehzahl)</b>		
Info	Phys. Einheit Geschwindigkeit	rw	INT32
Value	--	3276 U/min	

### 3.4.12 PNU 1270: Position Control Parameters

Mit dieser Parameternummer können Einstellungen des Lagereglers vorgenommen werden. Der Totbereich beschreibt den Bereich der Regelabweichung, innerhalb dessen der Lageregler keine Stellgröße (Drehzahlsollwert) erzeugt. Dies kann z.B. bei Antrieben mit Getriebeispiel von Vorteil sein.

Index	<b>1270</b>		
Name	<b>Position Control Parameters</b>		
Type	RECORD		2
Sub-Index	<b>02</b>		
Name	<b>Position error tolerance window (Totbereich Lageregler)</b>		
Info	Physikalische Einheit Lage	rw	UINT32
Value	0,001 U...1 U	0,01°	

### 3.4.13 PNU 1271: Position Window Data

Mit dieser Parameternummer wird das Fenster für die „Ziel erreicht“-Meldung definiert.

Index	<b>1271</b>		
Name	<b>Position Window Data</b>		
Type	RECORD		0
Sub-Index	<b>00</b>		
Name	<b>Target Window (Fenster für "Ziel erreicht")</b>		
Info	Physikalische Einheit Lage	rw	UINT32
Value	--	10°	

### 3.4.14 PNU 1272: Following Error Data

Mit dieser Parameternummer können Einstellungen vorgenommen werden, die den Bereich für eine Schleppfehlermeldung betreffen. Hierüber kann ein Positionsfenster definiert werden, außerhalb dessen z.B. eine Warnung erzeugt wird (je nach Parametrierung der Reaktion).

Index	<b>1272</b>		
Name	<b>Following Error Data</b>		
Type	RECORD		0
Sub-Index	<b>00</b>		
Name	<b>Following Error Window (Schleppfehler-Fenster)</b>		
Info	Physikalische Einheit Lage	rw	UINT32
Value	0...101 U	50°	

### 3.4.15 PNU 1273: Position Error Data

Mit dieser Parameternummer können Parameter des Positionsfensters modifiziert werden, außerhalb dessen der Servoregler eine parametrierbare Reaktion ausführt. Neben dem Schleppfehler-Fenster kann hierüber ein zweites Positionsfenster definiert werden, das z.B. zur Abschaltung des Servoreglers mit Fehlermeldung führen kann (je nach Parametrierung der Reaktion).

Index	<b>1273</b>		
Name	<b>Position Error Data</b>		
Type	RECORD		0
Sub-Index	<b>00</b>		
Name	<b>Position Error Limit (Abschaltgrenze Schleppfehler)</b>		
Info	Physikalische Einheit Lage	rw	UINT32
Value	0...2 <sup>31</sup> -1	180°	

## 3.5 PNUs zur Betriebsart Drehzahlregelung

In diesem Abschnitt werden die Parameter beschrieben, die für die Betriebsart Drehzahlregelung benötigt werden.

### 3.5.1 PNU 1010: Target Velocity

Über diesen Parameter wird der Drehzahlsollwert eingestellt. Für diese Sollwerte ist der feste Sollwert 1 vorgesehen. In der Betriebsart Drehzahlregelung wird dieser Sollwert auch automatisch selektiert. Prinzipiell kann der Feldbus-Sollwert 1 als fester Sollwert auch im Parametersatz gespeichert werden. Wenn die PROFINET-Kommunikation im Parametersatz aktiv ist, wird der Feldbussollwert immer auf null gesetzt. Der im Parametersatz gespeicherte Wert wird dadurch immer überschrieben.

Der Wert 0 für das [Control word 1](#) führt ggf. dazu, dass der Feldbussollwert nicht auf die Rampe geführt wird (Sollwert nicht freigeschaltet). Die Einstellung des Sollwertselektors für Drehzahlregelung muss vorab bei inaktiver PROFINET-Kommunikation erfolgen. Weitere Informationen hierzu sind Abschnitt 5 *Betriebsarten* auf Seite 68 zu entnehmen.

PNU	<b>1010</b>		
Name	<b>Target Velocity</b>		
Info	Physik. Einheit Geschwindigkeit	rw	INT32
Value		0	

### 3.5.2 PNU 1011: Accelerations for Velocity Control

Unter dieser Parameternummer können die Beschleunigungswerte für die Betriebsart Drehzahlregelung parametrisiert werden. Der Servoregler definiert 4 unterschiedliche Beschleunigungsrampen. Da in den meisten Anwendungsfällen mehrere Rampen gleich parametrisiert werden, steht die folgende Auswahl zur Verfügung:

- Beschleunigung, kombiniert für positive und negative Drehrichtung
- Bremsbeschleunigung, kombiniert für positive und negative Drehrichtung
- Kombination für Beschleunigung und Bremsbeschleunigung für positive und negative Drehrichtung

Die Daten werden so interpretiert, wie sie als physikalische Einheit eingestellt sind. Dazu stellt die Parametriersoftware Metronix ServoCommander® ein entsprechendes Fenster zur Verfügung, siehe Abschnitt 3.1 *Physikalische Einheiten* auf Seite 24.

Die Parameter mit den Subindizes 0 und 1 erlauben den Zugriff auf die Beschleunigung für jeweils beide Drehrichtungen. Intern werden daher immer die Beschleunigungen für beide Drehrichtungen geschrieben. Beim Lesen wird hingegen nur der aktuelle Wert der Beschleunigungsrampe für positive Drehrichtung zurückgegeben. Der Anwender muss selbst sicherstellen, dass das Lesen eines Wertes ausreicht. Dies kann z.B. durch einmaliges Lesen und anschließendes Zurückschreiben dieses Wertes erreicht werden.

Index	<b>1011</b>		
Name	<b>Accelerations for Velocity Control</b>		
Type	RECORD		2
Sub-Index	<b>00</b>		
Name	<b>Acceleration Velocity Control (Beschleunigungsrampe Drehzahlregelung)</b>		
Info	Physikalische Einheit Beschleunigung	rw	UINT32
Value	--		14.100 (U/min)/s
Sub-Index	<b>01</b>		
Name	<b>Deceleration Velocity Control (Bremsrampe Drehzahlregelung)</b>		
Info	Physikalische Einheit Beschleunigung	rw	UINT32
Value	--		14.100 (U/min)/s

Der Parameter **All Accelerations Velocity Control** erlaubt das gleichzeitige Schreiben von Beschleunigungs- und Bremsrampe für beide Drehrichtungen, so dass nur ein Datenwert übertragen werden muss. Beim Lesen wird hingegen nur der aktuelle Wert der Beschleunigungsrampe für positive Drehrichtung zurückgegeben. Um sicherzustellen, dass alle vier Werte konsistent sind, kann der Parameter einmalig gelesen und anschließend zurückgeschrieben werden.

Sub-Index	<b>02<sub>h</sub></b>		
Name	<b>All Accelerations Velocity Control (Beschleunigung und Bremsbeschleunigung für Drehzahlregelung)</b>		
Info	Physikalische Einheit Beschleunigung	rw	UINT32
Value	--		14.100 (U/min)/s

### 3.5.3 CAN-Objekt 2415<sub>h</sub>: current\_limitation

Mit der Objektgruppe **current\_limitation** kann der Maximalstrom für den Motor begrenzt werden, wodurch z.B. ein drehmomentbegrenzter Drehzahlbetrieb ermöglicht wird. Über das Objekt **limit\_current** wird das Begrenzungsmoment in mA vorgegeben.

Index	<b>2415<sub>h</sub></b>		
Name	<b>current_limitation</b>		
Type	RECORD		02 <sub>h</sub>
Sub-Index	<b>02<sub>h</sub></b>		
Name	<b>limit_current</b>		
Info	mA	rw	PDO INT32
Value	--		--

## 3.6 PNUs zur Betriebsart Drehmomentregelung

In diesem Abschnitt werden die Parameter beschrieben, die für die Betriebsart Drehmomentregelung benötigt werden.

### 3.6.1 CAN-Objekt 6071<sub>h</sub>: target\_torque

Dieser Parameter ist im drehmomentengeregelten Betrieb der Eingabewert für den Drehmomentenregler. Er wird in Tausendstel des Nennstroms angegeben.

Index	6071 <sub>h</sub>			
Name	target_torque			
Info	% (1000 = motor_rated torque)	rw	PDO	INT16
Value	--	--		

### 3.6.2 CAN-Objekt 2416<sub>h</sub>: speed\_limitation

Mit der Objektgruppe `speed_limitation` kann in der Betriebsart `Drehmomentregelung` die Maximaldrehzahl des Motors begrenzt werden, wodurch ein drehzahlbegrenzter Drehmomentbetrieb ermöglicht wird. Über das Objekt `limit_speed` wird die Begrenzungsdrehzahl (Quelle = Fester Wert) in `speed_units` vorgegeben. Bei Verwendung dieses CAN-Objekts muss beachtet werden, dass die CANopen Factor Group für Geschwindigkeit passend zur physikalischen Einheit Geschwindigkeit (siehe Abschnitt 3.1 *Physikalische Einheiten* auf Seite 24) eingestellt werden muss.

Index	2416 <sub>h</sub>			
Name	speed_limitation			
Type	RECORD			02 <sub>h</sub>
Sub-Index	02 <sub>h</sub>			
Name	limit_speed			
Info	speed_unit	rw	<del>PDO</del>	INT32
Value	--	--		

## 3.7 PNUs für den Tippbetrieb (Jogging)

In diesem Abschnitt werden die Parameter beschrieben, mit denen der Tippbetrieb parametrierbar wird.

### 3.7.1 PNU 1040: Symmetrical Jogging

Unter dieser Parameternummer kann auf die beiden Geschwindigkeiten und alle vier Beschleunigungswerte für das Tippen in vereinfachter Form zugegriffen werden. Im zyklischen Datentelegramm müssen auf diese Weise weniger Daten übertragen werden. Intern werden aber jeweils alle Parameter geschrieben. Beim Auslesen wird der Wert der positiven Geschwindigkeit/Beschleunigung zurückgegeben. Der Parameter für die Tippgeschwindigkeiten hat die Eigenschaft, dass dieser unmittelbar wirksam wird. Damit wirkt sich eine Änderung auch aus, wenn Tippen bereits aktiv ist.

