Applikations-Dokument

„servoTEC S2 FS Achsen bei Ansteuerung durch eine Siemens S7 über Profinet-IO“

Ausgabe 08/2024

Art.-Nr.: 1434514

Warenzeichen und Warennamen sind ohne Gewährleistung der freien Verwendbarkeit benutzt. Bei der Erstellung der Texte und Beispiele wurde mit großer Sorgfalt vorgegangen. Trotzdem können Fehler nicht ausgeschlossen werden. Die IEF-Werner GmbH kann für fehlende oder fehlerhafte Angaben und deren Folgen weder eine juristische Verantwortung noch irgendeine Haftung übernehmen.

Die IEF-Werner GmbH behält sich das Recht vor, ohne Ankündigung die Software oder Hardware oder Teile davon, sowie die mitgelieferten Druckschriften oder Teile davon zu verändern oder zu verbessern.

Alle Rechte der Vervielfältigung, der fotomechanischen Wiedergabe, auch auszugsweise sind ausdrücklich der IEF-Werner GmbH vorbehalten.

Für Verbesserungsvorschläge und Hinweise auf Fehler sind wir jederzeit dankbar.

© August 2024, IEF-Werner GmbH

Inhaltsverzeichnis

1 Änderungen dieser Applikationsschrift 5

2 Vorrausetzungen für die Inbetriebnahme 6

2.1 Parametriersoftware „S2_Commander“ herunterladen / aktualisieren 6

2.2 Parametrierschnittstellen am servoTEC S2 FS 6

2.3 Versorgung servoTEC S2 FS 7

2.4 Grundparametrierung des Geräts 7

2.5 Sicherheitsmodul 7

3 Einstellungen am servoTEC S2 FS mit dem S2 Commander 8

3.1 Physikalische Einheiten 9

3.1.1 Allgemeine Information 9

3.1.2 Anzeigeeinheiten S2 Commander 10

3.2 Betriebsparameter PROFINET 13

4 Einstellungen unter SIEMENS TIA Portal 14

4.1 Beispielprojekt 14

4.2 Hardwarekonfiguration Schritt 1: Gerät anwählen 14

4.3 Hardwarekonfiguration Schritt 2: IO-Controller zuweisen 15

4.4 Hardwarekonfiguration Schritt 3: IO-Controller auswählen 16

4.5 Hardwarekonfiguration Schritt 4: Zuweisung Input-Daten (SPS) 17

4.6 Hardwarekonfiguration Schritt 5: Zuweisung Output-Daten (SPS) 17

4.7 Zuordnung der Daten kontrollieren 18

4.8 Hardwarekonfiguration Schritt 6: IP-Adresse und Subnetz einstellen 19

4.9 Hardwarekonfiguration Schritt 7: HW-Kennungen prüfen 20

5 Mindestanforderungen für Integration in eigenes Projekt 22

6 Datenschnittstelle zwischen SPS und servoTEC S2 (z. B. DB2) 27

6.1 Standartsteuerbereich (→ Achse wird angesteuert) 27

6.1.1 Kommandos an Achse 27

6.1.2 Statusmeldungen von Achse 30

6.2 Erweiterter Steuerbereich (→ Parameter im servoTEC S2 FS werden eingestellt) 31

6.2.1 Kommandos an Controller 31

6.2.2 Statusmeldungen von Controller 33

6.3 Beispiel Achskommunikation [DB2] → Linearachse mit Absolutgeber 34

7 Ablaufdiagramme 37

7.1 Initialisierung des Antriebs 37

7.2 Referenzieren des Antriebs 38

7.3 Antrieb Absolut positionieren 39

8 Ergänzende Informationen 40

8.1 Telegramme (Ein- und Ausgangsdaten) 40

8.1.1 Empfangstelegramm für den servoTEC S2-Treiber 40

8.1.2 Antworttelegramm für den servoTEC S2-Treiber 41

1 Änderungen dieser Applikationsschrift

Dokumentenänderungen und Lebenslauf

Dokumentencode	Datum	Erstellung und Änderung
APP5048_DE_1434514_servoTECS2_Profinet-IO_SiemensS7_R1a.docx	06.05.2020	Vorlage modifiziert für Profinet-IO (SCL) → Version: fbservoTEC-S2_d (V1.3)
APP5048_DE_1434514_servoTECS2_Profinet-IO_SiemensS7_R2a.docx	30.06.2020	Vorgabe Ruckwert integriert, Nahtstelle modifiziert → Version: fbservoTEC-S2_d (V2.0)
APP5048_DE_1434514_servoTECS2_Profinet-IO_SiemensS7_R2b.docx	11.08.2021	Korrektur in Hardwarekonfiguration → Zuweisung der Input- und Output-Daten
APP5048_DE_1434514_servoTECS2_Profinet-IO_SiemensS7_R2c.docx	15.01.2024	Applikationsschrift überarbeitet durch Service
APP5048_DE_1434514_servoTECS2_Profinet-IO_SiemensS7_R3.docx	15.02.2024	Applikationsschrift erneut überarbeitet durch Service und Dokumentation → Release R3!
APP5048_DE_1434514_servoTECS2_Profinet-IO_SiemensS7_R3a.docx	15.08.2024	Änderungen in den Kapiteln 4.5 und 4.6. Neu hinzu: Kapitel 4.7.
APP5048_DE_1434514_servoTECS2_Profinet-IO_SiemensS7_R3b.docx	20.08.2024	Erneut neue Grafiken / Screenshots in den Abschnitten 4.5, 4.6 und 4.9.

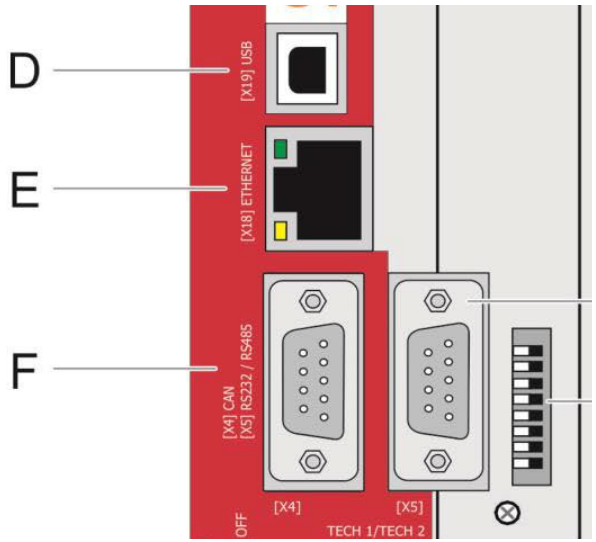
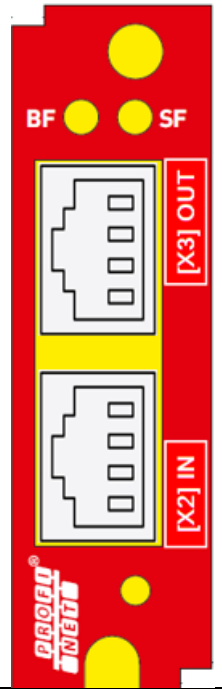
2 Vorrausetzungen für die Inbetriebnahme

Für eine erfolgreiche Inbetriebnahme müssen die folgenden Grundvoraussetzungen erfüllt sein:

2.1 Parametriersoftware „S2_Commander“ herunterladen / aktualisieren

Auf unserer Website (www.ief.de) finden Sie im Downloadbereich immer den aktuellsten „S2_Commander“. Bitte stellen Sie sicher, dass dieser auf ihrem Programmiergerät installiert ist.

2.2 Parametrierschnittstellen am servoTEC S2 FS

<table>
<tr>
<td rowspan="3"> 
</td>
<td>D: [X19] USB Typ-B Schnittstelle (empfohlen)</td>
<td rowspan="3"> 

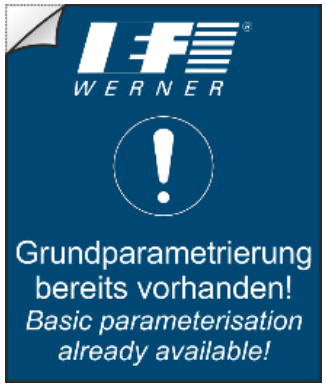
PN Technologiemodul</td>
</tr>
<tr>
<td>E: [X18] Ethernet-Schnittstelle
Standart-IP 10.1.6.1
Standard SUBNET Mask 255.255.0.0</td>
</tr>
<tr>
<td>F: [X5] RS232-Schnittstellen</td>
</tr>
</table>

2.3 Versorgung servoTEC S2 FS

Stellen Sie sicher, dass die Versorgungsspannung (Zwischenkreisspannung vorhanden) sowie die +24V-Versorgung am Gerät anliegen (7-Segement-Anzeige am Verstärker leuchtet). Ohne die Versorgungsspannungen kann das Gerät (servoTEC S2 FS) nicht erfolgreich In Betrieb genommen werden.

2.4 Grundparametrierung des Geräts

Das Gerät servoTEC S2 FS erhält seitens der IEF-Werner GmbH eine Grundparametrierung passend zum Paket, bestehend aus: servoTEC S2 FS, Motor und Achse. Dies ist anhand des Aufklebers über der USB/Ethernet- Parametrierschnittstelle ersichtlich. Für Informationen zur Bedienung und Fehlermeldungen öffnen Sie bitte im S2 Commander den Reiter „Hilfe“ und wählen „Themen“ an. Sollten Unklarheiten auftreten, setzen Sie sich bitte mit unserem Kundensupport in Verbindung.

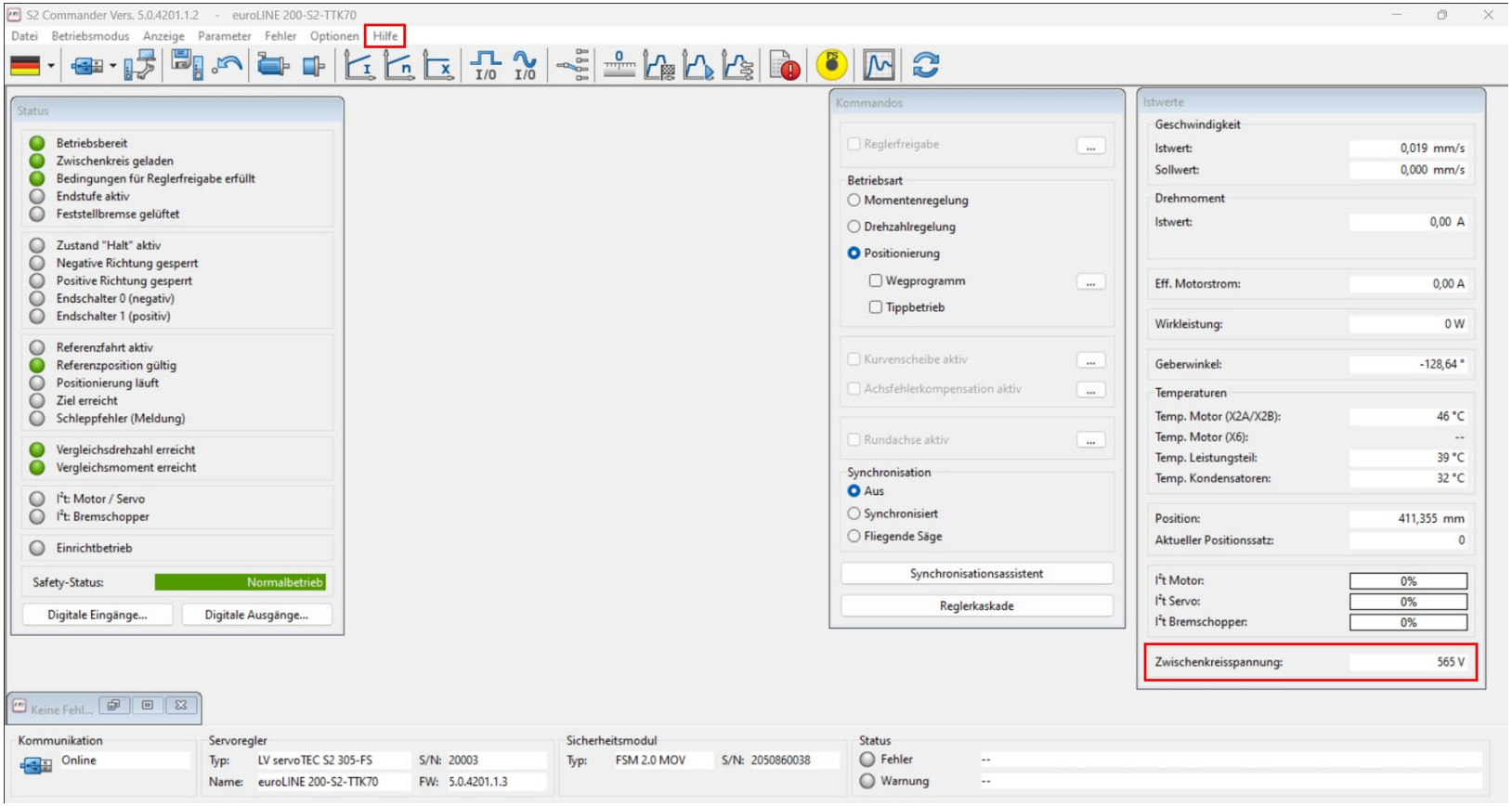


2.5 Sicherheitsmodul

Falls das Gerät ein Sicherheitsmodul besitzt, ist darauf zu achten, dass das Gerät entsprechend des mitgelieferten Schaltplans oder der Bedienungsanleitung angeschlossen ist und angesteuert werden kann. Andernfalls ist keine vollständige Inbetriebnahme des servoTEC S2 FS möglich.
Am STO- bzw. MOV-Modul muss die LED “State” grün leuchten, dies bedeutet „sicherer Zustand erreicht“.
Die Betriebsanleitungen zu beiden Modulen finden Sie auf unserer Website (www.ief.de) im Downloadbereich.

3 Einstellungen am servoTEC S2 FS mit dem S2 Commander

Die Einstellungen für den Antrieb werden mit dem S2 Commander durchgeführt.