Index	<b>1040</b>		
Name	<b>Symmetrical Jogging</b>		
Type	RECORD		1
Sub-Index	<b>00</b>		
Name	<b>Symmetrical Jogging Velocity (Tippgeschwindigkeiten, symmetrisch)</b>		
Info	Physikalische Einheit Geschwindigkeit	rw	INT32
Value	--		100 U/min
Sub-Index	<b>01</b>		
Name	<b>Symmetrical Jogging Accelerations (Tippbeschleunigungen, symmetrisch)</b>		
Info	Physikalische Einheit Beschleunigung	rw	UINT32
Value			1.000 (U/min)/s

## 3.7.2 PNU 1041: Jogging Positive

Unter dieser Parameternummer werden die Parameter für das Tippen in positiver Richtung (TIPP0) detaillierter parametrisiert. Je nach Applikation kann auf diese Weise auch gezielt ein einzelner Parameter geändert werden. Der Parameter für die Tippgeschwindigkeit hat die Eigenschaften, dass dieser unmittelbar wirksam wird. Damit wirkt sich eine Änderung auch aus, wenn Tippen in positiver Richtung bereits aktiv ist. Die Beschleunigungen können für Anfahren und Bremsen (Subindex 1 und 2) separat oder zusammen (Subindex 3) eingestellt werden. Beim Lesen liefert Subindex 3 den Wert von Subindex 1 zurück.

Index	<b>1041</b>		
Name	<b>Jogging Positive</b>		
Type	RECORD		3
Sub-Index	<b>00</b>		
Name	<b>Jogging Velocity Positive (Tippgeschwindigkeit positiv)</b>		
Info	Physikalische Einheit Geschwindigkeit	rw	INT32
Value	--		100 U/min
Sub-Index	<b>01</b>		
Name	<b>Jogging Acceleration Positive (Beschleunigung für Tippen positiv)</b>		
Info	Physikalische Einheit Beschleunigung	rw	UINT32
Value			1.000 (U/min)/s
Sub-Index	<b>02</b>		
Name	<b>Jogging Deceleration Positive (Bremsbeschleunigung für Tippen positiv)</b>		
Info	Physikalische Einheit Beschleunigung	rw	UINT32
Value			1.000 (U/min)/s
Sub-Index	<b>03</b>		
Name	<b>Symmetrical Jogging Accelerations Positive (Beschleunigungen für Tippen in positiver Richtung, symmetrisch)</b>		
Info	Physikalische Einheit Beschleunigung	rw	UINT32
Value			1.000 (U/min)/s

### 3.7.3 PNU 1042: Jogging Negative

Unter dieser Parameternummer werden die Parameter für das Tippen in negativer Richtung (TIPP1) detaillierter parametrisiert. Je nach Applikation kann auf diese Weise auch gezielt ein einzelner Parameter geändert werden. Der Parameter für die Tippgeschwindigkeit hat die Eigenschaften, dass dieser unmittelbar wirksam wird. Damit wirkt sich eine Änderung auch aus, wenn Tippen in negativer Richtung bereits aktiv ist. Die Beschleunigungen können für Anfahren und Bremsen (Subindex 1 und 2) separat oder zusammen (Subindex 3) eingestellt werden. Beim Lesen liefert Subindex 3 den Wert von Subindex 1 zurück.

Index	<b>1042</b>		
Name	<b>Jogging Negative</b>		
Type	RECORD		3
Sub-Index	<b>00</b>		
Name	<b>Jogging Velocity Negative (Tippgeschwindigkeit negativ)</b>		
Info	Physikalische Einheit Geschwindigkeit	rw	INT32
Value	--		100 U/min
Sub-Index	<b>01</b>		
Name	<b>Jogging Acceleration Negative (Beschleunigung für Tippen negativ)</b>		
Info	Physikalische Einheit Beschleunigung	rw	UINT32
Value			1.000 (U/min)/s
Sub-Index	<b>02</b>		
Name	<b>Jogging Deceleration Negative (Bremsbeschleunigung für Tippen negativ)</b>		
Info	Physikalische Einheit Beschleunigung	rw	UINT32
Value			1.000 (U/min)/s
Sub-Index	<b>03</b>		
Name	<b>Symmetrical Jogging Accelerations Negative (Beschleunigungen für Tippen in negativer Richtung, symmetrisch)</b>		
Info	Physikalische Einheit Beschleunigung	rw	UINT32
Value			1.000 (U/min)/s

## 3.8 Weitere PNUs

In diesem Abschnitt sind PNUs aus unterschiedlichen Bereichen zusammengefasst.

### 3.8.1 PNU 964: Device Identification

Der Parameter Device Identification liefert Informationen zum angeschlossenen Gerät.

Index	<b>964</b>		
Name	<b>Device Identification</b>		
Type	RECORD		4
Sub-Index	<b>00</b>		
Name	<b>Manufacturer (Hersteller)</b>		
Info	--	ro	UINT16
Value	277 (115 <sub>h</sub> )		277 (115 <sub>h</sub> )
Sub-Index	<b>01</b>		
Name	<b>Device Type (Gerätetyp)</b>		
Info	--	ro	UINT16
Value	siehe Tabelle	--	

Wert	Bedeutung	Wert	Bedeutung
8202 <sub>h</sub>	BL 4102-C	820A <sub>h</sub>	BL 4104-M ETH
8203 <sub>h</sub>	BL 4104-C	820C <sub>h</sub>	BL 4104-D ETH
8204 <sub>h</sub>	BL 4108-C	820D <sub>h</sub>	BL 4840-M ETH
8208 <sub>h</sub>	BL 4304-C	820F <sub>h</sub>	BL 4840-D ETH
8209 <sub>h</sub>	BL 4308-C	820B <sub>h</sub>	BL 4104-M CAN
8212 <sub>h</sub>	BL 4312-C	8210 <sub>h</sub>	BL 4104-D CAN
8213 <sub>h</sub>	BL 4320-C	820E <sub>h</sub>	BL 4840-M CAN
8214 <sub>h</sub>	BL 4340-C	8211 <sub>h</sub>	BL 4840-D CAN
8215 <sub>h</sub>	BL 4360W-C		

Sub-Index	<b>02</b>		
Name	<b>Version</b>		
Info	MMSS <sub>h</sub>	ro	UINT16
Value	--	--	

Wert	Bedeutung
M	main version
S	sub version

Die PNUs mit den Subindizes 3 und 4 geben das Erstellungsdatum der PROFINET-Implementierung an. Dies kann auch in verschiedenen Produktstufen bzw. Revisionen gleich sein, wenn sich an der grundlegenden Implementierung nichts geändert hat

Sub-Index	<b>03</b>		
Name	<b>Erstellungsdatum: Jahr</b>		
Info	--	ro	UINT16
Value	--	--	

Sub-Index	<b>04</b>		
Name	<b>Erstellungsdatum: Tag/Monat</b>		
Info	DDMM <sub>h</sub>	ro	UINT16
Value	--	--	

Wert	Bedeutung
D	Tag
M	Monat

### 3.8.2 PNU 971: Transfer into a non-volatile memory

Über diese Parameternummer können verschiedene Speichervorgänge von Parametern im Servoregler aktiviert werden.

PNU	<b>971</b>		
Name	<b>Transfer into a non-volatile memory</b>		
Info		ro	UINT16
Value	0, 1, 256, 257, 258	0	

Wert	Bedeutung
0	Keine Aktion
1	Speichern des aktuellen Parametersatzes und aller Positionsdatensätze
256	Speichern aller Positionsdatensätze ohne Standardparameter
257	Speichern im Winkelgeber: Nur Standard Parameter
258	Speichern im Winkelgeber: Standard Parameter + Absolutposition

Beim Speichern des Standard-Parametersatzes (Schreiben einer 1), kann durch Lesen dieser PNU ermittelt werden, ob der Speichervorgang abgeschlossen ist. Läuft der Speichervorgang noch wird eine 1 zurückgegeben, anderenfalls eine 0.

### 3.8.3 PNU 2000: PKW Access

Diese Parameternummer muss in einem Telegramm eingetragen werden, um während der Laufzeit einen asynchronen Zugriff auf einen beliebigen Antriebs-Parameter durchführen zu können. In den Empfangs- und Antworttelegrammen 1..3 darf diese PNU nur einmalig eingetragen werden. Dies muss vom Anwender entsprechend sichergestellt werden.

Diese PNU muss sowohl im Empfangs- als auch im Antworttelegrammen eingetragen werden. Um beispielsweise einen Lesezugriff durchzuführen, schreibt die Steuerung die PNU-Nummer in Byte 1..2 und 41<sub>h</sub> in Byte 0. Der Regler antwortet mit 41<sub>h</sub> in Byte 0, der PNU in Byte 1..2 und dem gelesenen Wert in Byte 4..7 des Antworttelegramms. Im Falle eines Schreibzugriffs muss die Steuerung die Bytes 4..7 im Empfangstelegramm entsprechend setzen.

Um aufeinanderfolgende Zugriffe gleichen Typs unterscheiden zu können, muss zwischenzeitlich Byte 0 mit 0 beschrieben werden. Erst wenn der Regler diesen Wert spiegelt, darf ein neuer Zugriff erfolgen.

PNU	<b>2000</b>		
Name	<b>PKW Access</b>		
Info	--	rw	2 * UINT32
Value	--	0	

Byte	Bedeutung
0	Zugriffsart: 00 <sub>h</sub> : Kein Zugriff 41 <sub>h</sub> : Lesezugriff 42 <sub>h</sub> : Schreibzugriff Alle übrigen Werte sind reserviert
1..2	Parameternummer (PNU) oder CAN-Index Bei der Verwendung von CAN-Objekten bitte beachten, dass Index und Subindex hexadezimal angegeben sind und die Objekte mit der CANopen Factor-Group skaliert sind.
3	Subindex PNU oder CAN-Subindex
4..7	Daten

### 3.8.4 PNU 2010: Placeholder

Dieser Parameter wird für die Parametrierung der zyklischen Telegramme benötigt, siehe hierzu auch Abschnitt 6.1 *Telegrammeditor* auf Seite 71.

Er erlaubt das Auffüllen von Telegrammen. Auf diese Weise können Datenbereiche (z.B. Datenbausteine) so angelegt werden, dass Parameter von einer Länge mit 2 Byte oder 4 Byte auf geraden Speicheradressen liegen.

Index	<b>2010</b>		
Name	<b>Placeholder</b>		
Type	RECORD		2
Sub-Index	<b>00</b>		
Name	<b>8 Bit</b>		
Info	--	rw	UINT8
Value	--	0	
Sub-Index	<b>01</b>		
Name	<b>16 Bit</b>		
Info	--	rw	UINT16
Value	--	0	
Sub-Index	<b>02</b>		
Name	<b>32 Bit</b>		
Info	--	rw	UINT32
Value	--	0	

### 3.8.5 PNU 2011: Element 0

Dieser Parameter wird für die Parametrierung der zyklischen Telegramme benötigt, siehe hierzu auch Abschnitt 6.1 *Telegrammeditor* auf Seite 71.