Standard IP: 10.1.6.1

Subnetzmaske: 255.255.0.0

	Im S2 Commander unter dem Bereich „Hilfe“ ▶ „Themen…“ finden sich hilfreiche Informationen zur Handhabung und Fehlermeldungen. **Die Zwischenkreisspannung muss bei einem einphasigen Gerät bei > 300V DC und bei einem dreiphasigen Gerät > 500V DC betragen, andernfalls fehlt die Versorgungsspannung.**

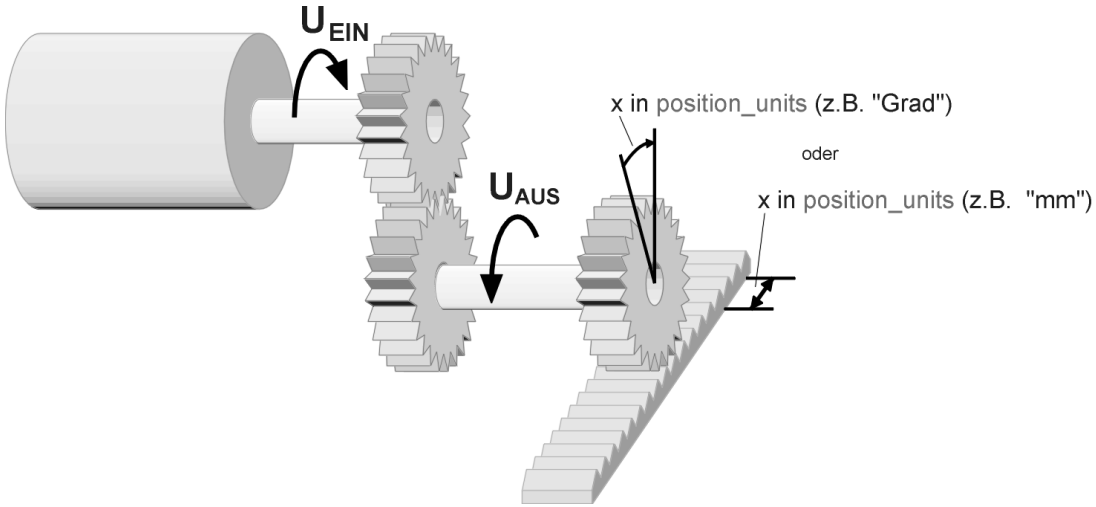
3.1 Physikalische Einheiten

3.1.1 Allgemeine Information

Für die korrekte Funktion und zur Erleichterung der Inbetriebnahme ist es wichtig, dass die physikalischen Einheiten des Feldbussystems (Profinet IO) und der Diagnoseoberfläche (S2 Commander) richtig eingestellt sind.

Die Einstellung der physikalischen Einheiten erfolgt mit dem S2 Commander in verschiedenen Fenstern:

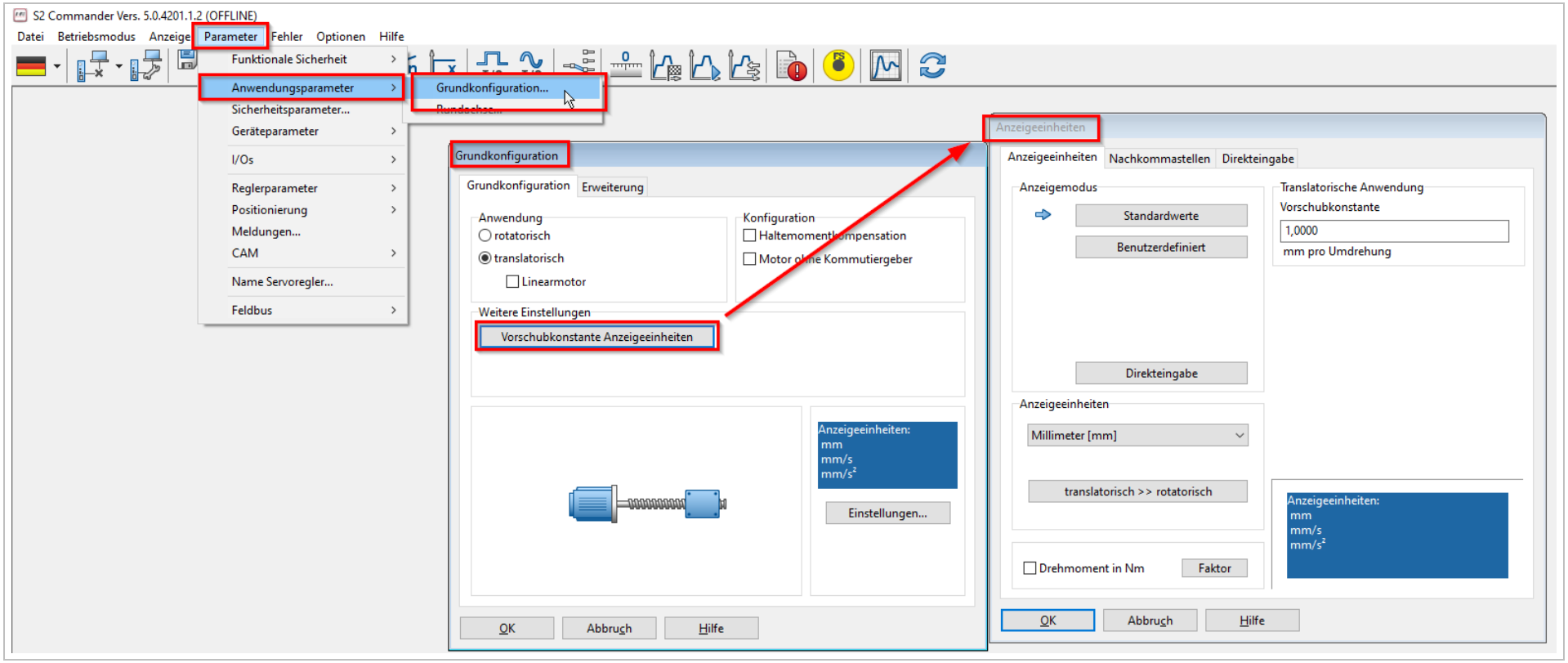
- Für die Diagnose mit S2 Commander → Anzeigeeinheiten
- Für den Profinet IO → Physikalische Einheiten Profinet IO
  → Physikalische Einheiten CANopen / Factor Group



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 parametriert, dann wird die Vorschubkonstante nicht berücksichtigt.

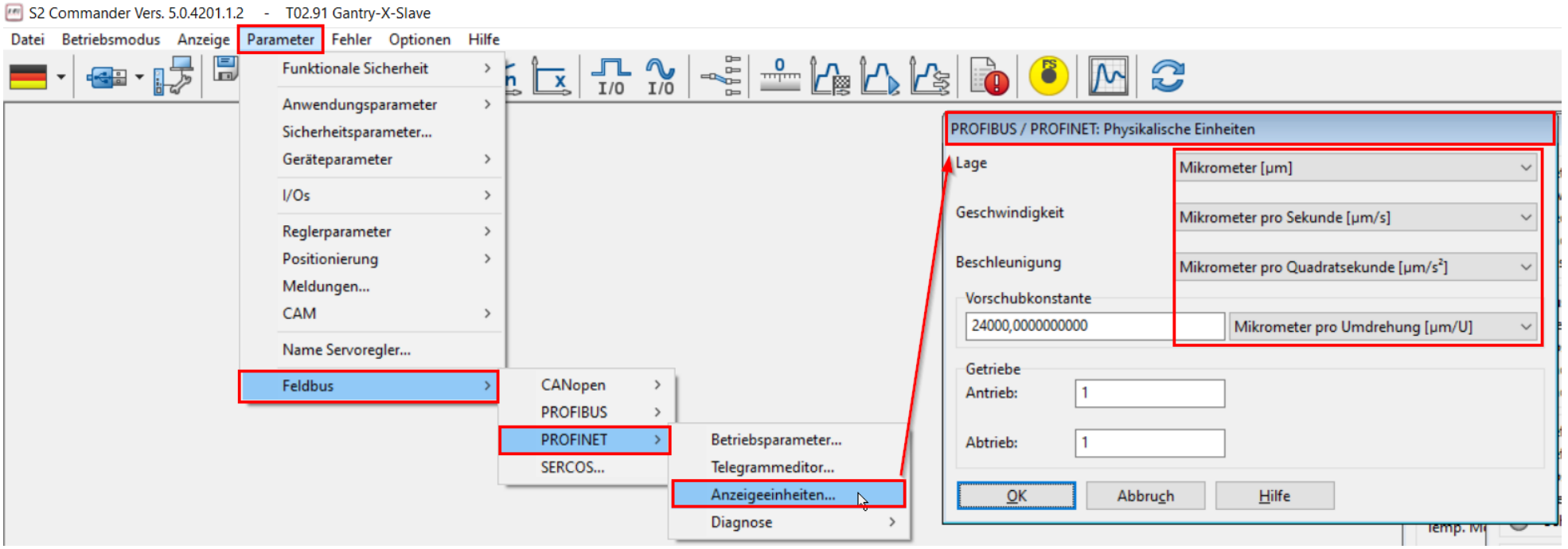
3.1.2 Anzeigeeinheiten S2 Commander

- Die Anzeigeeinheiten für den S2 Commander werden im Fenster „Anzeigeeinheiten“ eingestellt.
- **Prüfen Sie**, ob die im S2 Commander eingetragene Vorschubkonstante mit der auf dem Typenschild der mechanischen Achse (Linearachse, Drehachse, etc.) übereinstimmt.



3.1.2.1 Physikalische Einheiten Profinet-IO

Für die korrekte Funktion ist es notwendig, dass die Einheiten der über den Feldbus übertragenen Prozessdaten festgelegt werden. Diese können über das Fenster **PROFIBUS / PROFINET: Physikalische Einheiten** unter **Parameter ▶ Feldbus ▶ PROFINET ▶ Anzeigeeinheiten…** eingestellt werden. **Die Parameter für die physikalischen Einheiten müssen einmalig eingestellt und nicht während einer laufenden Applikation geändert werden.**



Bei Auswahl der Einheiten werden intern entsprechende Faktoren berechnet, so dass der Anwender die gewünschte Einheit nur noch selektieren muss. Der Getriebefaktor und die Vorschubkonstante werden als separate Parameter angegeben. **Wichtig: Die Physikalischen Einheiten sind in µm anzugeben! In der SPS kann weiterhin in mm gearbeitet werden. Dementsprechend muss hier die Vorschubkonstante um den Faktor 1000 größer angegeben werden. Dies ist die Voraussetzung, damit der Treiberbaustein von IEF-Werner korrekt funktioniert.**

Ergänzende Information: Aus den eingestellten physikalischen Einheiten werden bei der Eingabe in der Firmware automatisch Konvertierungsfaktoren gewonnen. Diese bestehen aus Zähler und Nenner, die jeweils nicht größer als 32 Bit werden dürfen. 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.

3.1.2.2 Physikalische Einheiten CANopen-Bus (relevant für PROFINET)

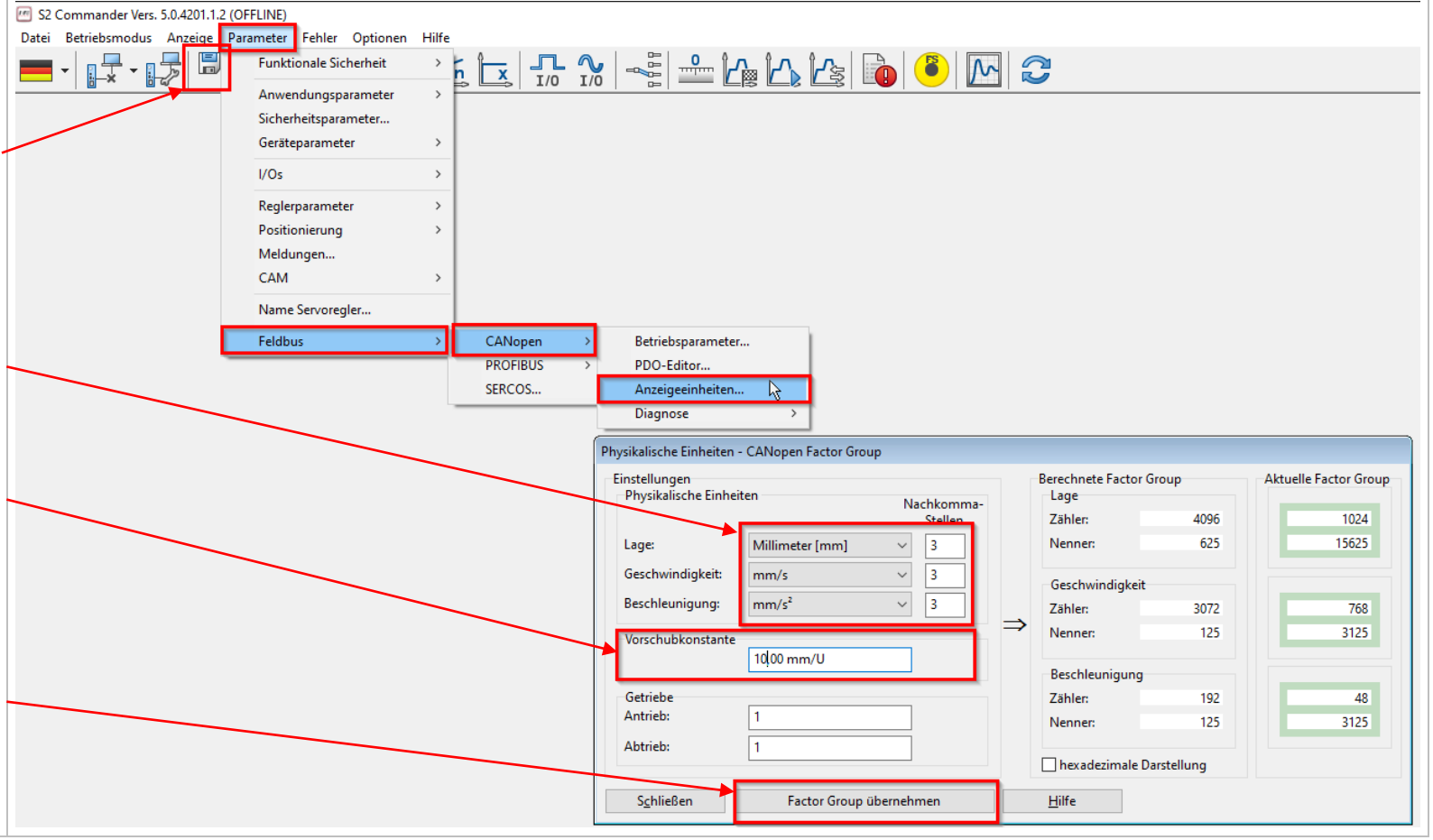
Einzustellen sind:

Nach einer Änderung muss der Parametersatz gesichert werden

Einheiten in mm, U, ° und immer drei Nachkommastellen

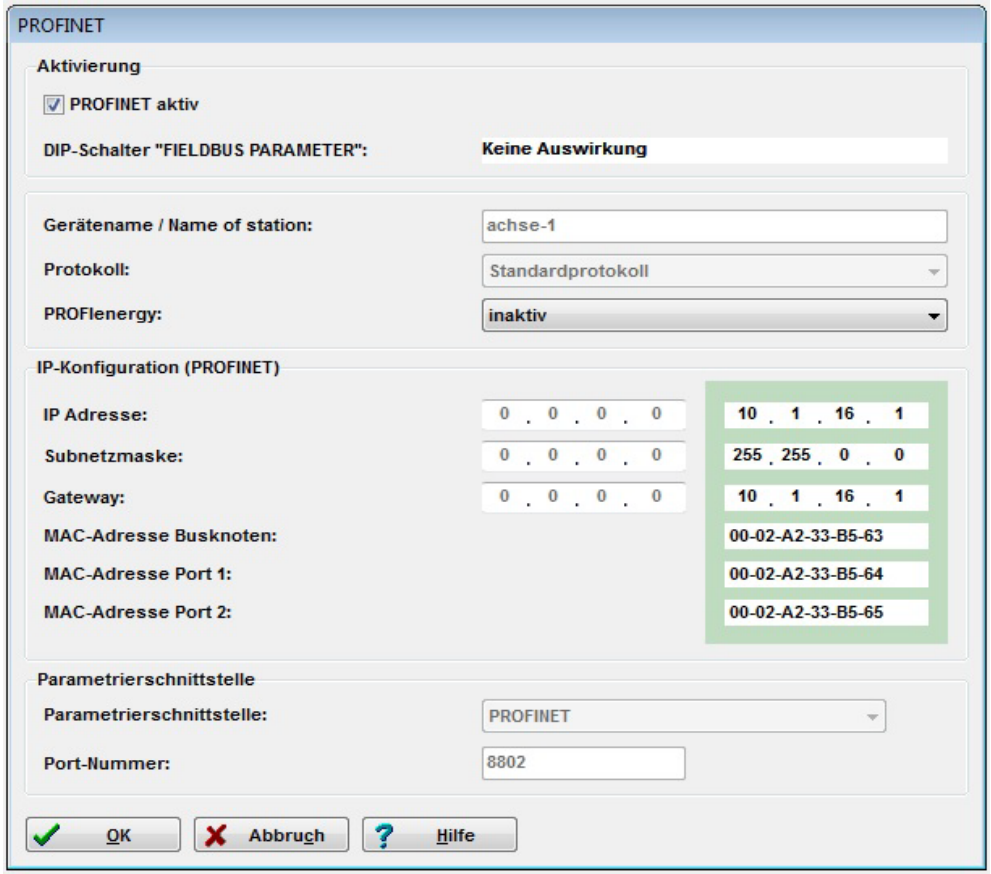
Vorschubkonstante (vom Typenschild des Geräts)

Übernehmen in Factor Group und dann Speichern der Einstellungen!



Damit für die Referenzfahrt über PROFINET die Parameter des servoTEC S2 FS richtig interpretiert werden, ist es erforderlich, die physikalischen Einheiten (Anzeigeeinheiten) für den CANopen-Bus richtig einzustellen. Anschließend ist die Factor Group im Servopositionierregler zu übernehmen und speichern.

3.2 Betriebsparameter PROFINET



Konfiguration der Profinet-Kommunikation auf Seiten des Servopositionierreglers (Parameter ▶ Feldbus ▶ PROFINET ▶ Betriebsparameter).

Vorgehensweise:

1. „**PROFINET aktiv**“ deaktivieren
2. „Gerätename“ eingeben
3. „IP-Adresse“ eingeben
4. „Subnetzmaske“ eingeben
5. „Parametrierschnittstelle“ vorgeben
6. „**PROFINET aktiv**“ aktivieren

Es ist zu beachten, dass die Aktivierung der Profinet-Kommunikation erst nach einigen Sekunden vollständig angezeigt wird.

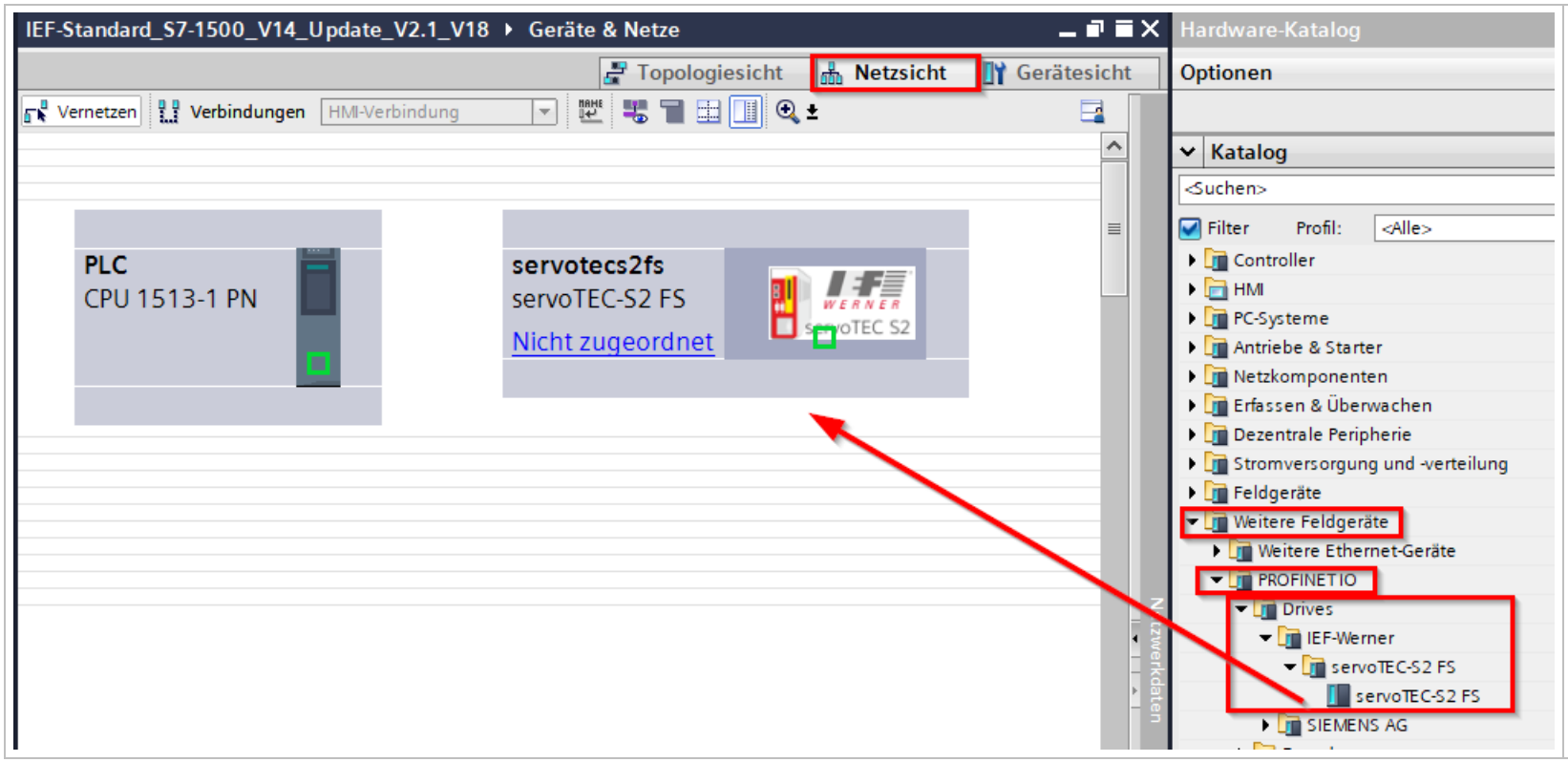
SPEICHERN! → Die Parametrierung ist nun dauerhaft im Servoverstärker gespeichert.

4 Einstellungen unter SIEMENS TIA Portal

4.1 Beispielprojekt

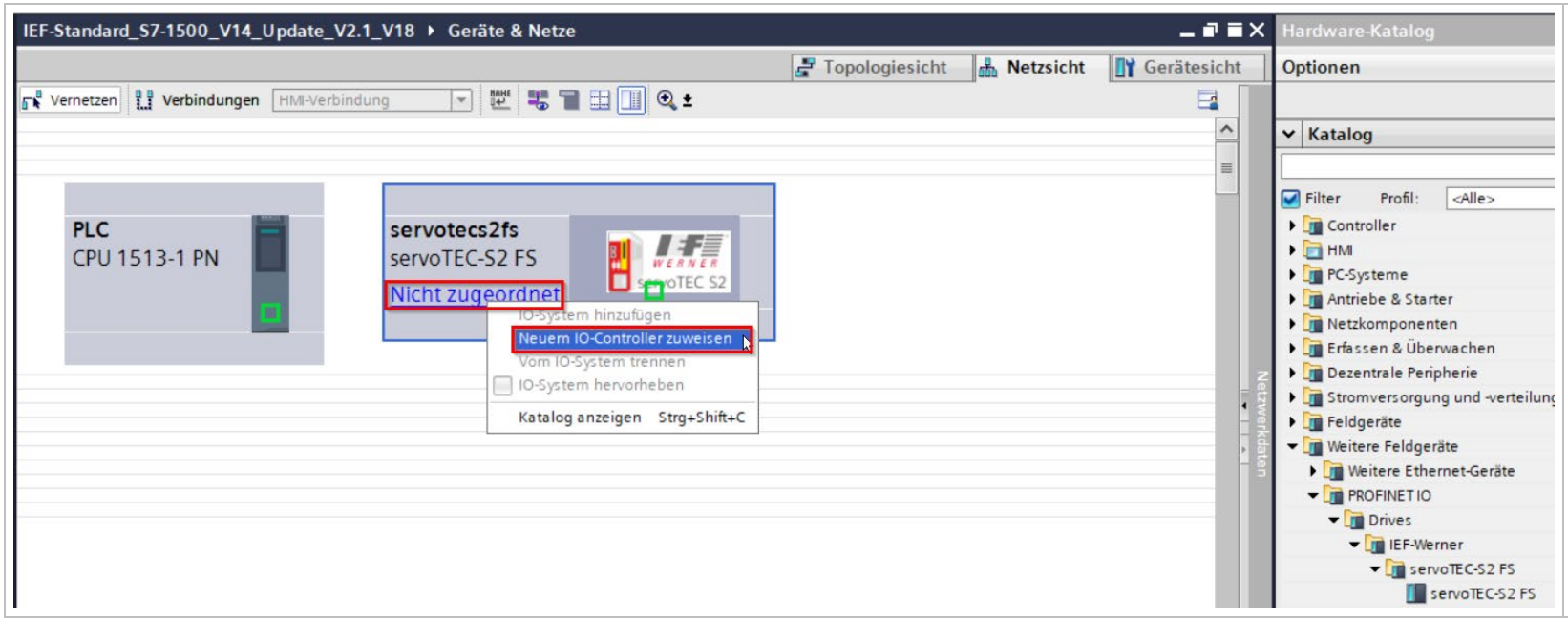
- Das Beispielprojekt finden Sie im Downloadbereich der IEF-Werner Website.
- Das Beispielprojekt muss einmalig geöffnet werden. Hierbei wird das Feldgerät Ihrem TIA Hardwarekatalog hinzugefügt.
- Sie können nun im Beispielprojekt die Achse testweise betreiben oder eine Integration in Ihr Projekt vornehmen.

4.2 Hardwarekonfiguration Schritt 1: Gerät anwählen



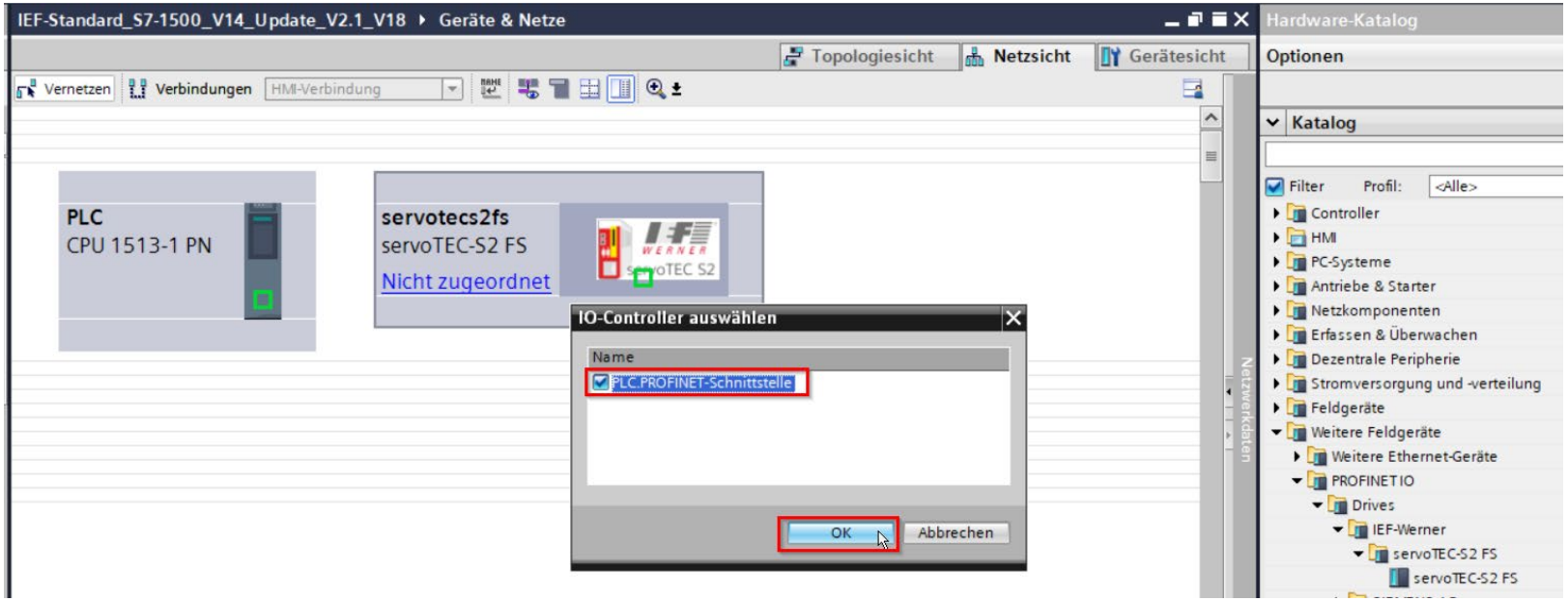
Das Feldgerät befindet sich im hier markierten Pfad des Hardware-katalogs. Es kann beliebig oft durch doppelklicken oder Drag & Drop der Netzsicht hinzugefügt werden.

4.3 Hardwarekonfiguration Schritt 2: IO-Controller zuweisen



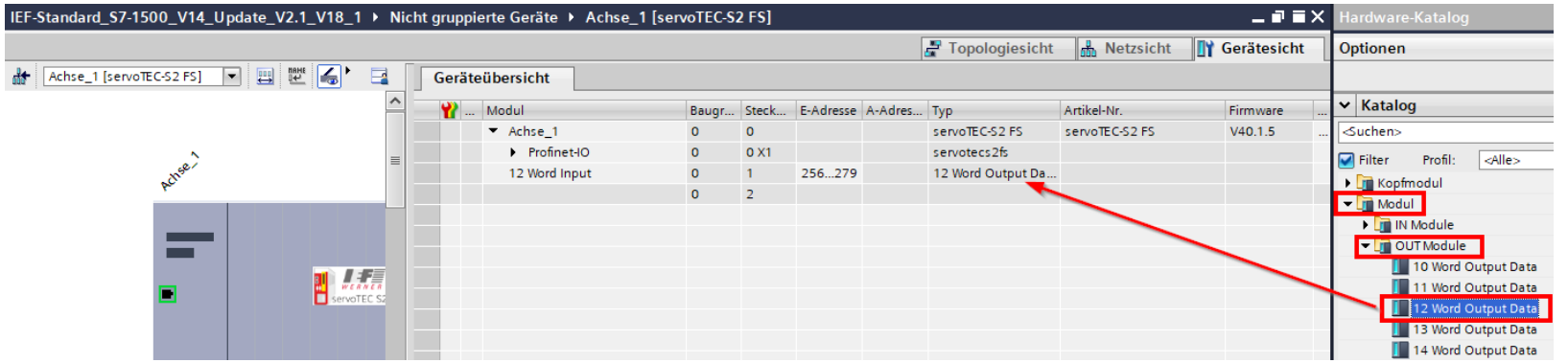
Mit einem Rechtsklick auf den blau markierten Text „Nicht zugeordnet“ öffnet sich das Menü mit der Option „Neuem IO-Controller zuweisen“.

4.4 Hardwarekonfiguration Schritt 3: IO-Controller auswählen



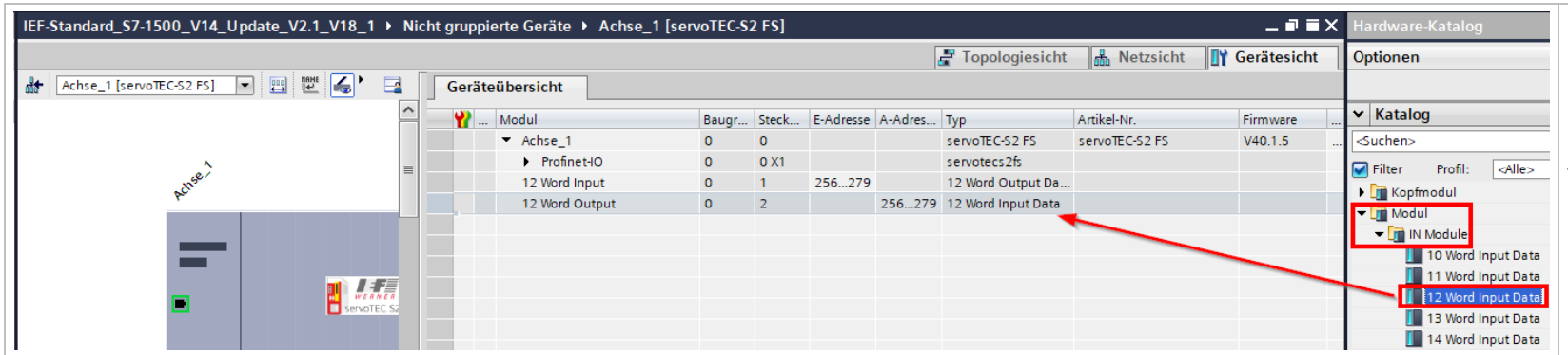
Die Schnittstelle auswählen und mit „OK“ bestätigen.

4.5 Hardwarekonfiguration Schritt 4: Zuweisung Input-Daten (SPS)



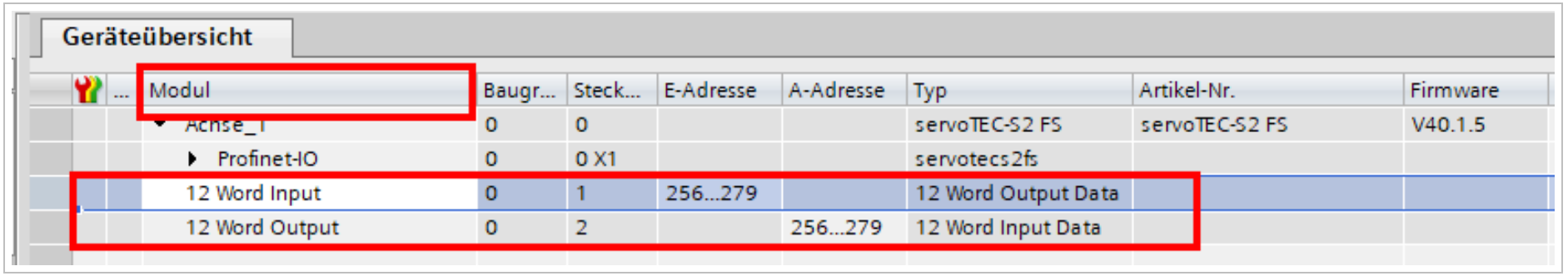
Die Bereiche können per Doppelklick oder Drag & Drop hinzugefügt werden.

4.6 Hardwarekonfiguration Schritt 5: Zuweisung Output-Daten (SPS)



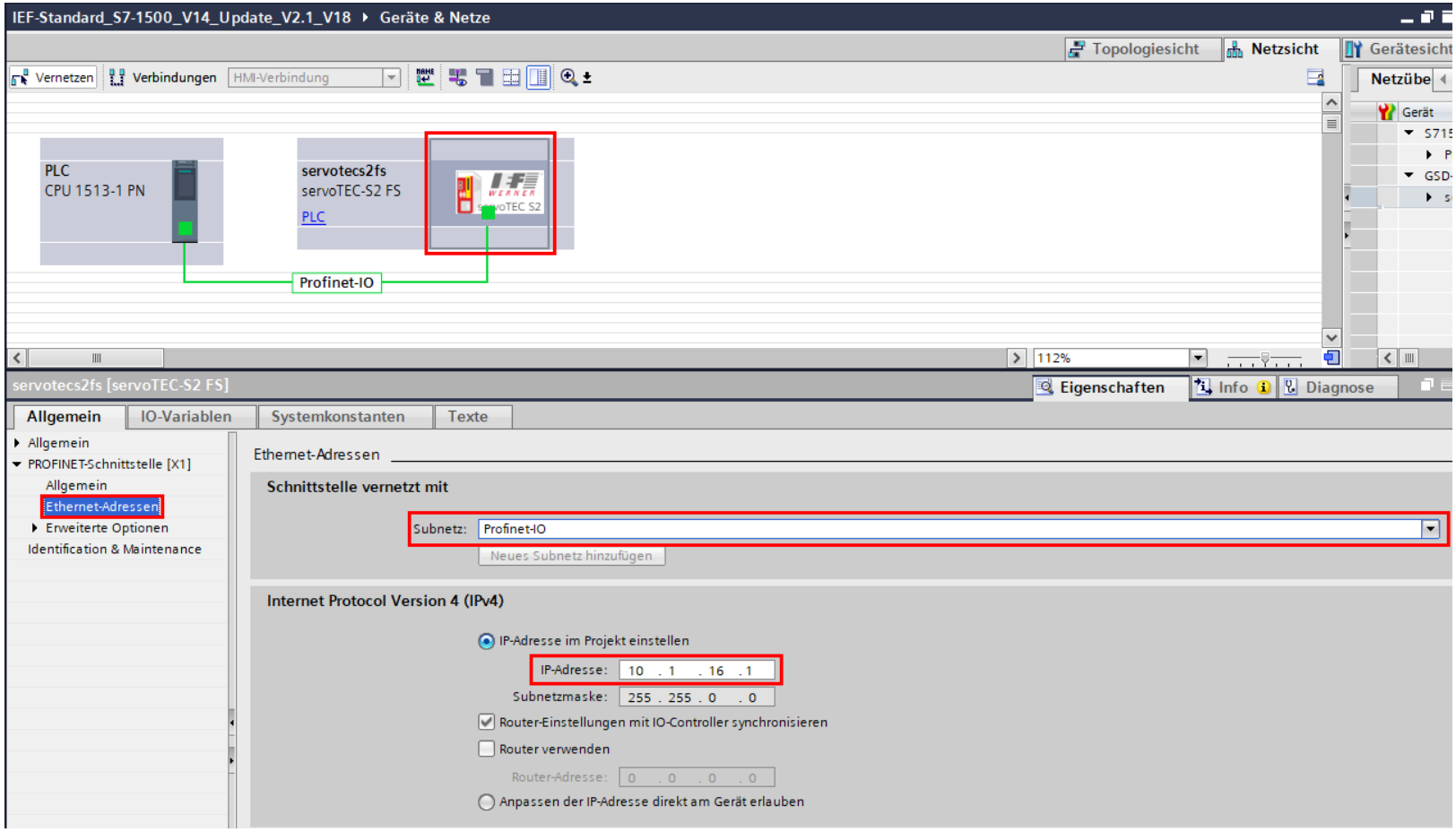
Die Bereiche können per Doppelklick oder Drag & Drop hinzugefügt werden.

4.7 Zuordnung der Daten kontrollieren



Bitte beachten Sie die korrekte Zuordnung der Input/Output Daten (hier im Bild dargestellt). Falls diese nicht korrekt ist, können Sie die Beschreibung im Bereich Module ändern, indem Sie den Text in „12 Word Input“ zu „12 Word Output“ ändern und umgekehrt. Danach sollte die Darstellung so wie auf dem Bild (oben) sein.

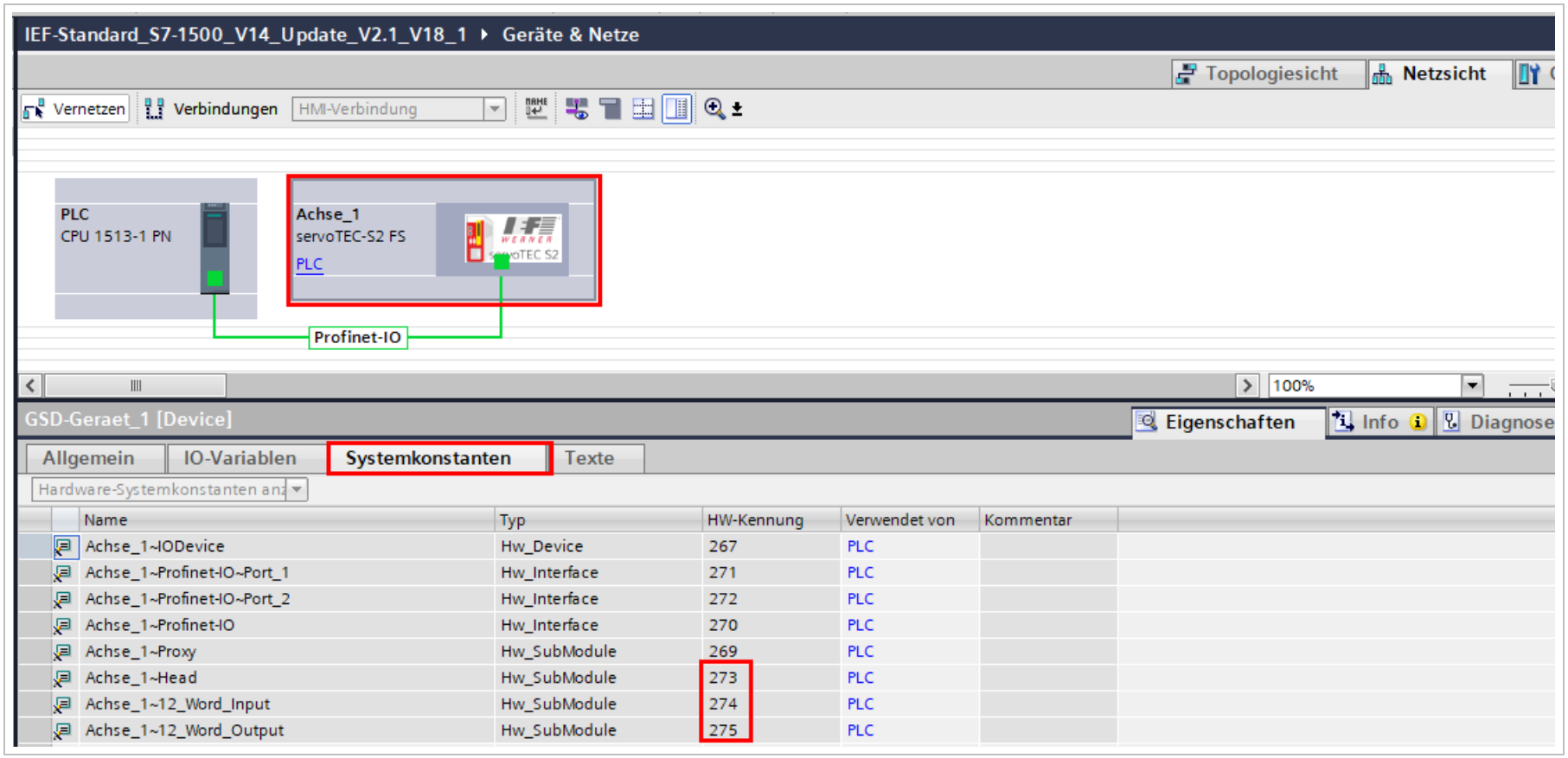
4.8 Hardwarekonfiguration Schritt 6: IP-Adresse und Subnetz einstellen



Die Standard IP-Adresse für den ersten Teilnehmer im Bus ist:

10.1.16.1.