Er verhält sich identisch zum Parameter mit der PNU 2010.2. Der Unterschied besteht darin, dass dieser im Telegrammeditor des Metronix ServoCommander® nicht dargestellt wird, wenn dieser am Ende eines Telegramms eingetragen wird. Die Anzahl der Einträge in einem Telegramm ist immer auf 10 festgelegt. Nicht benötigte Einträge erhalten daher diese PNU.

Index	<b>2011</b>		
Name	<b>Element 0</b>		
Type	RECORD		0
Sub-Index	<b>00</b>		
Name	<b>32 Bit</b>		
Info	--	rw	UINT32
Value	--	0	

## 4 Gerätesteuerung

Zur Ansteuerung des Reglers sind zwei Datenworte spezifiziert. Über das **Control word 1** werden die wesentlichen Gerätefunktionen durch den Master gesteuert, während der Status des Gerätes im **Status word 1** zurückgelesen wird. Die Gerätesteuerung erfolgt in Anlehnung an die PROFIdrive-Spezifikation, einige Funktionen sind allerdings herstellerspezifisch implementiert.

Im Folgenden werden zunächst **Control word 1** und **Status word 1** beschrieben. Danach wird die Gerätesteuerung unter Verwendung dieser beiden Datenworte erläutert.

### 4.1 Verwendete Begriffe

Zur Beschreibung der Gerätesteuerung werden die folgenden Begriffe verwendet:

Begriff	Erklärung
Zustand: (State)	Je nachdem ob beispielsweise die Endstufe eingeschaltet oder ein Fehler aufgetreten ist, befindet sich der Servoregler in verschiedenen Zuständen. Die unter PROFINET definierten Zustände werden im Laufe des Kapitels vorgestellt. Beispiel: SWITCHING_ON_INHIBITED
Zustandsübergang (State Transition)	Ebenso wie die Zustände ist es ebenfalls definiert, wie man von einem Zustand zu einem anderen gelangt (z.B. um einen Fehler zu quittieren). Zustandsübergänge werden vom Master durch Setzen von Bits im <b>Control word 1</b> ausgelöst oder intern durch den Servoregler, wenn dieser beispielsweise einen Fehler erkennt.
Kommando (Command)	Zum Auslösen von Zustandsübergängen müssen bestimmte Kombinationen von Bits im <b>Control word 1</b> gesetzt werden. Eine solche Kombination wird als Kommando bezeichnet. Beispiel: <code>Enable Operation</code>
Zustandsdiagramm (State Machine)	Die Zustände und Zustandsübergänge bilden zusammen das Zustandsdiagramm, also die Übersicht über alle Zustände und die von dort möglichen Übergänge.

## 4.2 PNU 967: Control word 1

Mit dem **Control word 1** werden verschiedene Gerätefunktionen gesteuert, z.B. die Reglerfreigabe über die Bits 0 bis 3. Die Anwendung dieser Bits wird in Abschnitt 4.4 *Das Zustandsdiagramm (State Machine)* auf Seite 62 beschrieben. Anderen Bits ist direkt eine bestimmte Aktion zugeordnet, wie z.B. der Start einer Positionierung. Diese Bits können auch in Abhängigkeit von der Betriebsart unterschiedliche Bedeutungen haben.

PNU	<b>967</b>		
Name	<b>Control word 1</b>		
Info	--	rw	UINT16
Value	--	0	

In den Empfangstelegrammen 0..2 ist das Control word 1 an einer festen Position enthalten. Es wird jeweils als letztes Datum ausgewertet. Dadurch wird z.B. eine neue Zielposition zuerst geschrieben. Ein gleichzeitig übertragenes Kommando zum Starten einer Positionierung bezieht sich daher immer auf die Daten, die im gleichen Telegramm übertragen worden sind.

Bit	Wert	Funktion
0	0001 <sub>h</sub>	Steuerung der Zustandsübergänge. (Diese Bits werden gemeinsam ausgewertet)
1	0002 <sub>h</sub>	
2	0004 <sub>h</sub>	
3	0008 <sub>h</sub>	
4	0010 <sub>h</sub>	Enable ramp Generator / Do not reject traversing task
5	0020 <sub>h</sub>	Unfreeze ramp Generator / No intermediate stop
6	0040 <sub>h</sub>	Enable setpoint / Activate traversing task
7	0080 <sub>h</sub>	Fault reset
8	0100 <sub>h</sub>	Jog 1 on
9	0200 <sub>h</sub>	Jog 2 on
10	0400 <sub>h</sub>	Control by PLC
11	0800 <sub>h</sub>	Start homing
12	1000 <sub>h</sub>	Relative
13	2000 <sub>h</sub>	Change immediately
14	4000 <sub>h</sub>	Reserviert, mit 0 beschreiben
15	8000 <sub>h</sub>	Reserviert, mit 0 beschreiben

<b>Bit 4</b>	<b>Abhängig von der Betriebsart:</b>
Do not reject traversing task	Im <b>Positioniermodus</b> : 0: Eine laufende Positionierung wird abgebrochen 1: Keine Auswirkung
Enable ramp Generator	In <b>Drehzahlregelung</b> : 0: Alle Drehzahlsollwerte gesperrt ┌┐: Beschleunigung auf Sollgeschwindigkeit 1: Alle Drehzahlsollwerte freigegeben └┘: Bremsen auf Drehzahl 0
<b>Bit 5</b>	<b>Abhängig von der Betriebsart:</b>
No intermediate stop	Im <b>Positioniermodus</b> : 0: Positionierung angehalten ┌┐: Beschleunigung (gemäß aktuellem Positionssatz) auf Fahrgeschwindigkeit 1: Positionierung zulässig/läuft └┘: Unterbricht die aktuelle Positionierung und bremst auf 0 (mit Bremsbeschleunigung aus aktuellem Positionssatz)
Unfreeze ramp Generator	In <b>Drehzahlregelung</b> : 0: Sollwertrampe aktiv 1: Sollwertrampe eingefroren Der gerade aktive Sollwert wird "eingefroren" und bleibt somit anstehen. Bit 4 (Enable ramp Generator) hat in diesem Fall keine Auswirkung.
<b>Bit 6</b>	<b>Abhängig von der Betriebsart:</b>
Activate traversing task	Im <b>Positioniermodus</b> : 0: Positionierung angehalten ┌┐: Start der Positionierung, wenn <ul style="list-style-type: none"> <li>• Bit 4 = 1 (Fahrauftrag nicht abbrechen)</li> <li>• Bit 5 = 1 (Kein Zwischenstop)</li> <li>• Keine Referenzfahrt aktiv</li> </ul> 1: Positionierung zulässig/läuft └┘: Positionierung wird unterbrochen. Es wird mit der Bremsbeschleunigung auf 0 heruntergerammt. Welcher Positionssatz verwendet wird, wird über die PNU 1002.0 festgelegt. Wird der PROFINET-Positionssatz ausgewählt, werden die Optionen (z.B. relative) aus dem <b>Control word 1</b> verwendet, ansonsten die aus dem jeweiligen Satz.
<b>Bit 6</b>	<b>Abhängig von der Betriebsart:</b>
Enable setpoint	In <b>Drehzahlregelung</b> : 0: Drehzahlsollwert gesperrt 1: Drehzahlsollwert freigegeben Bei Verwendung dieses Bits werden die Sollwerte unmittelbar auf 0 gesetzt (Bit wird gelöscht) oder freigegeben (Bit wird gesetzt), ohne den Sollwert ab- oder aufzurampen.

Bit 7	
Fault reset	<p>┘: Quittierung von anstehenden Fehlern, falls deren Ursache behoben ist</p>
Bit 8	
Jog 1 on	<p>Es wird eine Bewegung in positiver Richtung ausgeführt, die durch den Positionssatz TIPP 0 (pos) bestimmt ist:</p> <p>0: Bewegung gestoppt</p> <p>┘: Start in positive Richtung</p> <p>1: Bewegung aktiv</p> <p>┘: Bremsen mit Bremsbeschleunigung</p>
Bit 9	
Jog 2 on	<p>Es wird eine Bewegung in negativer Richtung ausgeführt, die durch den Positionssatz TIPP 1 (neg) bestimmt ist:</p> <p>0: Bewegung gestoppt</p> <p>┘: Start in negative Richtung</p> <p>1: Bewegung aktiv</p> <p>┘: Bremsen mit Bremsbeschleunigung</p>
Bit 10	
Control by PLC	<p>0: Control word 1 wird NICHT ausgewertet.</p> <p>1: Control word 1 wird ausgewertet.</p>
Bit 11	
Start homing	<p>┘: Start Referenzfahrt</p> <p>┘: Abbruch der Referenzfahrt ohne Fehler</p> <p>Bei der Referenzfahrt werden die im Metronix ServoCommander® eingestellten Optionen berücksichtigt.</p>
Bit 12	
Relative	<p>Nur im <b>Positioniermodus</b>:</p> <p>Bei Start einer Positionierung:</p> <p>0: Absolute Positionierung</p> <p>1: Relative Positionierung</p>
Bit 13	
Change immediately	<p>Nur im <b>Positioniermodus</b>:</p> <p>Bei Start einer Positionierung:</p> <p>0: Neue Positionierung an laufende anhängen</p> <p>1: Aktuelle Positionierung unterbrechen und neue sofort starten</p>

## 4.3 PNU 968: Status word 1

Mit dem Status word 1 werden verschiedene Gerätezustände wiedergespiegelt, z.B. eine aktive Reglerfreigabe. Einzelne Bits haben dafür entsprechende Bedeutung. Dies wird in Abschnitt 4.4 *Das Zustandsdiagramm (State Machine)* auf Seite 62 beschrieben.