4.9 Hardwarekonfiguration Schritt 7: HW-Kennungen prüfen

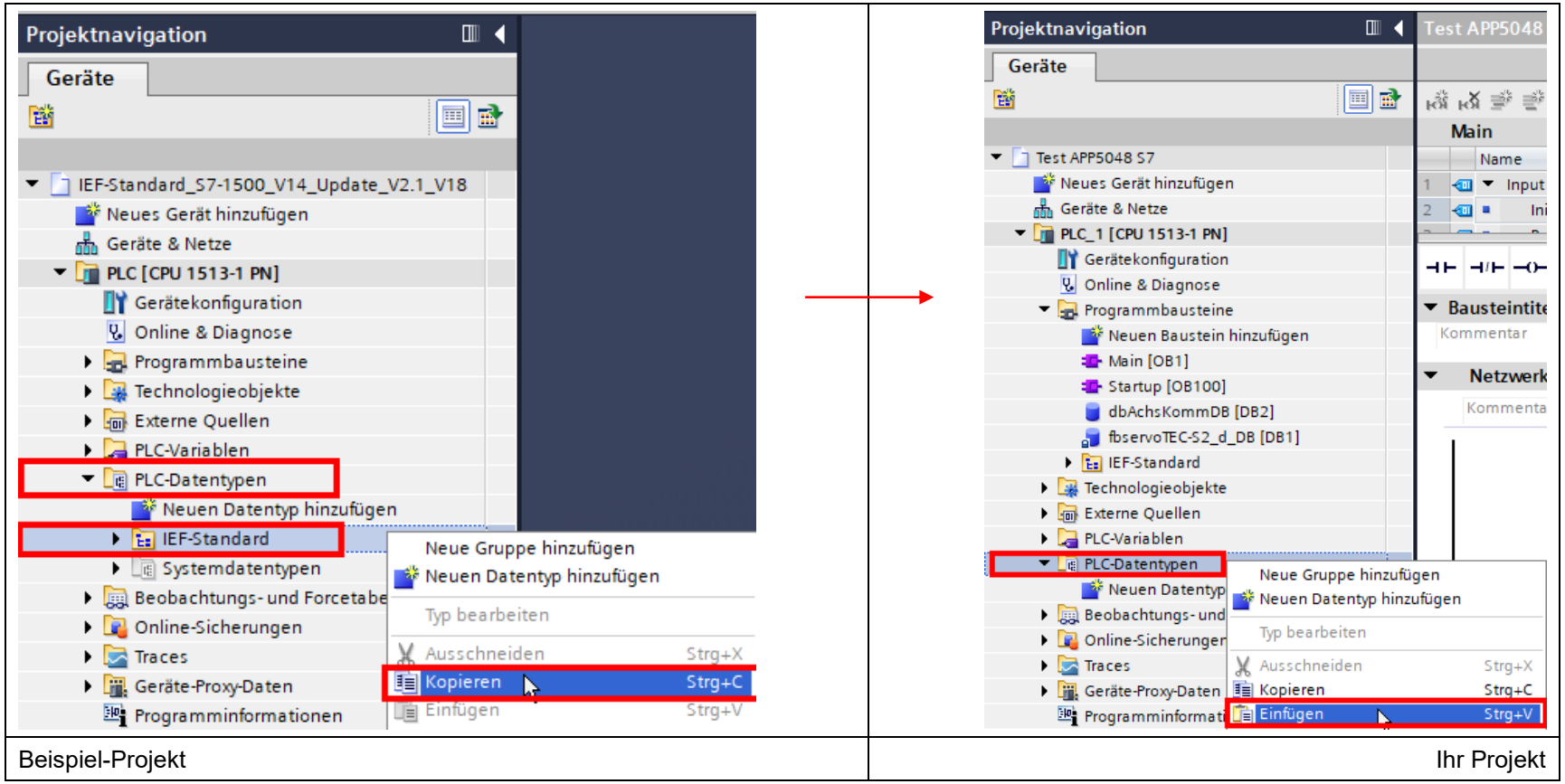


PRÜFEN	In der Hardwarekonfiguration muss die HW-Kennung des Geräts (Head, Input Daten und Output Daten) aufsteigend und direkt aufeinanderfolgend angegeben sein. Nur die HW-Kennung des Geräts (Head) wird an den Funktionsbaustein übergeben!
WENN NICHT, DANN	Falls die HW-Kennung nicht aufsteigend ist, kann Abhilfe geschaffen werden, indem man einen neuen servoTEC S2 FS aus den Feldgeräten hinzufügt und die Hardwarekonfiguration ab Schritt 1 wiederholt. Falls bei dem neu hinzugefügten Gerät die HW-Kennung korrekt ist, kann das alte Gerät aus der Netzsicht gelöscht werden. Falls dies nicht der Fall ist, nochmals ein neues Gerät hinzufügen.
HINWEIS	Die direkt aufeinanderfolgende HW-Kennung ist notwendig, damit die Parameter für die Referenzfahrt des servoTEC S2 FS über PROFINET richtig interpretiert werden. Leider erfolgt die Zuweisung automatisch durch TIA und kann nicht beeinflusst oder nachträglich geändert werden.

5 Mindestanforderungen für Integration in eigenes Projekt

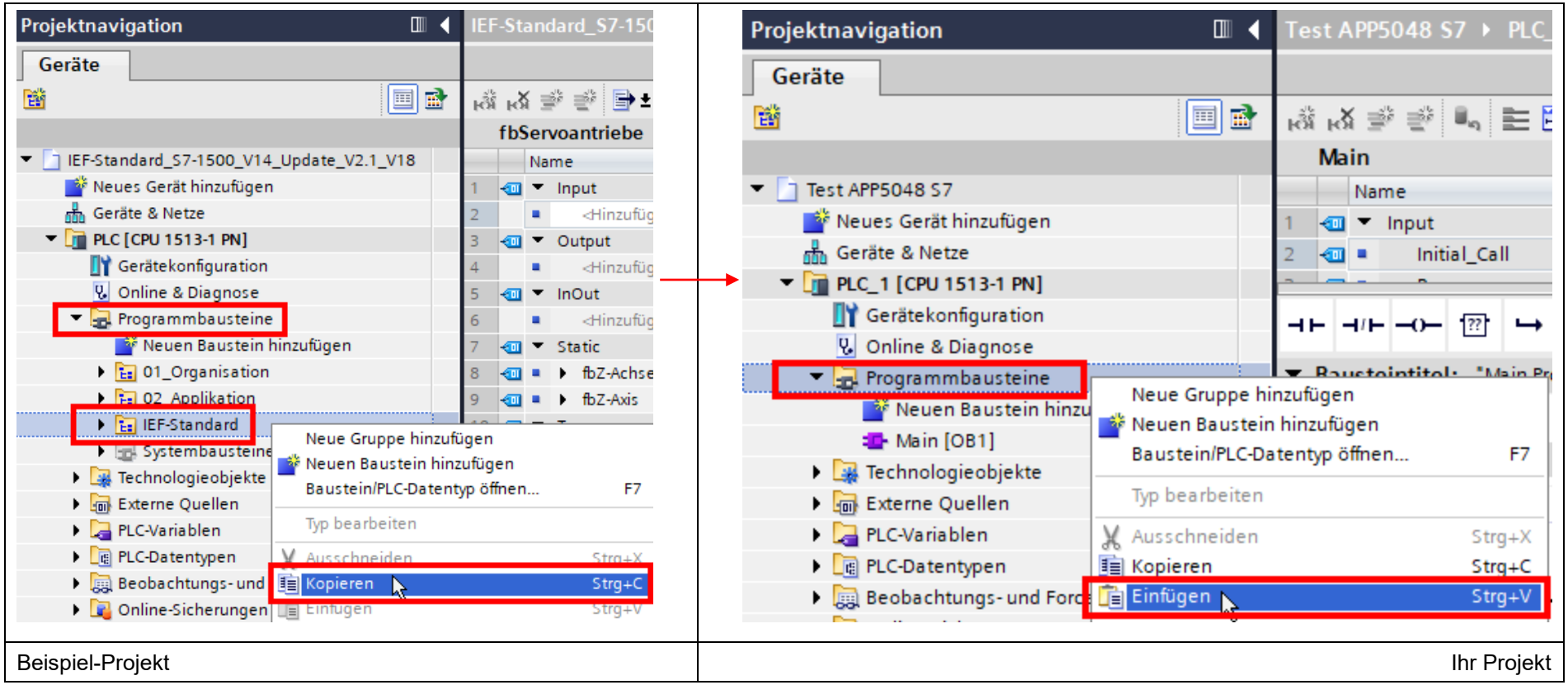
Für die Integration in Ihr Projekt müssen Sie die folgenden Aktionen durchführen:

1. Kopieren Sie aus unserem Beispielprojekt unter den „PLC-Datentypen“ den Ordner „IEF-Standard“ und fügen diesen Ihrem Projekt den PLC-Datentypen hinzu.



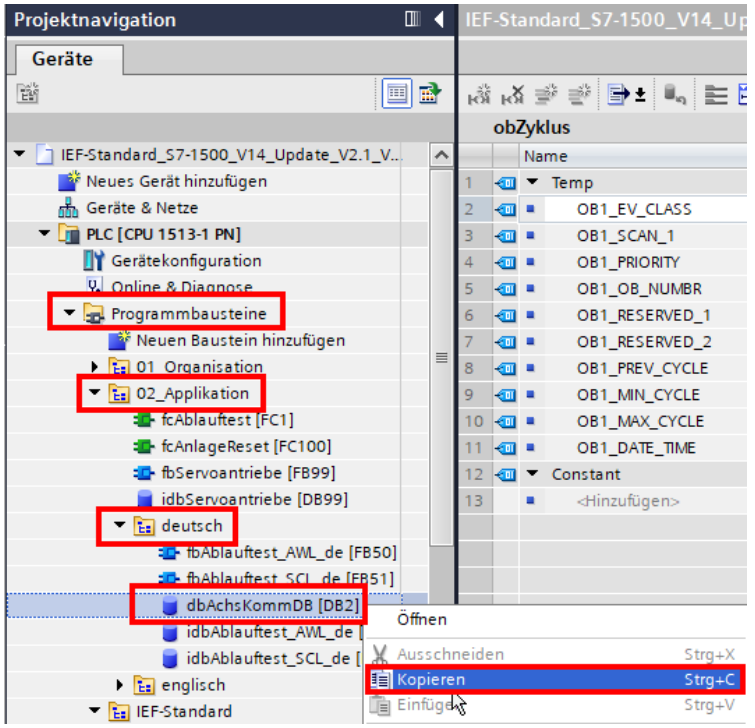
Beispiel-Projekt	Ihr Projekt

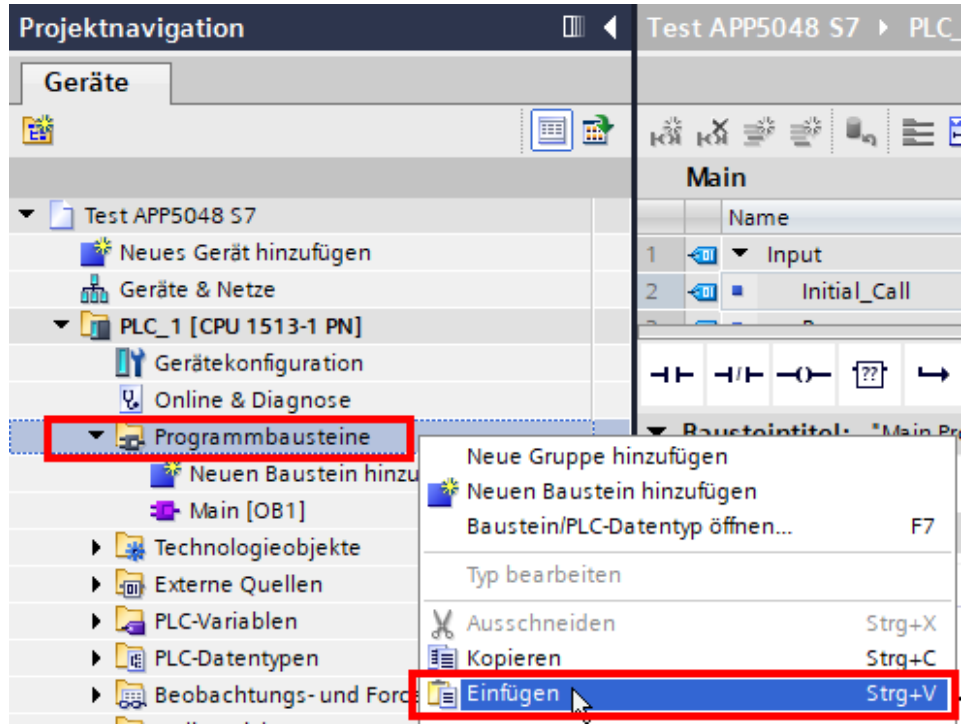
2. Kopieren Sie aus unserem Beispielprojekt unter „Programmbausteine“ den Ordner „IEF-Standard“ und fügen diesen Ihrem Projekt den Programmbausteinen hinzu.



Beispiel-Projekt	Ihr Projekt

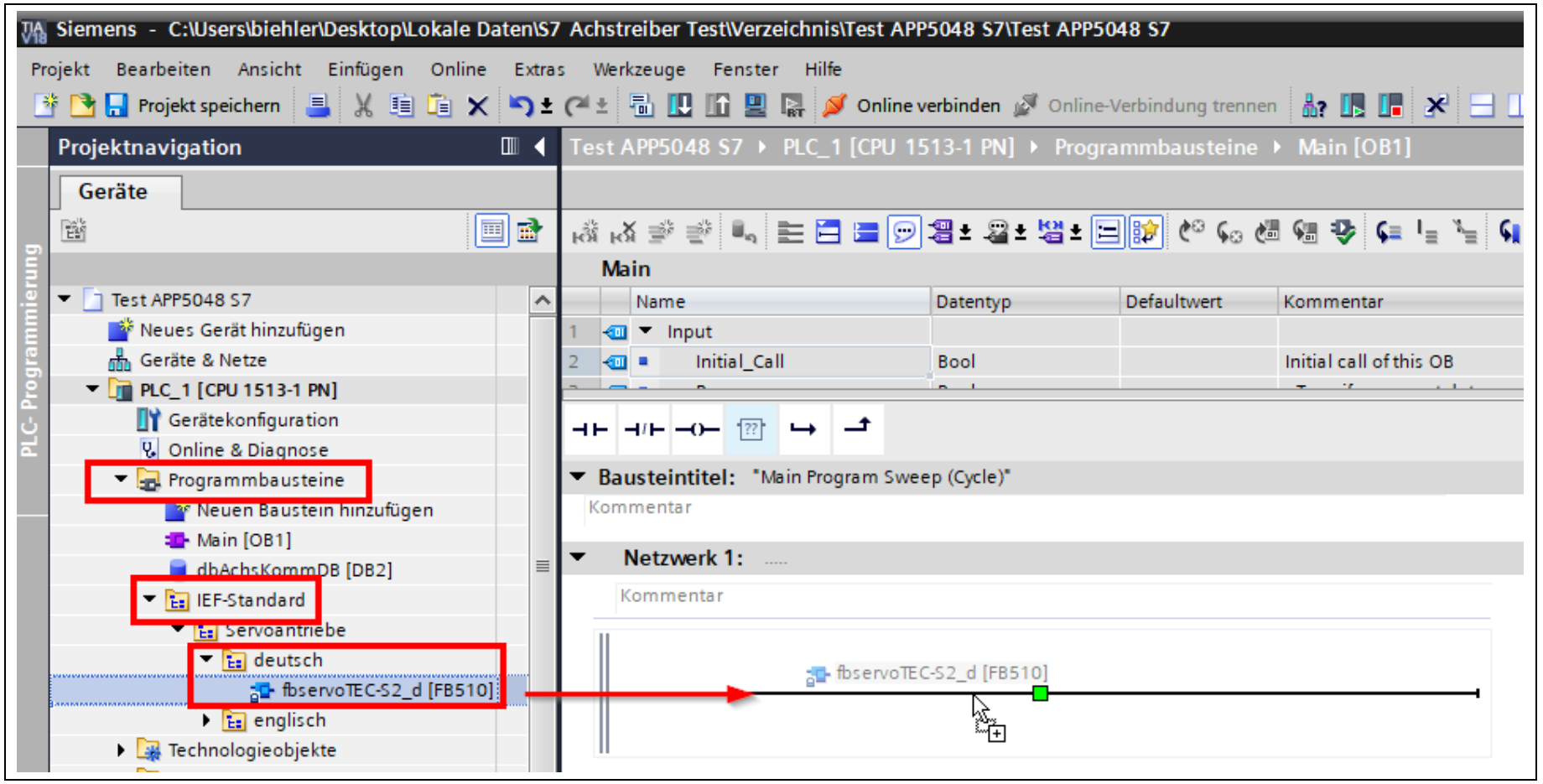
3. Kopieren Sie aus unserem Beispielprojekt unter „Programmbausteine“ → „02_Applikation“ → „deutsch“ den Datenbaustein „dbAchsKommDB [DB2]“ und fügen diesen Ihrem Projekt hinzu.



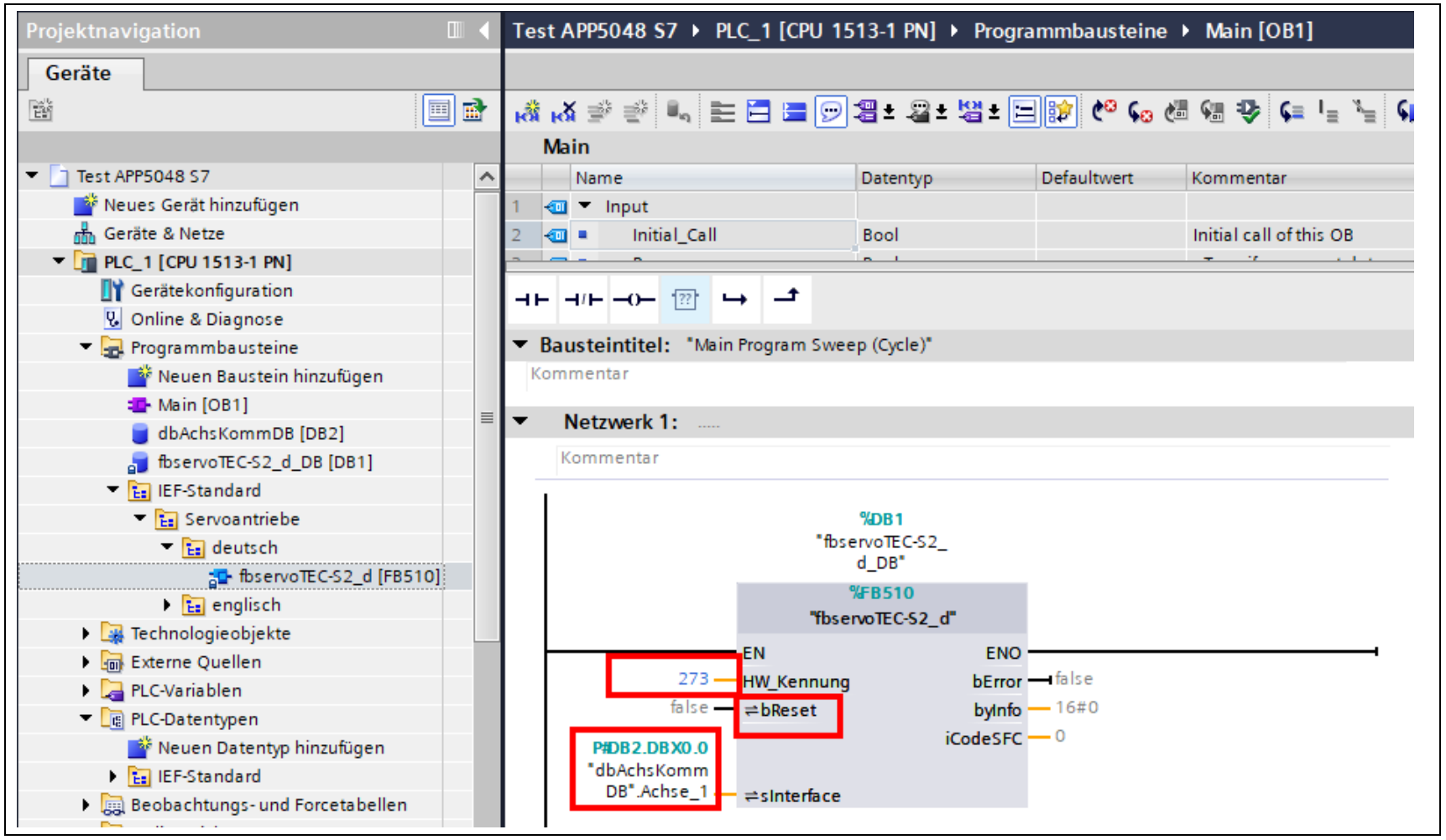


Beispiel-Projekt	Ihr Projekt

4. Fügen Sie den [FB510] Ihrem Main OB hinzu.



5. Beschalten Sie den Baustein mit der HW-Kennung (Head) und dem „dbAchsKommDB“ Achse_xx. Der Reset des Bausteins sollte bei jedem Startup der CPU durchgeführt werden. Deshalb sollte dieser im OB100 ausgeführt werden.



6 Datenschnittstelle zwischen SPS und servoTEC S2 (z. B. DB2)

Als Datenschnittstelle werden zwei Nahtstellenstrukturen verwendet:

- „Standartsteuerbereich (→ Achse wird angesteuert)“
- „Erweiterter Steuerbereich (→ Parameter im servoTEC-S2 FS werden eingestellt)“

Diese Signale (Daten) müssen bei Bedarf im Anwenderprogramm verknüpft werden.

6.1 Standartsteuerbereich (→ Achse wird angesteuert)

6.1.1 Kommandos an Achse

Benennung	Type	Funktion
AxisCmd.rZielPosition	Real	Zielposition für eine Absolutfahrt. → Kann während eines laufenden Fahrauftrages verändert werden!
AxisCmd.rDistanzWert	Real	Distanzwert für eine Relativfahrt. → Kann während eines laufenden Fahrauftrages verändert werden!
AxisCmd.rOverrideWert	Real	Dieser Parameter gibt die „**AxisCmd.rGeschwindigkeit**“ in Prozent vor. → Kann während eines laufenden Fahrauftrages verändert werden!
AxisCmd.rGeschwindigkeit	Real	Achsgeschwindigkeit → wird von **„AxisCmd.rOverrideWert“** beeinflusst. Geschwindigkeit für einen absoluten oder relativen Auftrag.
AxisCmd.rRefGeschwindigkeit	Real	Referenzgeschwindigkeit → Suchlauf des Schalters (search for switch)
AxisCmd.rJogGeschwindigkeit	Real	Tippgeschwindigkeit → JOG-Betrieb!
AxisCmd.rBeschleunigung	Real	Achsbeschleunigung → „**Startrampe**“
AxisCmd.rVerzoegerung	Real	Achsverzögerung → „**Bremsrampe**“
AxisCmd.bFreigabe	Bool	Achsfreigabe von Applikation → Lastversorgung für Servoantrieb eingeschaltet!

Benennung	Type	Funktion
AxisCmd.bStartInit	Bool	Die Achse wird initialisiert → Vorgabe Betriebsart „**Positionieren**“! Nach der Initialisierung wird das Flag „**AxisStatus.bInitFlag**“ gesetzt und das Kommando „**AxisCmd.bStartInit**“ als Quittung zurückgesetzt. Wenn die Achse bereits einen gültigen Referenzpunkt hat, wird das Flag „**AxisStatus.bRefFlag**“ ebenfalls gesetzt. Eine Initialisierung wird sofort abgebrochen, wenn das Kommando „**AxisCmd.bStartInit**“ zurückgesetzt wird.
AxisCmd.bStartHoming	Bool	Die Achse führt eine Referenzfahrt aus. → Suchen des Schalters mit „**AxisCmd.rRefGeschwindigkeit**“ → Runterfahren von Schalter mit „**ServoCmd.rRefSchleichSpeed**“ → Nullpunktverschiebung um „**ServoCmd.rReferenzOffset**“ speichern Nach der Referenzfahrt wird das Flag „**AxisStatus.bRefFlag**“ gesetzt und das Kommando „**AxisCmd.bStartHoming**“ als Quittung zurückgesetzt. Eine Referenzfahrt wird sofort abgebrochen, wenn das Kommando „**AxisCmd.bStartHoming**“ zurückgesetzt wird. Eine Referenzfahrt mit einem Absolutwertgeber kann nur ausgeführt werden, wenn zusätzlich das Kommando „**ServoCmd.bABS_NeuLernen**“ gesetzt ist. Nach der Referenzfahrt wird das Flag „**AxisStatus.bJustFlag**“ gesetzt und „**ServoCmd.bABS_NeuLernen**“ als Quittung zurückgesetzt.
AxisCmd.bStartSetPos	Bool	Die Achse wird auf die Referenzposition gesetzt. → Absolute Position auf „**ServoCmd. bReferenzPosition**“ speichern Nach dem Setzen der Referenzposition werden die beiden Flags „**AxisStatus.bRefFlag**“ und „**AxisStatus.bSetFlag**“ gesetzt und das Kommando „**AxisCmd.bStartSetPos**“ als Quittung zurückgesetzt. Das Referenz setzen wird sofort abgebrochen, wenn das Kommando „**AxisCmd.bStartSetPos**“ zurückgesetzt wird. Ein Setzen der Referenz mit einem Absolutwertgeber kann nur ausgeführt werden, wenn zusätzlich das Kommando „**ServoCmd.bABS_NeuLernen**“ gesetzt ist. Nach Referenz setzen wird das Flag „**AxisStatus.bJustFlag**“ gesetzt und „**ServoCmd.bABS_NeuLernen**“ als Quittung zurückgesetzt.

Benennung	Type	Funktion
AxisCmd.bStartAbs	Bool	Die Achse wird mit der „**AxisCmd.rGeschwindigkeit**“ absolut in die „**AxisCmd.rZielPosition**" gestartet. Hat die Achse die Zielposition erreicht, wird das Flag „**AxisStatus.bAbsInPos**“ gesetzt und das Kommando „**AxisCmd.bStartAbs**“ als Quittung zurückgesetzt. Der gestartete Fahrauftrag wird sofort abgebrochen und verworfen, wenn das Kommando „**AxisCmd.bStartAbs“** zurückgesetzt wird.
AxisCmd.bStartRelPos	Bool	Die Achse wird mit „**AxisCmd.rGeschwindigkeit**“ um den „**AxisCmd.rDistanzWert**“ in positive Richtung gestartet. Wurde um den Distanzwert verfahren, wird das Flag „**AxisStatus.bRelInPos**“ gesetzt und das Kommando „**AxisCmd.bStartRelPos**“ als Quittung zurückgesetzt. Der gestartete Fahrauftrag wird sofort abgebrochen und verworfen, wenn das Kommando „**AxisCmd.bStartRelPos**“ zurückgesetzt wird.
AxisCmd.bStartRelNeg	Bool	Die Achse wird mit „**AxisCmd.rGeschwindigkeit**“ um den „**AxisCmd.rDistanzWert**“ in negative Richtung gestartet. Wurde um den Distanzwert verfahren, wird das Flag „**AxisStatus.bRelInPos**“ gesetzt und das Kommando „**AxisCmd.bStartRelNeg**“ als Quittung zurückgesetzt. Der gestartete Fahrauftrag wird sofort abgebrochen und verworfen, wenn das Kommando „**AxisCmd.bStartRelNeg**“ zurückgesetzt wird.
AxisCmd.bStartJogPos	Bool	Die Achse fährt mit „**AxisCmd.rJogGeschwindigkeit**“ in positive Richtung, solange das Kommando „**AxisCmd.bStartJogPos**“ gesetzt ist. → Der Override wird intern mit 100% vorgegeben!
AxisCmd.bStartJogNeg	Bool	Die Achse fährt mit „**AxisCmd.rJogGeschwindigkeit**“ in negative Richtung, solange das Kommando „**AxisCmd.bStartJogNeg**“ gesetzt ist. → Der Override wird intern mit 100% vorgegeben!
AxisCmd.bStoppAchse	Bool	Die Achse wird gestoppt → Fahrauftrag wird unterbrochen. Wird während eines Fahrauftrages das Kommando „**AxisCmd.bStoppAchse**“ ausgegeben, wird die Achse sofort gestoppt. Bei Rücknahme des Kommandos wird der unterbrochene Fahrauftrag fortgesetzt.

Benennung	Type	Funktion
AxisCmd.bQuittierung	Bool	Fehler an Servocontroller zurücksetzen

6.1.2 Statusmeldungen von Achse

Benennung	Type	Funktion
AxisStatus.rIstPosition	Real	Aktuelle Achsposition
AxisStatus.bBereit	Bool	Gesetzt, wenn Achse betriebsbereit ist → Servoregler ist freigegeben!
AxisStatus.bInitFlag	Bool	Gesetzt, wenn Achse initialisiert ist → Betriebsart „**Positionieren**“ aktiv!
AxisStatus.bJustFlag	Bool	Gesetzt, wenn Aktion „**Referenzfahrt**“ bzw. „**Referenz setzen**“ ausgeführt wurde → Flag ist nur bei einem **Absolutwertgeber** relevant!
AxisStatus.bRefFlag	Bool	Gesetzt, wenn Achse referenziert ist → nach „**Referenzfahrt**“ bzw. „**Referenz setzen**“
AxisStatus.bSetFlag	Bool	Gesetzt, wenn Aktion „**Referenz setzen**“ ausgeführt wurde
AxisStatus.bAbsInPos	Bool	Gesetzt, wenn Achse nach einem absoluten Fahrauftrag in Position steht. → Zielvorgabe: „**AxisCmd.rZielPosition**“, → Startfreigabe: „**AxisCmd.bStartAbs**"
AxisStatus.bRelInPos	Bool	Gesetzt, wenn Achse nach einem relativen Fahrauftrag in Position steht. → Zielvorgabe: „**AxisCmd.rDistanzWert**“, → Startfreigabe: „**AxisCmd.bStartRelPos**“ oder „**AxisCmd.bStartRelNeg**“
AxisStatus.bBusy	Bool	Achse führt eine Aktion aus → Treiberablauf ist aktiv!
AxisStatus.bFehler	Bool	Gesetzt, wenn am Controller ein Fehler ansteht
AxisStatus.bWarnung	Bool	Gesetzt, wenn am Controller eine Warnung ansteht