PNU	<b>968</b>		
Name	<b>Status word 1</b>		
Info	--	rw	UINT16
Value	--	0	

Bit	Wert	Name
0	0001 <sub>h</sub>	Zustand des Servoreglers, siehe Abschnitt 4.4.1 <i>Zustandsdiagramm: Zustände</i> auf Seite 63. Diese Bits müssen gemeinsam ausgewertet werden
1	0002 <sub>h</sub>	
2	0004 <sub>h</sub>	
3	0008 <sub>h</sub>	Fault present
4	0010 <sub>h</sub>	Coast Stop Not Activated (No OFF2)
5	0020 <sub>h</sub>	Quick Stop Not Activated (No OFF3)
6	0040 <sub>h</sub>	Zustand des Servoreglers, siehe Bits 0...2
7	0080 <sub>h</sub>	Warning Present
8	0100 <sub>h</sub>	Following Error Within Tolerance Range / Speed error Within Tolerance Range
9	0200 <sub>h</sub>	Control Requested
10	0400 <sub>h</sub>	Target Position Reached / f Or n Reached Or Exceeded
11	0800 <sub>h</sub>	Home Position Set
12	1000 <sub>h</sub>	Traversing Task Acknowledgment
13	2000 <sub>h</sub>	Drive Stopped
14	4000 <sub>h</sub>	Reserved
15	8000 <sub>h</sub>	Reserved

<b>Bit 3</b>	
Fault Present	Es ist ein Fehler vorhanden
<b>Bit 4</b>	
Coast Stop Not Activated (No OFF2)	Kein <a href="#">Coast Stop</a> aktiv. Siehe Abschnitt 4.4 <i>Das Zustandsdiagramm (State Machine)</i> auf Seite 62
<b>Bit 5</b>	
Quick Stop Not Activated (No OFF3)	Kein <a href="#">Quick Stop</a> aktiv. Siehe Abschnitt 4.4 <i>Das Zustandsdiagramm (State Machine)</i> auf Seite 62
<b>Bit 7</b>	
Warning Present	Es ist eine Warnung oder eine Sollwertsperrung durch einen Endschalter aktiv
<b>Bit 8</b>	
Following Error Within Tolerance Range	<b>Abhängig von der Betriebsart:</b> Im <a href="#">Positioniermodus</a> : Kein Schleppfehler
Speed error Within Tolerance Range	In <a href="#">Drehzahlregelung</a> : Die Istdrehzahl liegt innerhalb des parametrierbaren Meldefensters um die Soll-drehzahl. Das Meldefenster wird im Metronix ServoCommander® unter <a href="#">Parameter   Meldungen, Tab Drehzahlmeldung (Drehzahlmeldung)</a> parametrierbar.
<b>Bit 9</b>	
Control by PLC	Kontrolle durch die Steuerung. Spiegelung von Bit 10 aus dem <a href="#">Control word 1</a>
<b>Bit 10</b>	
Target Position Reached	Im <a href="#">Positioniermodus</a> : Die laufende Positionierung ist beendet und die Istposition liegt innerhalb des Zielfensters
Frequency Reached Or Exceeded	In <a href="#">Drehzahlregelung</a> : Die Istdrehzahl ist gleich oder größer als die freie Vergleichsdrehzahl. Der Vergleich erfolgt unter Berücksichtigung des Vorzeichens! Die freie Vergleichsdrehzahl wird im Metronix ServoCommander® unter <a href="#">Parameter   Meldungen, Tab Drehzahlmeldung (Schwelle)</a> parametrierbar.
<b>Bit 11</b>	
Home Position Set	Die Referenzposition ist gültig (Durch Referenzfahrt oder aufgrund des angeschlossenen Gebersystems)
<b>Bit 12</b>	
Traversing Task Acknowledgment	Das Bit ist gelöscht, wenn der Regler einen neuen Positionierauftrag verarbeiten kann.
<b>Bit 13</b>	
Drive Stopped	Drehzahl = 0 und keine Positionierung aktiv bzw. Zwischenstopp aktiv.

## 4.4 Das Zustandsdiagramm (State Machine)

Die Zustände sind aus der PROFIdrive-Spezifikation weitgehend übernommen. PROFIdrive unterscheidet zwischen *Ramp stop* und *Quick stop*. Der Servoregler schaltet hier einheitlich die Reglerfreigabe aus, so dass sich das vereinfachte Zustandsdiagramm gemäß *Abbildung 16: Zustandsdiagramm des Servoreglers* ergibt.

### **▲ VORSICHT** Verletzungsgefahr durch falsch parametrieren Servoregler

Ein falsch parametrierter Servoregler kann unkontrollierte Drehbewegungen und dadurch Personenschäden oder Sachschäden verursachen.

Stellen Sie vor dem allerersten Einschalten der Endstufe sicher, dass der Servoregler die von Ihnen gewünschten Parameter enthält.

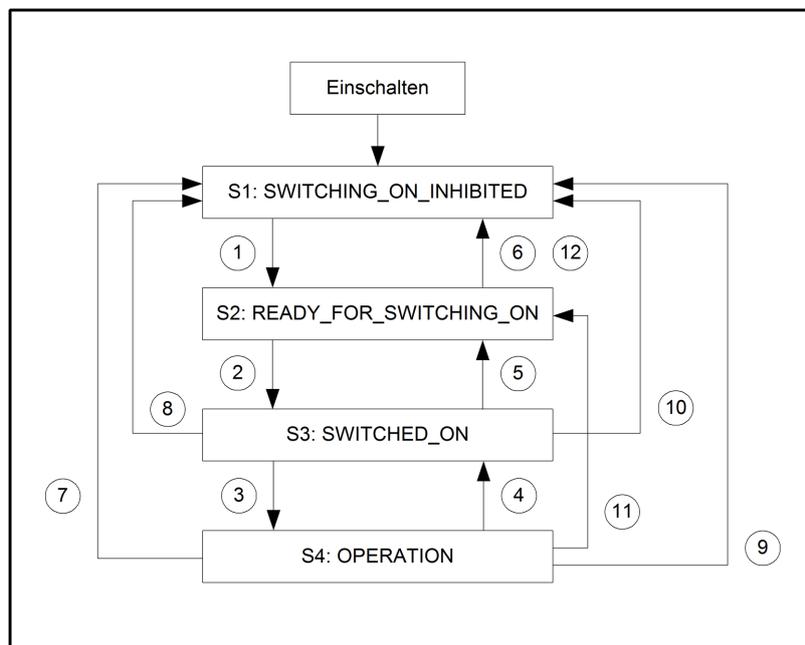


Abbildung 16: Zustandsdiagramm des Servoreglers

Nach dem Einschalten initialisiert sich der Servoregler und erreicht schließlich den Zustand `SWITCHING_ON_INHIBITED`. Die Endstufe ist deaktiviert und die Motorwelle ist frei drehbar. Durch die Zustandsübergänge 1, 2 und 3 gelangt man in den Zustand `OPERATION`. Dies entspricht dem Erteilen der Reglerfreigabe. In diesem Zustand ist die Endstufe eingeschaltet und der Antrieb wird gemäß der eingestellten Betriebsart geregelt. Stellen Sie daher vorher unbedingt sicher, dass der Servoregler richtig parametrier ist und ein entsprechender Sollwert gleich Null ist.

Der Zustandsübergang 4 entspricht der Wegnahme der Reglerfreigabe, d.h. ein noch laufender Motor wird gemäß eingestellter Nothaltrampe kontrolliert in den Stillstand abgebremst. Der Zustandsübergang 7 entspricht einem sofortigen Ausschalten der Endstufe, d.h. ein noch laufender Motor würde unregelt austrudeln.

Tritt ein Fehler auf, so wird (egal aus welchem Zustand) letztlich in den Zustand `SWITCHING_ON_INHIBITED` verzweigt. Je nach Schwere des Fehlers werden vorher noch bestimmte Aktionen, wie z.B. eine Notbremsung ausgeführt werden.

## 4.4.1 Zustandsdiagramm: Zustände

In der folgenden Tabelle sind alle Zustände und deren Bedeutung aufgeführt:

Name	Bedeutung
EINSCHALTEN	Der Servoregler führt einen Selbsttest durch. Die PROFINET-Kommunikation arbeitet noch nicht.
SWITCHING_ON_INHIBITED	Der Servoregler hat seinen Selbsttest abgeschlossen. PROFINET-Kommunikation ist möglich.
READY_FOR_SWITCHING_ON	Der Servoregler wartet, bis der digitale Eingang „Reglerfreigabe“ an 24 V liegt. (Wenn Reglerfreigabelogik = „Digitaler Eingang und PROFINET“)
SWITCHED_ON	Die Endstufe ist eingeschaltet.
OPERATION	Der Motor liegt an Spannung und wird entsprechend der Betriebsart geregelt.

### ➤ Auslesen des Servoreglerzustands

Ähnlich wie über die Kombination mehrerer Bits des **Control word 1** verschiedene Zustandsübergänge ausgelöst werden können, kann über die Kombination verschiedener Bits des **Status word 1** ausgelesen werden, in welchem Zustand sich der Servoregler befindet. Die folgende Tabelle listet die möglichen Zustände des Zustandsdiagramms sowie die zugehörige Bitkombination auf, mit der sie im **Status word 1** angezeigt werden.

Zustand	Bit 6	Bit 2	Bit 1	Bit 0	Maske	Wert
	0040 <sub>h</sub>	0004 <sub>h</sub>	0002 <sub>h</sub>	0001 <sub>h</sub>		
SWITCHING_ON_INHIBITED	0	0	0	0	0047 <sub>h</sub>	0040 <sub>h</sub>
READY_FOR_SWITCHING_ON	1	0	0	0	0047 <sub>h</sub>	0001 <sub>h</sub>
SWITCHED_ON	0	0	0	1	0047 <sub>h</sub>	0003 <sub>h</sub>
OPERATION	0	0	1	1	0047 <sub>h</sub>	0007 <sub>h</sub>

### BEISPIEL

Dieses Beispiel zeigt, wie der aktuelle Status des Reglers aus dem **Status word 1** ermittelt wird. Zur eindeutigen Bestimmung des Reglerzustands müssen auch nicht gesetzte Bits im **Status word 1** abgefragt werden. Daher muss das **Status word 1** entsprechend maskiert werden.

Zustand	
SWITCHING_ON_INHIBITED	(Status word 1 & 0047 <sub>h</sub> ) = 0040 <sub>h</sub>
READY_FOR_SWITCHING_ON	(Status word 1 & 0047 <sub>h</sub> ) = 0001 <sub>h</sub>
SWITCHED_ON	(Status word 1 & 0047 <sub>h</sub> ) = 0003 <sub>h</sub>
OPERATION	(Status word 1 & 0047 <sub>h</sub> ) = 0007 <sub>h</sub>

## 4.4.2 Zustandsdiagramm: Zustandsübergänge

**⚠ GEFAHR ⚠ Lebensgefahr durch elektrischen Schlag!**

Endstufe gesperrt bedeutet, dass die Leistungshalbleiter nicht mehr angesteuert werden. Wenn dieser Zustand bei einem drehenden Motor eingenommen wird, so trudelt dieser ungebremst aus. Eine eventuell vorhandene mechanische Motorbremse wird hierbei automatisch angezogen.

Das Signal garantiert nicht, dass der Motor wirklich spannungsfrei ist

**⚠ VORSICHT ⚠ Verletzungsgefahr**

Endstufe freigegeben bedeutet, dass der Motor entsprechend der gewählten Betriebsart angesteuert und geregelt wird. Eine eventuell vorhandene mechanische Motorbremse wird automatisch gelöst.

Bei einem Defekt oder einer Fehlparametrierung (Motorstrom, Polzahl, Resolver-Offsetwinkel etc.) kann es zu einem unkontrollierten Verhalten des Antriebes kommen.

In der folgenden Tabelle ist für alle Zustandsübergänge (siehe Abschnitt 4.4 *Das Zustandsdiagramm (State Machine)* auf Seite 62) das notwendige Kommando und die notwendige Bedingung aufgelistet. Die Spalte "Bits 0...3" gibt direkt die Bitkombination an, die im [Control word 1](#) angegeben werden muss. Ein x drückt dabei aus, dass dieses Bit nicht relevant ist.

Nr.	Kommando	Bits 0...3	Bedingung	Aktion
1	OFF	x110	Endstufen- und Reglerfreigabe vorhanden + kein Quickstop + kein Coast stop	-
2	On	x111		Einschalten der Endstufenfreigabe
3	Enable Operation	1111		Regelung gemäß eingestellter Betriebsart
4	Disable Operation	0111		Wegnahme der Reglerfreigabe
5	OFF	x110		Endstufe wird gesperrt. Motor ist frei drehbar
6	Coast Stop	xx0x		-
7	Coast Stop	xx0x		Endstufe wird gesperrt. Motor trudelt aus und ist frei drehbar
8	Coast Stop	xx0x		Endstufe wird gesperrt. Motor trudelt aus und ist frei drehbar
9	Quick Stop	x01x		Wegnahme der Reglerfreigabe
10	Quick Stop	x01x		Wegnahme der Reglerfreigabe

Nr.	Kommando	Bits 0...3	Bedingung	Aktion
11	OFF	x110		Wegnahme der Reglerfreigabe
12	Quick Stop	x01x		Wegnahme der Reglerfreigabe

Nach Schreiben eines Kommandos (durch Setzen von Bits im **Control word 1**), muss abgewartet werden, bis der angeforderte Zustand im **Status word 1** zurückgelesen werden kann.

## BEISPIEL

Der Servoregler soll freigegeben werden.  
Er befindet sich zunächst im Zustand SWITCHING\_ON\_INHIBITED.

Über-gang	Zustand	Aktionen
1	SWITCHING_ON_INHIBITED → READY_FOR_SWITCHING_ON	(Control word 1 = 0406 <sub>h</sub> )  Warten bis (Status word 1 & 0047 <sub>h</sub> ) = 0001 <sub>h</sub>
2	READY_FOR_SWITCHING_ON → SWITCHED_ON	(Control word 1 = 0407 <sub>h</sub> )  Warten bis (Status word 1 & 0047 <sub>h</sub> ) = 0003 <sub>h</sub>
3	SWITCHED_ON → OPERATION	(Control word 1 = 040F <sub>h</sub> )  Warten bis (Status word 1 & 0047 <sub>h</sub> ) = 0007 <sub>h</sub>

Das Beispiel geht davon aus, dass keine weiteren Bits im **Control word 1** gesetzt sind. Bit 10 muss gesetzt sein, um eine Steuerung durch den Master zu ermöglichen.

## 4.5 Anzeige des aktuellen Steuer-/Statusworts (Control word 1/Status word 1)

Zur Vereinfachung der Inbetriebnahme können das Steuerwort (**Control word 1**) und das Statuswort (**Status word 1**) im Metronix ServoCommander® angezeigt werden. Das Fenster lässt sich über **Parameter | Feldbus | PROFINET | Diagnose | Steuer-/Statuswort...** aufrufen.

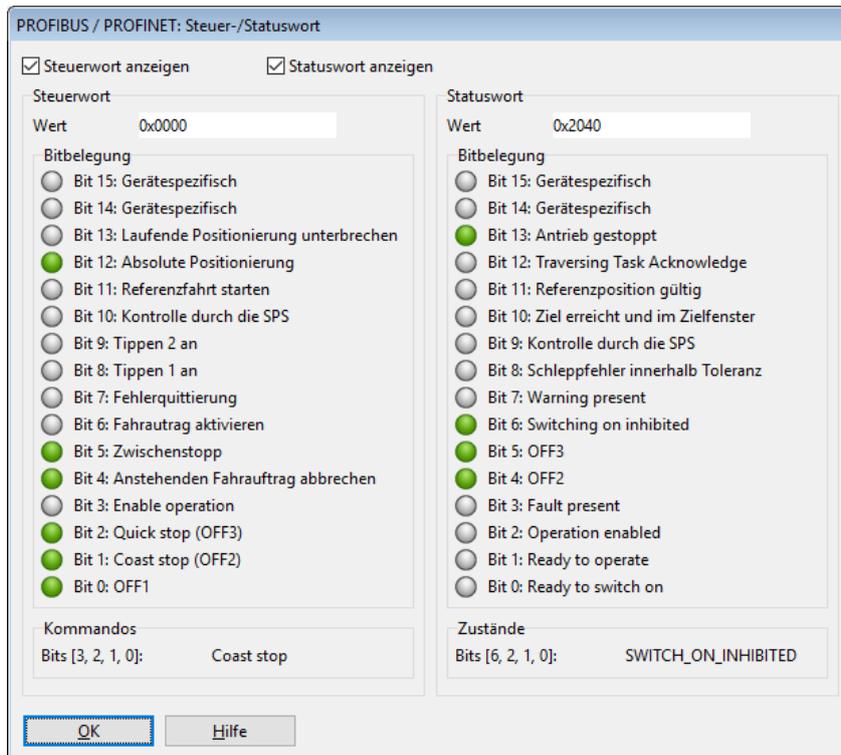


Abbildung 17: Fenster "PROFIBUS/PROFINET: Steuer-/Statuswort"

Über das jeweilige Kontrollkästchen **Statuswort anzeigen** bzw. **Steuerwort anzeigen** kann die Anzeige des Status- bzw. Steuerwortes ein- bzw. ausgeschaltet werden. Die Anzeige des Steuerwortes kann nicht ausgeschaltet werden, wenn das Statuswort nicht angezeigt wird und umgekehrt.

Der aktuelle Wert des Steuer- bzw. Statusworts wird sowohl als hexadezimale Zahl als auch aufgeschlüsselt nach Bedeutung dargestellt. Die einzelnen Bits werden dabei als LEDs angezeigt. Die oberste LED stellt Bit 15 und die unterste LED Bit 0 des jeweiligen Wortes dar. Da das Steuer- und Statuswort in den verschiedenen Betriebsarten unterschiedlich belegt ist, werden die dargestellten Texte je nach Betriebsart und Bitbedeutung angepasst. Zusätzlich wird beim Steuerwort das zuletzt empfangene **Kommando** und beim Statuswort der aktuelle **Zustand** in Klartext dargestellt. Die Bits, die zur Bestimmung des Kommandos/Zustands ausgewertet werden, sind in eckigen Klammern angegeben.

## 4.6 Diagnose - Alarme

Der Servoregler unterstützt PROFINET Diagnose-Alarme, um Störungen an die Steuerung zu melden. Wenn ein spezielles Problem im Servoregler auftritt, wie zum Beispiel Schleppfehler, Winkelgeberfehler, etc., wird ein Diagnose-Alarm an die Steuerung geschickt. Dies führt zum Ansteuern der roten Diagnose-LED in der Steuerung. Für eine schnelle Analyse wird die Ursache im Klartext in der PROFINET-Steuerung (z.B. in der Kanaldiagnose) angegeben, z.B.:

### **E08: Winkelgeber: E08-6: Kommunikationsfehler Winkelgeber**

Die Fehlernummer setzt sich aus einem Hauptindex (HH, in diesem Beispiel 08) und einem Subindex (S, in diesem Beispiel 6) zusammen.

Der Hauptindex wird im herstellerspezifischen Bereich der Kanaldiagnose (ChannelErrorType) von  $1000_h$  bis  $7FFF_h$  übertragen. Der Subindex wird im herstellerspezifischen Bereich der erweiterten Kanaldiagnose (ExtChannelErrorType) von  $1000_h$  bis  $100F_h$  übertragen.

Für den Fehler 08-6 ergäben sich somit beispielhaft folgende Werte:

Fehlernummer	ChannelErrorType	ExtChannelErrorType
08-6	$HH_h + 1000_h = 1008_h$	$S_h + 1000_h = 1006_h$

## 5 Betriebsarten

### 5.1 Übersicht

Die Servoregler verfügen über 3 Basisbetriebsarten:

- Drehmomentregelung
- Drehzahlregelung
- Positionieren

Bei der Verwendung des Beispiel-Funktionsbausteins erfolgt die Betriebsart-Umschaltung über den Eingang `operation_mode` (siehe Abschnitt 2.5.1.2 *Eingänge/Ausgänge für alle Betriebsarten* auf Seite 17) ohne dass weitere Randbedingungen zu beachten wären. Bei der Verwendung eigener zyklischer Telegramme ist zu beachten, dass die Betriebsart fest an die ersten beiden Telegramme gebunden ist:

Betriebsart	Empfangstelegramm
Positionieren	0
Drehzahlregelung	1

Durch Wechseln des Empfangstelegramms erfolgt automatisch ein Wechsel der Betriebsart. Dies ist im Abschnitt 6.1 *Telegrammeditor* auf Seite 71 noch einmal näher erläutert. Bei Verwendung der Empfangstelegramme 2 und 3, muss die Betriebsart hingegen explizit durch den Parameter *PNU 1500: Operating Mode* vorgegeben werden.

Da der Wechsel der Betriebsart einige Zeit in Anspruch nehmen kann, muss die aktuell gültige Betriebsart durch Lesen der PNU 1500 bestimmt werden. Sie sollte daher in allen Antworttelegrammen (0...3) eingetragen sein.

### 5.2 Betriebsart Drehmomentregelung

In der Betriebsart Drehmomentregelung gibt die Steuerung direkt einen Sollwert für den Drehmomentregler vor. Dieser wird zur Begrenzung der Momentenänderung über eine Sollwertrampe geführt.

Da keine PNU für die Vorgabe eines Drehmoment-Sollwerts existiert, wird der Sollwert über das CAN-Objekt 6071<sub>h</sub> (Abschnitt 3.6.1 *CAN-Objekt 6071h: target\_torque* auf Seite 48) vorgegeben.

## 5.3 Betriebsart Drehzahlregelung

In der Betriebsart Drehzahlregelung gibt die Steuerung direkt einen Sollwert für den Drehzahlregler vor. Dieser wird zur Begrenzung der Beschleunigung bei Drehzahl-Änderungen über eine Sollwertrampe geführt. Der Ausgangswert der Drehzahlrampe kann angehalten („eingefroren“) werden.