6.2 Erweiterter Steuerbereich (→ Parameter im servoTEC S2 FS werden eingestellt)

6.2.1 Kommandos an Controller

Benennung	Type	Funktion
ServoCmd.rReferenzOffset	Real	Die Achse wird bei einer „**Referenzfahrt**“ um diesen Offset zum Bezugspunkt verschoben. → Vorgabe ReferenzOffset: Nullpunkt minus Bezugspunkt (Endschalter, Anschlag)
ServoCmd.rReferenzPosition	Real	Die Achse wird der Sequenz „**Referenz setzen**“ auf diese absolute Position gesetzt.
ServoCmd.rReduzierterStrom	Real	Motorstrom für die Drehmomentreduzierung → Vorgabe in [A]. Das Drehmoment verhält sich proportional zum Motorstrom. Bei zu geringem Motorstrom ist kein Verfahren der Achse mehr möglich.
ServoCmd.rRefSchleichSpeed	Real	Referenzschleichgeschwindigkeit → Runterfahren von Schalter (search for Zero)
ServoCmd.rRefBeschleunigung	Real	Beschleunigung bei Referenzfahrt (→ Startrampe / Bremsrampe)
ServoCmd.rJogBeschleunigung	Real	Beschleunigung bei JOG-Fahrt (→ Startrampe / Bremsrampe)
ServoCmd.rVerfahrBereichMin	Real	Vorgabe minimale Verfahrbereichsgrenze der Achse
ServoCmd.rVerfahrBereichMax	Real	Vorgabe maximale Verfahrbereichsgrenze der Achse
ServoCmd.rRuckwertVorgabe	Real	Vorgabe Ruckwert für Begrenzung der Rampen (→ Startrampe / Bremsrampe)
ServoCmd.iReferenzMethode	Int	Vorgabe der Referenzfahrtmethode -1 / -2 = Anschlag negativ / positiv mit Nullimpuls, 1 / 2 = Endschalter negativ / positiv mit Nullimpuls, 7 / 11 = Referenzschalter negativ / positiv mit Nullimpuls, -17 / -18 = Anschlag negativ / positiv, 17 / 18 = Endschalter negativ / positiv, 23 / 27 = Referenzschalter positiv / negativ, 34 = aktuelle Position → siehe „MAN_DE_1078649_LV-servoTEC_S2_Softwarehandbuch.pdf“

Benennung	Type	Funktion
ServoCmd.iModusRundachse	Int	Fahrmodus für Rundachse 0 = inaktiv, 1 = kürzester Weg (Modulo), 2 = Drehrichtung aus Positionssatz, 3 = Feste Drehrichtung positiv, 4 = Feste Drehrichtung negativ
ServoCmd.bAbsolutgeber	Bool	True: Motor mit Absolutwertgeber (→ **EnDat , Hiperface!**) False: Motor ohne Absolutwertgeber (→ **Resolver!**)
ServoCmd.bABS_NeuLernen	Bool	True: Absolutwertgeber neu einlernen False: keine Bedeutung Um ein Absolutwertgeber neu einzulernen, ist vor dem Start der Aktionen „**Referenzfahrt**“ bzw. „**Referenz setzen**“ dieses Flag zu setzen. Nach Beendigung der Aktionen wird der berechnete Positionsoffset auf dem Geber-EEPROM gespeichert und das Freigabe-Flag zurückgesetzt.
ServoCmd.bRuckBegrenzen	Bool	True: mit Ruckbegrenzung → berechnete Rampe (sin²) False: keine Ruckbegrenzung → lineare Rampe
ServoCmd.bMomentReduzieren	Bool	True: Momentreduzierung aktiv (→ über Stromvorgabe!) False: keine Momentreduzierung
ServoCmd.bOhneSchleppgrenze	Bool	True: Schleppfehlergrenze deaktivieren False: Schleppfehlergrenze aktivieren
ServoCmd.bOhneSWGrenzen	Bool	True: Softwareendschalter deaktivieren False: Softwareendschalter aktivieren
ServoCmd.bSichererHalt	Bool	True: Anforderung „Sicherer HALT“ False: kein „Sicherer HALT“ angefordert

6.2.2 Statusmeldungen von Controller

Benennung	Type	Funktion
ServoStatus.rIstgeschwindigkeit	Real	Aktuelle Geschwindigkeit
ServoStatus.rIstMotorstrom	Real	Aktueller Motorstrom → Bezug Nennstrom!
ServoStatus.uiFehlerNummer	UInt	Fehlernummer [Format: **xxy**] → Hauptindex=**xx**, Unterindex=**y**
ServoStatus.uiWarnungNummer	UInt	Warnungnummer [Format: **xxy**] → Hauptindex=**xx**, Unterindex=**y**
ServoStatus.bRuckBegrenzt	Bool	True: Ruckbegrenzung aktiv (→ Rampe berechnet [sin²]!) False: Ruckbegrenzung nicht aktiv (→ Rampe linear!)
ServoStatus.bMomentReduziert	Bool	True: Drehmomentreduzierung aktiv False: Drehmomentreduzierung nicht aktiv
ServoStatus.bSchleppgrenzeAktiv	Bool	True: Schleppfehlergrenze aktiv False: Schleppfehlergrenze nicht aktiv
ServoStatus.bSWGrenzenAktiv	Bool	True: Softwareendschalter aktiv False: Softwareendschalter nicht aktiv
ServoStatus.bAntriebSicher	Bool	True: „Sicherer HALT“ aktiv False: „Sicherer HALT“ nicht aktiv

6.3 Beispiel Achskommunikation [DB2] → Linearachse mit Absolutgeber

HINWEIS Verzögerungs- / Beschleunigungswerte, max. Geschwindigkeit und empfohlener Ruck des Antriebs sind i.d.R. im kaufmännischen Angebot des Antriebs angegeben. Weitere Informationen erhalten Sie über unseren Kundensupport.

Benennung	Type	Anfangswert	Aktualwert	Zugriff	Funktion
Achse_1.AxisCmd.rZielPosition	Real	0.0	**150.0**	W	Zielposition für Absolutfahrt [in LU]
Achse_1.AxisCmd.rDistanzWert	Real	0.0	**50.0**	W	Distanzwert für Relativfahrt [in LU]
Achse_1.AxisCmd.rOverrideWert	Real	0.0	**100.0**	W	Override-Wert der Achse (0..100%)
Achse_1.AxisCmd.rGeschwindigkeit	Real	0.0	**300.0**	W	Fahrgeschwindigkeit [in LU/s]
Achse_1.AxisCmd.rRefGeschwindigkeit	Real	0.0	**5.0**	W	Referenzgeschwindigkeit [in LU/s]
Achse_1.AxisCmd.rJogGeschwindigkeit	Real	0.0	**30.0**	W	Tippgeschwindigkeit [in LU/s]
Achse_1.AxisCmd.rBeschleunigung	Real	0.0	**3000.0**	W	Beschleunigung [in LU/s²]
Achse_1.AxisCmd.rVerzoegerung	Real	0.0	**3000.0**	W	Verzögerung [in LU/s²]
Achse_1.AxisCmd.bFreigabe	Bool	FALSE	**TRUE**	W	Freigabe von Applikation
Achse_1.AxisCmd.bStartInit	Bool	FALSE	FALSE	RW	Start Initialisierung
Achse_1.AxisCmd.bStartHoming	Bool	FALSE	FALSE	RW	Start Referenzfahrt
Achse_1.AxisCmd.bStartSetPos	Bool	FALSE	FALSE	RW	Start Referenz setzen
Achse_1.AxisCmd.bStartAbs	Bool	FALSE	FALSE	RW	Start Absolutfahrt
Achse_1.AxisCmd.bStartRelPos	Bool	FALSE	FALSE	RW	Start Relativfahrt positiv
Achse_1.AxisCmd.bStartRelNeg	Bool	FALSE	FALSE	RW	Start Relativfahrt negativ
Achse_1.AxisCmd.bStartJogPos	Bool	FALSE	FALSE	RW	Start Tippen positiv (JOG!)
Achse_1.AxisCmd.bStartJogNeg	Bool	FALSE	FALSE	RW	Start Tippen negativ (JOG!)
Achse_1.AxisCmd.bStoppAchse	Bool	FALSE	FALSE	W	Stopp Positionierung
Achse_1.AxisCmd.bQuittierung	Bool	FALSE	FALSE	W	Fehler zurücksetzen
Achse_1.AxisStatus.rIstPosition	Real	0.0	**150.0**	R	Istposition der Achse [in LU]
Achse_1.AxisStatus.bBereit	Bool	FALSE	**TRUE**	R	Achse ist betriebsbereit
Achse_1.AxisStatus.bInitFlag	Bool	FALSE	**TRUE**	R	Achse ist initialisiert

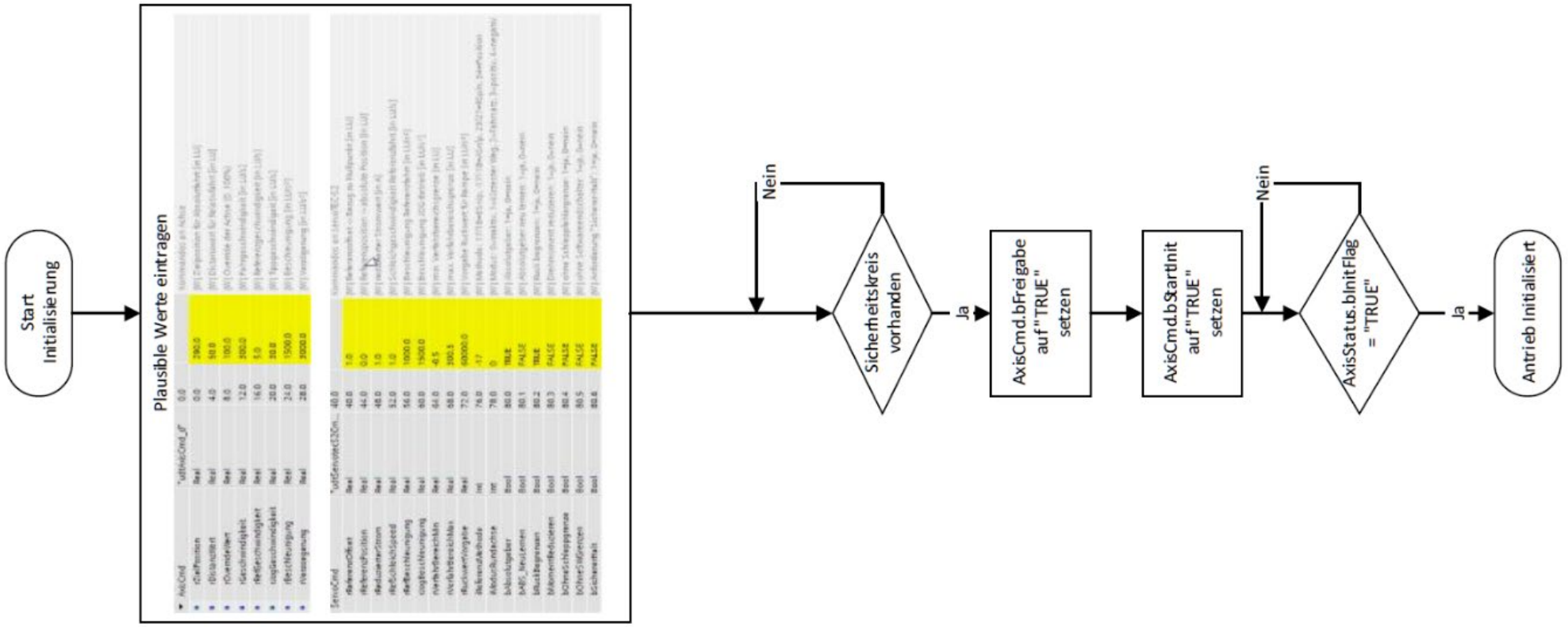
Benennung	Type	Anfangswert	Aktualwert	Zugriff	Funktion
Achse_1.AxisStatus.bJustFlag	Bool	FALSE	**TRUE**	R	Geber-Justage beendet
Achse_1.AxisStatus.bRefFlag	Bool	FALSE	**TRUE**	R	Achse ist referenziert
Achse_1.AxisStatus.bSetFlag	Bool	FALSE	FALSE	R	Referenz wurde gesetzt
Achse_1.AxisStatus.bAbsInPos	Bool	FALSE	**TRUE**	R	Achse "absolut" in Position
Achse_1.AxisStatus.bRelInPos	Bool	FALSE	FALSE	R	Achse "relativ" in Position
Achse_1.AxisStatus.bBusy	Bool	FALSE	FALSE	R	Achse bearbeitet Auftrag
Achse_1.AxisStatus.bFehler	Bool	FALSE	FALSE	R	Sammelmeldung Fehler
Achse_1.AxisStatus.bWarnung	Bool	FALSE	FALSE	R	Sammelmeldung Warnung
Achse_1.ServoCmd.rReferenzOffset	Real	0.0	**1.0**	W	Positionsoffset bei Referenzfahrt [in LU]
Achse_1.ServoCmd.rReferenzPosition	Real	0.0	**0.0**	W	Absolutposition bei Referenz setzen [in LU]
Achse_1.ServoCmd.rReduzierterStrom	Real	0.0	**1.0**	W	Stromwert bei Momentreduzierung [in A]
Achse_1.ServoCmd.rRefSchleichSpeed	Real	0.0	**1.0**	W	Schleichspeed bei Referenzfahrt [in LU/s]
Achse_1.ServoCmd.rRefBeschleunigung	Real	0.0	**1000.0**	W	Beschleunigung bei Referenzfahrt [in LU/s²]
Achse_1.ServoCmd.rJogBeschleunigung	Real	0.0	**1500.0**	W	Beschleunigung bei JOG-Fahrt [in LU/s²]
Achse_1.ServoCmd.rVerfahrBereichMin	Real	0.0	**-0.5**	W	min. Verfahrbereichsgrenze [in LU]
Achse_1.ServoCmd.rVerfahrBereichMax	Real	0.0	**300.5**	W	max. Verfahrbereichsgrenze [in LU]
Achse_1.ServoCmd.rVorgabeRuckwert	Real	0.0	**60000.0**	W	Vorgabe Ruckwert für Rampe [in LU/s³]
Achse_1.ServoCmd.iReferenzMethode	Int	0	**-17**	W	Referenzmethode → -17 = Anschlag negativ
Achse_1.ServoCmd.iModusRundachse	Int	0	**0**	W	Modus Rundachse → 0 = inaktiv
Achse_1.ServoCmd.bAbsolutgeber	Bool	FALSE	**TRUE**	W	Achse mit Absolutwertgeber
Achse_1.ServoCmd.bABS_NeuLernen	Bool	FALSE	FALSE	W	Absolutwertgeber neu einlernen