Bei Aktivierung der Drehzahlregelung erfolgt zusätzlich zur Betriebsart folgende Einstellung:

- Im Sollwertselektor wird bei Aktivierung der PROFINET-Kommunikation beim Addierer der Feste Sollwert 1 aktiviert. Dieser Selektor wird im Metronix ServoCommander<sup>®</sup> in diesem Fall auch mit PROFINET bezeichnet.
- Wenn der Sollwert durch das Bit `Enable setpoint` in der *PNU 967: Control word 1* gesperrt ist, dann ist im Addierer kein Sollwert aktiviert (d.h. im Metronix ServoCommander<sup>®</sup> ist kein Kontrollkästchen ausgewählt).

## 5.4 Betriebsart Positionieren

In der Betriebsart Positionieren kann der Servoregler selbstständig eine Zielposition anfahren. Beim Verfahrenvorgang werden folgende Parameter (siehe Abschnitt 3.4.2 *PNU 1001: Position Data* auf Seite 33) berücksichtigt:

- Fahrgeschwindigkeit
- Endgeschwindigkeit (Geschwindigkeit im Ziel, in der Regel 0)
- Beschleunigung und Bremsbeschleunigung

Der Start einer Positionierung erfolgt auf eine steigende Flanke von Bit 6 (Activate traversing task) in *PNU 967: Control word 1*. Über weitere Bits im Control word 1 lässt sich festlegen,

- ob die Positionierung absolut oder relativ ausgeführt wird (Bit 12, `Relative`).
- ob beim Start einer Positionierung eine ggf. laufende Positionierung unterbrochen werden soll oder ob die zu startende Positionierung unmittelbar an die laufende Positionierung angehängt wird (Bit 13, `Change immediately`).

In manchen Anwendungen soll eine lückenlose Folge von Fahraufträgen ausgeführt werden, wie Sie auf der rechten Seite der folgenden Abbildung zu sehen ist:

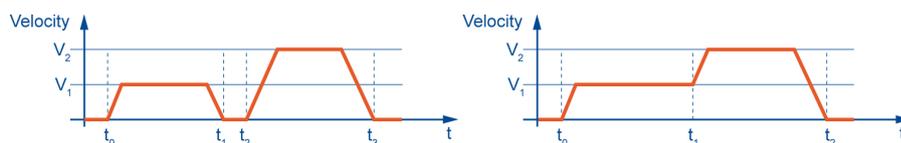


Abbildung 18: Einfacher Fahrauftrag (links) und Folge von Fahraufträgen (rechts)

Dies kann auf zwei verschiedenen Wegen erreicht werden:

- Durch Unterbrechen der laufenden Positionierung
- Durch Starten einer Anschlusspositionierung, wobei für den ersten Fahrauftrag die Endgeschwindigkeit gleich der Fahrgeschwindigkeit ist.

Der erste Fall lässt sich erreichen, indem das Bit Change `immediately` im `Control word 1` beim Starten des zweiten Fahrauftrags gesetzt wird. Der Zeitpunkt  $t_1$  ist in diesem Fall nicht genau definiert und von der Verarbeitung in Steuerung und Regler abhängig. Soll die zweite Positionierung an einer bestimmten Position starten, so muss im ersten Fahrauftrag die Endgeschwindigkeit (PNU 1001.2, `End Velocity`) gleich der Profilgeschwindigkeit (PNU 1001.1, `Profile Velocity`) gesetzt werden. Der zweite Fahrauftrag kann dann unmittelbar nach dem Start des ersten erfolgen (ohne `Change Immediately`). Der Antrieb erreicht dann zum Zeitpunkt  $t_1$  die Zielposition des ersten Fahrauftrags und fährt dann Ziel 2 an, ohne auf Null abzubremesen.

In der Betriebsart Positionierung wird auch die Referenzierung des Antriebs ausgeführt (siehe Abschnitt 3.4.8 *PNU 1050: Homing Method* auf Seite 37).

## 6 Zyklische Kommunikation

Der Austausch von Kommandos, Sollwerten und Istwerten zwischen Steuerung und Servoregler basiert auf einer zyklischen Kommunikation. Für den fehlerfreien Aufbau der zyklischen Kommunikation, müssen die ausgetauschten Telegramme sowohl in der Steuerung als auch im Servoregler übereinstimmend parametrisiert werden. Im Folgenden wird zunächst die Konfiguration im Servoregler und anschließend im TIA-Portal beschrieben.

### 6.1 Telegrammeditor

Mit dem Telegrammeditor wird festgelegt, wie der Servoregler die zwischen Servoregler und Steuerung ausgetauschten Daten in den zyklischen Telegrammen interpretiert. Diese Festlegung muss vor dem Aufbau der zyklischen Kommunikation erfolgen. Zu diesem Zweck verfügt der Metronix ServoCommander® über ein Telegrammeditorfenster, welches über folgenden Menüeintrag aufgerufen wird:

PROFINET: **Parameter/Feldbus/PROFINET/Telegrammeditor**

Es werden folgende Gruppen von Telegrammen unterschieden:

- **Empfangstelegramme:** Übertragene Daten vom Master zum Slave, auch als Ausgangsdaten bezeichnet.
- **Antworttelegramme:** Zu übertragende Daten vom Slave zum Master, auch als Eingangsdaten bezeichnet.

Jedes Telegramm kann maximal 10 Einträge aufweisen.

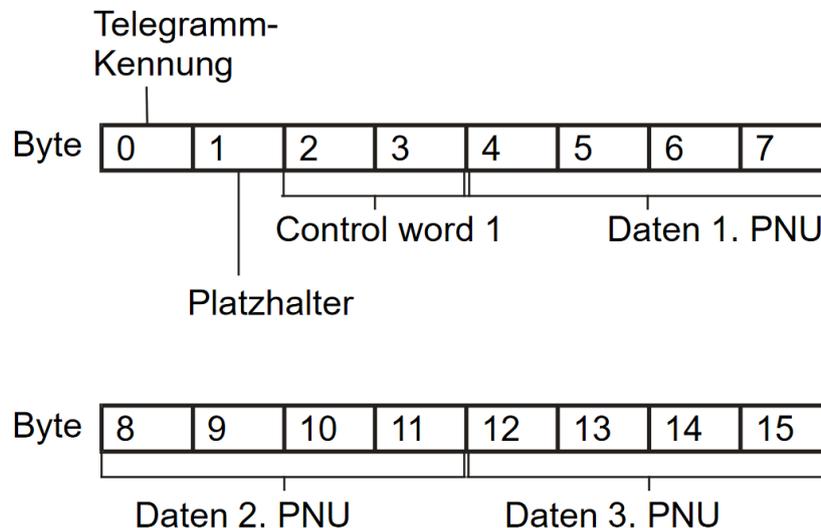


Abbildung 19: Beispiel Telegrammformat

Die Abbildung zeigt das Beispiel eines Standard-Telegramms vom Master zum Slave. Neben der Kennung in Byte 0 erfordert dieser Telegrammtyp in den Bytes 2 und 3 das PROFIdrive **Control word 1** zur Gerätesteuerung. Der Inhalt der nachfolgenden Bytes kann frei konfiguriert werden. In diesem Beispiel werden 3 weitere Daten übertragen, jeweils mit einer Größe von 4 Bytes. Für das gesamte Telegramm ergibt sich hier eine Länge von 16 Bytes.

Im Projekt des Masters werden Datenbereiche erstellt, z.B. Datenbausteine. In diese Datenbereiche werden die Ein- und Ausgangsdaten von Master und Slave abgelegt. Beim Projektieren muss der Anwender die Inhalte und deren Reihenfolge sowie die Größe der beiden Datenbereiche übereinstimmend auf Seiten von Master und Slave angeben. Diese Parametrierung muss vor der Aktivierung der Kommunikation durchgeführt werden.

### › Empfangstelegramme

Der Servoregler unterstützt 4 Empfangstelegramme. Einige dieser Telegramme sind fest an eine Betriebsart gebunden. Dies erleichtert dem Anwender den Wechsel zwischen verschiedenen Betriebsarten. Ein zusätzlicher Parameter für die Betriebsart muss nicht mit übertragen werden. Die nachfolgende Tabelle gibt eine Übersicht über Betriebsartbindung der Empfangstelegramme:

Telegramm	Betriebsart
Empfangstelegramm 0	Positionieren
Empfangstelegramm 1	Drehzahlregelung
Empfangstelegramm 2	keine (reserviert für Drehmomentregelung)
Empfangstelegramm 3	keine (freies Telegrammformat)

Sobald eine entsprechende Telegrammkennung im Servoregler gelesen wird, erfolgt die Überprüfung und ggf. die Parametrierung der entsprechenden Betriebsart.

Für jedes Empfangstelegramm sind die Parameternummern einzutragen. Damit ist die Information über die Bedeutung der Daten im Servoregler hinterlegt. Es ist zu beachten, dass bei den Empfangstelegrammen 0..2 an der Adresse 2 jeweils das **Control word 1** eingetragen ist (Länge: 2 Byte). Diese einheitliche Festlegung erleichtert die Erstellung von Applikationen. Die weiteren Einträge können beliebig aus dem Objektverzeichnis der Parameternummern ausgewählt werden. Hierbei ist lediglich die Eignung zu beachten. Reine Istwertdaten können beispielsweise nicht in Empfangstelegramme eingetragen werden.

Bei den Empfangstelegrammen sind zusätzlich noch die Antworttelegramme zu selektieren. Der Anwender kann für jedes Empfangstelegramm ein eigenes Antworttelegramm festlegen und konfigurieren. In den meisten Fällen ist es jedoch einfacher, für alle Betriebsarten (Empfangstelegramme 0..2) das gleiche Antworttelegramm zu verwenden. Dies reduziert den Programmieraufwand auf Seiten des Masters. Darüber hinaus werden vom Master in der Regel in allen Betriebsarten die gleichen Istwertdaten vom Servoregler benötigt.

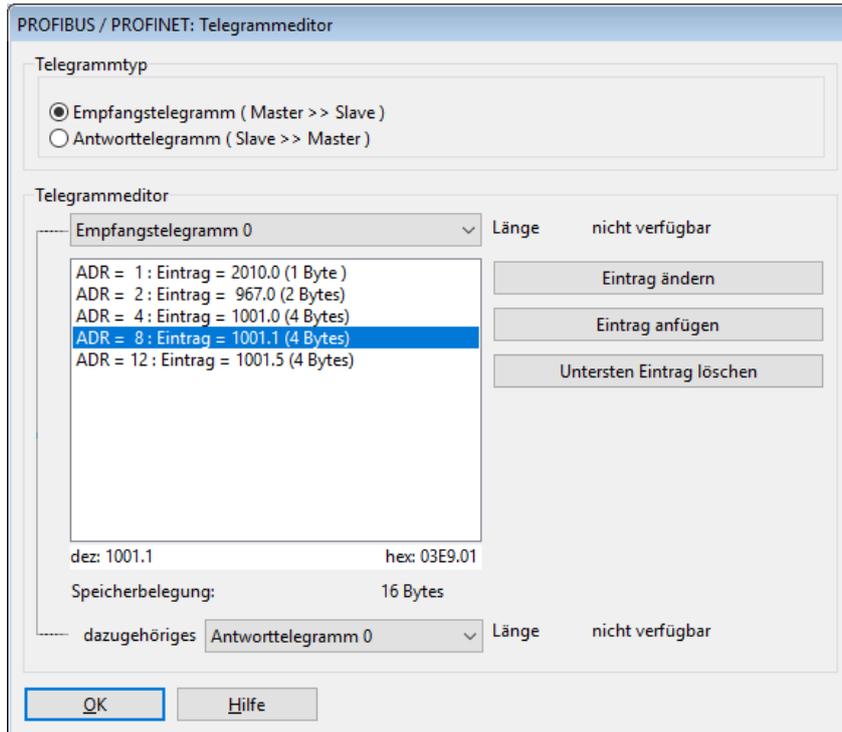


Abbildung 20: Zusammenstellung eines Empfangstelegramms

Das abgebildete Telegrammeditor-Fenster zeigt ein Beispiel für das **Empfangstelegramm 0** (Betriebsart Positionieren). Die Einträge können durch Markieren direkt geändert bzw. vom letzten Eintrag ausgehend sukzessive gelöscht werden. Beim Markieren eines Eintrages erscheint ein zusätzliches Feld, in dem die Parameternummer eingegeben werden kann:

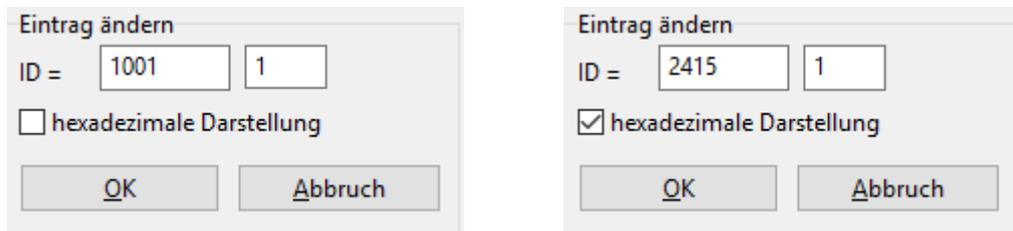


Abbildung 21: Eingabe einer PNU (links) oder eines CAN-Objektes (rechts)

Zur Eingabe einer PNU wird die Nummer dezimal eingegeben (Kontrollkästchen **hexadezimale Darstellung** nicht markiert). Falls Objekte aus dem CANopen Objektverzeichnis genutzt werden, muss das Kontrollkästchen **hexadezimale Darstellung** markiert werden. Die im CANopen Handbuch aufgeführte Nummer kann dann direkt übernommen werden. Sobald ein Eintrag im Feld ID eingegeben und mit OK bestätigt wird, erfolgt ein Handshake mit dem angeschlossenen Servoregler. Es wird geprüft, ob der Parameter existiert, und es wird die Anzahl Bytes für diesen Parameter ermittelt. Daher ist auch diese Funktion im Offline-Modus des Parametrierprogramms nicht verfügbar.

Neue Telegrammeinträge werden am Ende angefügt. Sofern zwischen Master und Slave eine Kommunikation etabliert werden konnte, wird über der Schaltfläche „Eintrag ändern“ eine zusätzliche Diagnoseinformation eingeblendet. Es wird die vom Master konfigurierte tatsächliche Länge des Telegramms vom Master zum Slave angezeigt. Im oben dargestellten Beispiel werden folgende Parameter übertragen:

Adresse	Inhalt (Parameternummer)	Beschreibung
0	Kennung (= E0 <sub>h</sub> )	Fest eingestellte Kennung
1	8 Bit Platzhalter (PNU 2010.0)	Frei
2	Control word 1 (PNU 967.0)	Steuerwort zur Gerätesteuerung, muss fest an dieser Adresse liegen
4	Zielposition (PNU 1001.0)	Zielposition, Angabe in der eingestellten physikalischen Einheit einer Position
12	Beschleunigungen (PNU 1001.5)	Kombination der Werte für Beschleunigung und Bremsbeschleunigung, Angabe in der eingestellten physikalischen Einheit einer Beschleunigung

### › Antworttelegramme

Der Servoregler unterstützt 4 Antworttelegramme, die analog zu den Empfangstelegrammen parametrisiert werden. Für jedes Antworttelegramm sind die Parameternummern einzutragen. Damit ist die Information über die Bedeutung der Daten im Servoregler hinterlegt. Es ist zu beachten, dass bei den Antworttelegrammen 0..2 an der Adresse 2 jeweils das so genannte **Status word 1** eingetragen ist (Länge: 2 Byte). Diese einheitliche Festlegung erleichtert die Erstellung von Applikationen. Die weiteren Einträge können beliebig aus dem Objektverzeichnis der Parameternummern ausgewählt werden. Hierbei ist lediglich die Eignung zu beachten. Parameter, die nur geschrieben werden können, können beispielsweise nicht in Antworttelegramme eingetragen werden.

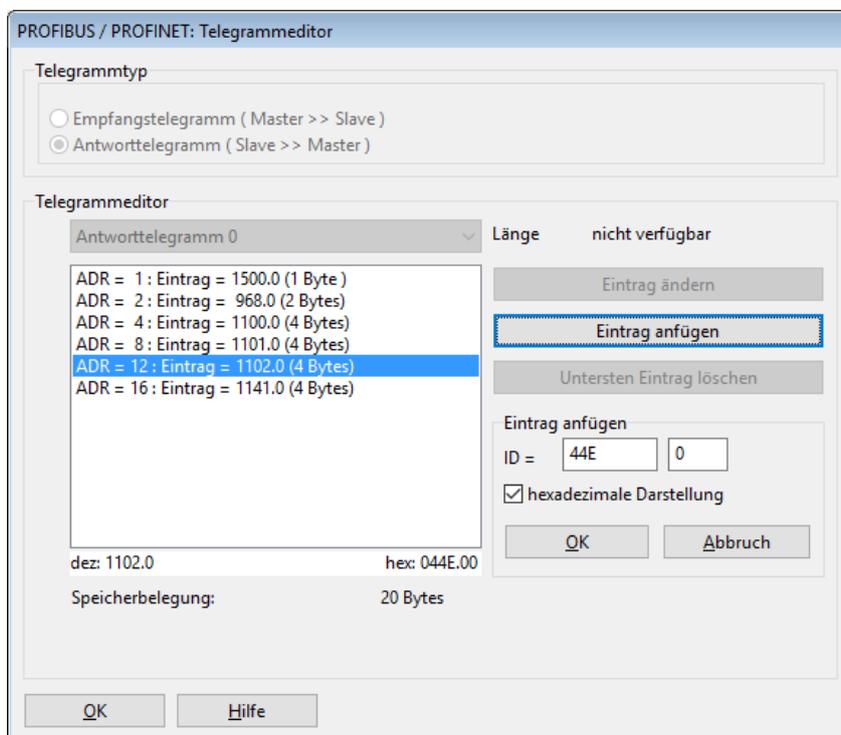


Abbildung 22: Zusammenstellung eines Antworttelegramms

Das abgebildete Telegrammeditor-Fenster zeigt ein Beispiel für das Antworttelegramm 0. Die Einträge können durch Markieren direkt geändert bzw. vom letzten Eintrag ausgehend sukzessive gelöscht werden. Beim Markieren eines Eintrages erscheint ein zusätzliches Feld, in dem die Parameternummer eingegeben werden kann. Neue Telegrammeinträge werden am Ende angefügt. Sofern zwischen Master und Slave eine Kommunikation etabliert werden konnte, wird über der Schaltfläche **Eintrag ändern** eine zusätzliche Diagnoseinformation eingeblendet. Es wird die vom Master konfigurierte tatsächliche Länge des Telegramms vom Slave zum Master angezeigt. Im oben dargestellten Beispiel werden folgende Parameter übertragen:

Adresse	Inhalt (Parameternummer)	Beschreibung
0	Kennung (= F0 <sub>h</sub> )	Fest eingestellte Kennung
1	Betriebsart (PNU 1500.0)	Aktuelle Betriebsart des Servoreglers
2	Status word 1 (PNU 968.0)	Statuswort zur Gerätesteuerung, muss fest an dieser Adresse liegen
4	Istposition (PNU 1100.0)	Aktuelle Istposition, Angabe in der eingestellten physikalischen Einheit einer Position
8	Drehzahlistwert (PNU 1101.0)	Aktueller Drehzahlistwert, Angabe in der eingestellten physikalischen Einheit einer Drehzahl
12	Wirkstromistwert (PNU 1102.0)	Über diesen Parameter wird der Wirkstrom-Istwert gelesen. Dieser wird bezogen auf den Motornennstrom zurückgegeben
16	Status digitale Eingänge (PNU 1141.0)	Aktueller Status der digitalen Eingänge, Bitbelegung siehe Beschreibung der PNU.

## 6.1.1 Anzeige der aktuellen Telegramm Daten

Zur Vereinfachung der Inbetriebnahme besitzt das Parametrierprogramm Metronix ServoCommander® ein Fenster, in dem die aktuellen Daten des Empfangs- bzw. Antworttelegramms angezeigt werden. Dadurch ist es zum Beispiel möglich Zuordnungen von Daten bzw. Vertauschungen in Datenfeldern zu erkennen. Die Reihenfolge der Bytes für 2 Byte bzw. 4 Byte Datentypen entspricht stets „Low Byte ... High Byte“.

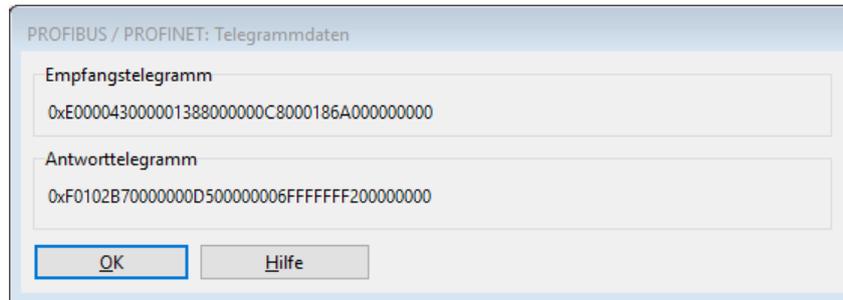


Abbildung 23: Fenster "PROFIBUS/PROFINET: Telegramm Daten"

Die angezeigten Daten lassen sich wie folgt interpretieren: Das ganze linke Byte entspricht dem Telegrammkopf (1 Byte), anschließend folgen die Daten, wie im Telegrammeditor angegeben (siehe Abschnitt 6.1 *Telegrammeditor* auf Seite 71), d.h. Einträge von oben nach unten im Editor entsprechen Werten von links nach rechts in den Telegramm Daten.

## 6.2 Konfiguration der Telegramme im TIA-Portal

Die Länge der Ein- und Ausgangsdaten muss im TIA-Portal passend zum Servoregler festgelegt werden. Dazu müssen in der **Gerätesicht** die Metronix Servoregler ausgewählt werden. Anschließend können die Telegrammmoduldaten konfiguriert werden. Es werden per Drag&Drop die IN- und OUT-Module aus dem Hardware-Katalog in die leeren Zeilen der Gerätesicht verschoben.

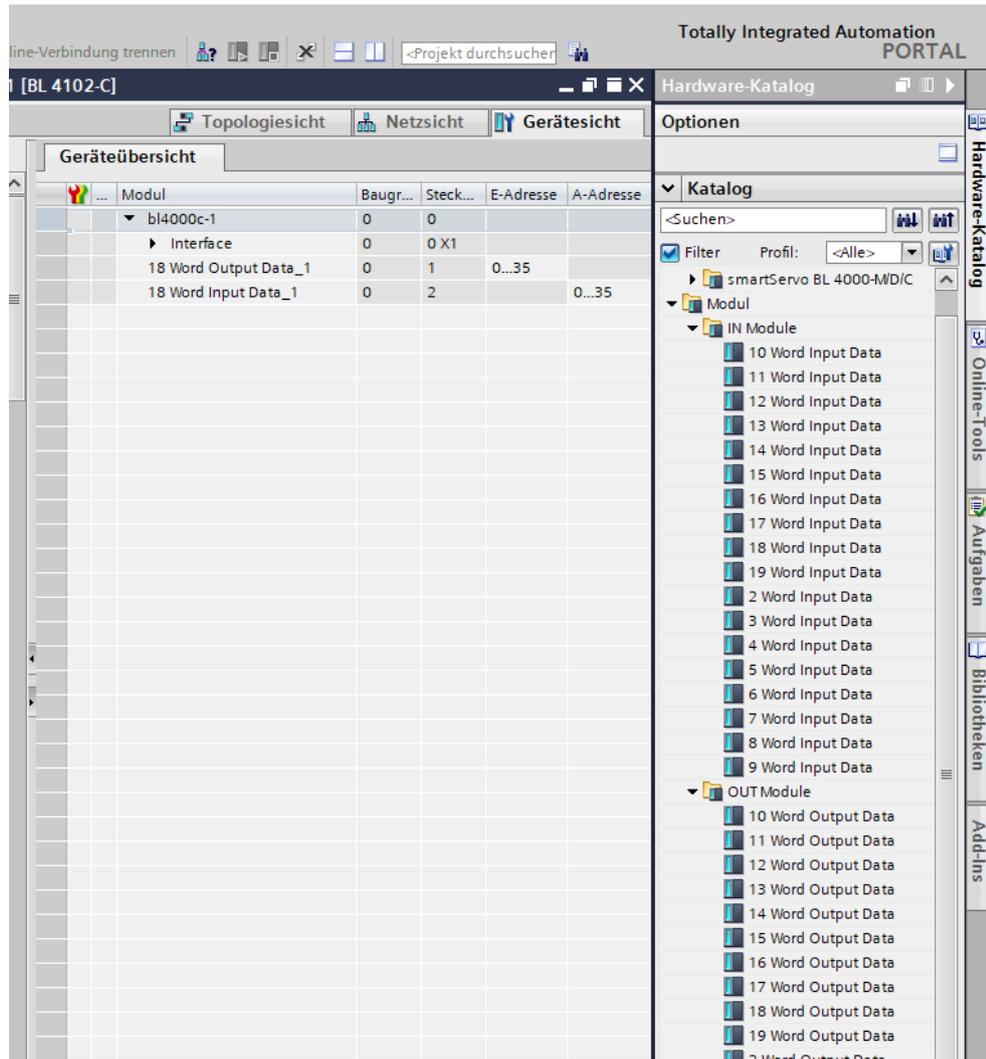


Abbildung 24: Konfiguration der Ein-/Ausgangsdaten im TIA-Portal

Im Unterschied zur Konfiguration im Servoregler, muss an dieser Stelle nicht die genaue Bedeutung der Daten parametrisiert werden, sondern nur die Länge. Dafür stehen Module von 2 Worten bis zu 19 Worten zur Verfügung.

Das Empfangstelegramm entspricht den Input-Daten, das Antworttelegramm den Output-Daten.

# 7 Anhang

## 7.1 Einstellung für den Beispiel-Funktionsbaustein

Bei Verwendung des Beispiel-Funktionsbausteins sind folgende Telegramm-Einstellungen notwendig, die über die passende Parameterdatei gleichzeitig im Servoregler eingestellt werden können (Abschnitt 2.5 *Einbinden des Servoreglers im TIA-Portal* auf Seite 14). Alle Telegramme haben eine einheitliche Länge von 36 Byte:

### › Telegramm 0

Adresse	PNU	Beschreibung
0	E0 <sub>h</sub>	Fest eingestellte Kennung
1	1050.0	<i>PNU 1050: Homing Method</i>
2	967.0	<i>PNU 967: Control word 1</i>
4	1001.0	<i>PNU 968: Status word 1</i>
8	1001.1	Profile Velocity, <i>PNU 1001: Position Data</i>
12	1001.2	End Velocity, <i>PNU 1001: Position Data</i>
16	1001.5	All Accelerations Positioning, <i>PNU 1001: Position Data</i>
20	1040.0	Jogging Velocity, <i>PNU 1040: Symmetrical Jogging</i>
24	2000.0	<i>PNU 2000: PKW Access</i>
32	2415 <sub>h_02h</sub>	<i>CAN-Objekt 2415h: current_limitation</i>

Tabelle 1: Empfangstelegramm 0 für Beispiel-Funktionsbaustein

Adresse	PNU	Beschreibung
0	F0 <sub>h</sub>	Fest eingestellte Kennung
1	1500.0	<i>PNU 1500: Operating Mode</i>
2	968.0	<i>PNU 968: Status word 1</i>
4	1100.0	<i>PNU 1100: Position Actual Value</i>
8	1101.0	<i>PNU 1101: Velocity Actual Value</i>
12	6077 <sub>h</sub>	<i>CAN-Objekt 6077h: torque_actual_value</i>
14	1141.0	<i>PNU 1141: Digital Inputs</i>
18	2010.1	<i>PNU 2010: Placeholder</i>
20	2010.2	<i>PNU 2010: Placeholder</i>
24	2000.0	<i>PNU 2000: PKW Access</i>
32	2010.2	<i>PNU 2010: Placeholder</i>

Tabelle 2: Antworttelegramm 0 für Beispiel-Funktionsbaustein

> **Telegramm 1**

Adresse	PNU	Beschreibung
0	E1 <sub>h</sub>	Fest eingestellte Kennung
1	2010.0	<i>PNU 2010: Placeholder</i>
2	967.0	<i>PNU 967: Control word 1</i>
4	1010.0	<i>PNU 1010: Target Velocity</i>
8	2010.2	<i>PNU 2010: Placeholder</i>
12	2010.2	<i>PNU 2010: Placeholder</i>
16	1011.2	All Accelerations Velocity Control, <i>PNU 1011: Accelerations for Velocity Control</i>
20	1040.0	Jogging Velocity, <i>PNU 1040: Symmetrical Jogging</i>
24	2000.0	<i>PNU 2000: PKW Access</i>
32	2415 <sub>h</sub> _02 <sub>h</sub>	<i>CAN-Objekt 2415h: current_limitation</i>

Tabelle 3: Empfangstelegramm 1 für Beispiel-Funktionsbaustein

Adresse	PNU	Beschreibung
0	F1 <sub>h</sub>	Fest eingestellte Kennung
1	1500.0	<i>PNU 1500: Operating Mode</i>
2	968.0	<i>PNU 968: Status word 1</i>
4	1100.0	<i>PNU 1100: Position Actual Value</i>
8	1101.0	<i>PNU 1101: Velocity Actual Value</i>
12	6077 <sub>h</sub>	<i>CAN-Objekt 6077h: torque_actual_value</i>
14	1141.0	<i>PNU 1141: Digital Inputs</i>
18	2010.1	<i>PNU 2010: Placeholder</i>
20	2010.2	<i>PNU 2010: Placeholder</i>
24	2000.0	<i>PNU 2000: PKW Access</i>
32	2010.2	<i>PNU 2010: Placeholder</i>

Tabelle 4: Antworttelegramm 1 für Beispiel-Funktionsbaustein

› **Telegramm 2**

Adresse	PNU	Beschreibung
0	E2 <sub>h</sub>	Fest eingestellte Kennung
1	2010.2	<i>PNU 2010: Placeholder</i>
2	967.0	<i>PNU 967: Control word 1</i>
4	6071 <sub>h</sub>	<i>CAN-Objekt 6071h: target_torque</i>
6	2010.1	<i>PNU 2010: Placeholder</i>
8	2010.2	<i>PNU 2010: Placeholder</i>
12	2010.2	<i>PNU 2010: Placeholder</i>
16	2010.2	<i>PNU 2010: Placeholder</i>
20	2010.2	<i>PNU 2010: Placeholder</i>
24	2000.0	<i>PNU 2000: PKW Access</i>
32	2416 <sub>h_02h</sub>	<i>CAN-Objekt 2416h: speed_limitation</i>

Tabelle 5: Empfangstelegramm 2 für Beispiel-Funktionsbaustein

Adresse	PNU	Beschreibung
0	F2 <sub>h</sub>	Fest eingestellte Kennung
1	1500.0	<i>PNU 1500: Operating Mode</i>
2	968.0	<i>PNU 968: Status word 1</i>
4	1100.0	<i>PNU 1100: Position Actual Value</i>
8	1101.0	<i>PNU 1101: Velocity Actual Value</i>
12	6077 <sub>h</sub>	<i>CAN-Objekt 6077h: torque_actual_value</i>
14	1141.0	<i>PNU 1141: Digital Inputs</i>
18	2010.1	<i>PNU 2010: Placeholder</i>
20	2010.2	<i>PNU 2010: Placeholder</i>
24	2000.0	<i>PNU 2000: PKW Access</i>
32	2010.2	<i>PNU 2010: Placeholder</i>

Tabelle 6: Antworttelegramm 2 für Beispiel-Funktionsbaustein