Benennung	Type	Anfangswert	Aktualwert	Zugriff	Funktion
Achse_1.ServoCmd.bRuckBegrenzen	Bool	FALSE	**TRUE**	W	Ruck begrenzen
Achse_1.ServoCmd.bMomentReduzieren	Bool	FALSE	FALSE	W	Drehmoment reduzieren
Achse_1.ServoCmd.bOhneSchleppgrenze	Bool	FALSE	FALSE	W	Schleppfehlergrenze deaktivieren
Achse_1.ServoCmd.bOhneSWGrenzen	Bool	FALSE	FALSE	W	Softwareendschalter deaktivieren
Achse_1.ServoCmd.bSichererHalt	Bool	FALSE	FALSE	W	Anforderung "Sicherer HALT"
Achse_1.ServoStatus.rIstGeschwindigkeit	Real	0.0	0.0	R	Aktuelle Geschwindigkeit [in LU/s]
Achse_1.ServoStatus.rIstMotorstrom	Real	0.0	0.0	R	Aktueller Motorstrom [→ Bezug INenn]
Achse_1.ServoStatus.uiFehlerNummer	UInt	0	0	R	Fehlernummer (HF:00..96, UF:0..9)
Achse_1.ServoStatus.uiWarungNummer	UInt	0	0	R	Warnungnummer (HF:00..96, UF:0..9)
Achse_1.ServoStatus.bRuckBegrenzt	Bool	FALSE	**TRUE**	R	Ruckbegrenzung aktiv (→ Rampe berechnet!)
Achse_1.ServoStatus.bMomentReduziert	Bool	FALSE	FALSE	R	Drehmoment reduziert
Achse_1.ServoStatus.bSchleppgrenzeAktiv	Bool	FALSE	**TRUE**	R	Schleppfehlergrenze aktiv
Achse_1.ServoStatus.bSWGrenzenAktiv	Bool	FALSE	**TRUE**	R	Softwareendschalter aktiv
Achse_1.ServoStatus.bAntriebSicher	Bool	FALSE	FALSE	R	"Sicherer HALT" aktiv

7 Ablaufdiagramme

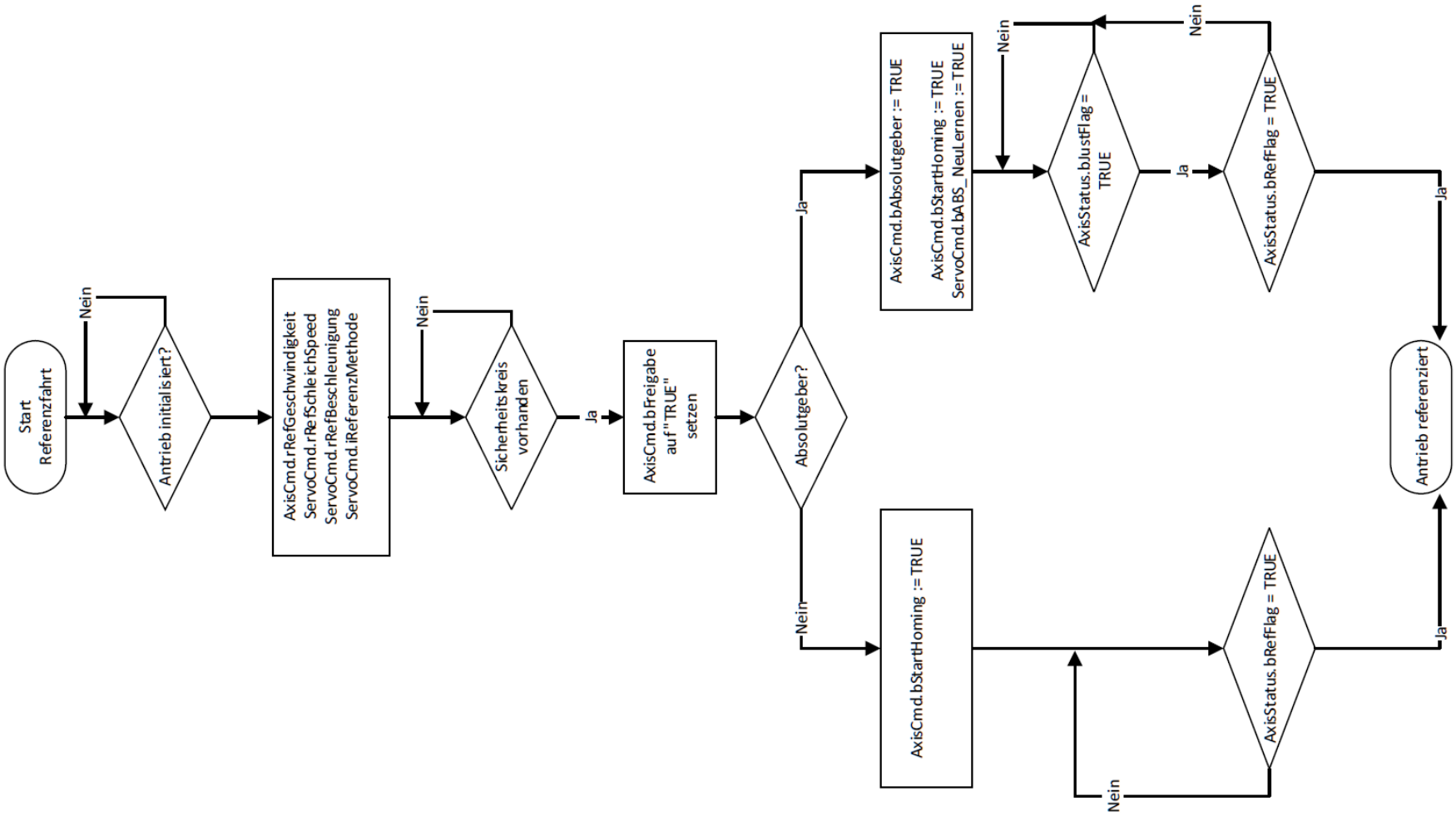
7.1 Initialisierung des Antriebs

Start / Ende	Entscheidung	Prozess

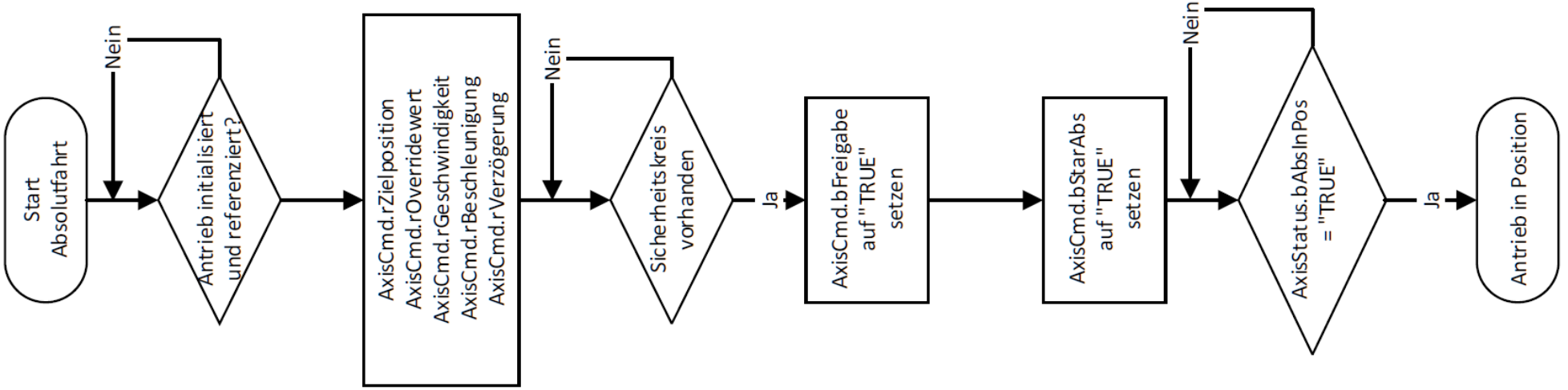


7.2 Referenzieren des Antriebs

HINWEIS Die Referenzfahrt wird bei einem Absolutgeber in der Regel nur einmalig durchgeführt. Nach der erfolgreich durchgeführter Referenzfahrt setzt sich das Flag „Referenz gültig“ im servoTEC S2 FS. Dieses meldet nach der Initialisierung der Antriebe die Referenz zurück, und somit muss keine Referenzfahrt mehr durchgeführt werden.



7.3 Antrieb Absolut positionieren



8 Ergänzende Informationen

8.1 Telegramme (Ein- und Ausgangsdaten)

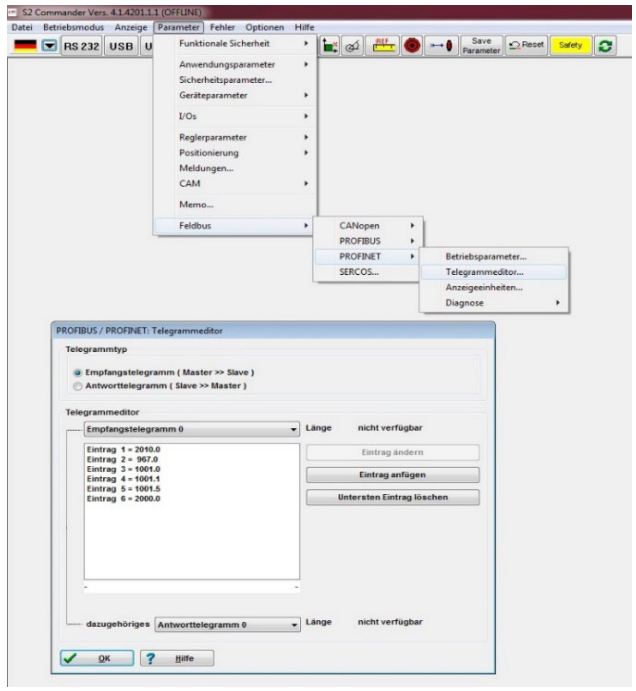
Im Projekt des PROFINET-Masters werden Datenbereiche erstellt, z. B. Datenbausteine. In diese Datenbereiche werden die Eingangs- und Ausgangsdaten von Master und Slave abgelegt. Beim Projektieren muss der Anwender die Inhalte und deren Reihenfolge sowie die Größe der beiden Datenbereiche übereinstimmend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

Empfangstelegramm: Übertragene Daten vom Master (SPS) zum Slave (servoTEC S2), auch als **Ausgangsdaten** bezeichnet

Antworttelegramm: Zu übertragende Daten vom Slave (servoTEC S2) zum Master (SPS), auch als **Eingangsdaten** bezeichnet

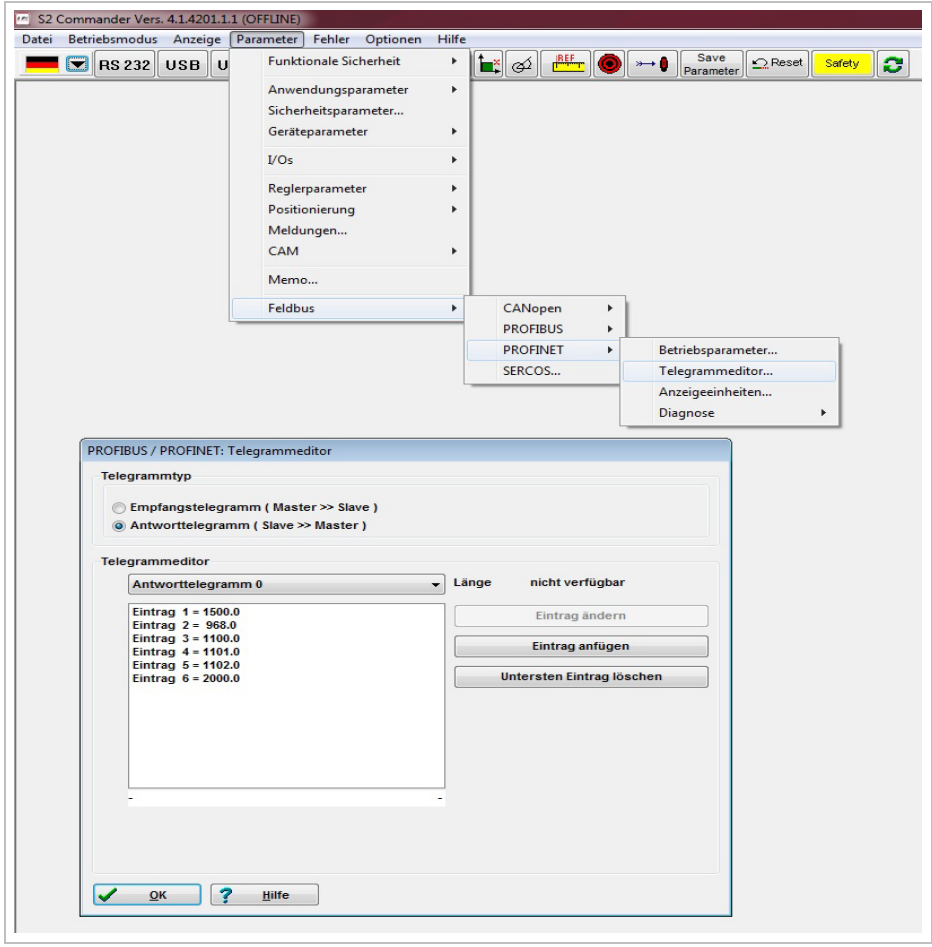
Die Parametrierung der Telegramme ist vor Aktivierung der Profinet-Kommunikation durchzuführen!

8.1.1 Empfangstelegramm für den servoTEC S2-Treiber



Adresse	Inhalt (Parameternummer)	Beschreibung
0	Kennung (= 0xE0)	Fest eingestellte Kennung
1	8 Bit Platzhalter (PNU 2010 0)	Frei
2	Control Word 1 (PNU 967 0)	Steuerwort zur Gerätesteuerung
4	Zielposition (PNU 1001 0)	Zielposition, Angabe in der für PROFINET eingestellten physikalischen Einheit einer Position
8	Fahrgeschwindigkeit (PNU 1001 1)	Fahrgeschwindigkeit während der Positionierung, Angabe in der für PROFINET eingestellten physikalischen Einheit einer Drehzahl
12	Beschleunigungen (PNU 1001 5)	Kombination der Werte für Beschleunigung und Verzögerung, Angabe in der für PROFINET eingestellten physikalischen Einheit einer Beschleunigung
16	Parameterkanal (PNU 2000)	Parameternutzdatenkanal

8.1.2 Antworttelegramm für den servoTEC S2-Treiber



Adresse	Inhalt (Parameternummer)	Beschreibung
0	Kennung (= 0xF0)	Fest eingestellte Kennung
1	Betriebsart (PNU 1500 0)	Aktuelle Betriebsart des servoTEC S2
2	Status Word 1 (PNU 968 0)	Statuswort zur Gerätesteuerung
4	Istposition (PNU 1100 0)	Aktuelle Istposition, Angabe in der für PROFINET eingestellten physikalischen Einheit einer Position
8	Istgeschwindigkeit (PNU 1101 0)	Aktuelle Istgeschwindigkeit, Angabe in der für PROFINET eingestellten physikalischen Einheit einer Geschwindigkeit
12	Wirkstromistwert (PNU 1102 0)	Über diesen Parameter wird der Wirkstrom-Istwert gelesen. Dieser wird bezogen auf den Motornennstrom zurückgegeben
16	Parameterkanal (PNU 2000)	Parameternutzdatenkanal

Application Document

“servoTEC S2 axes to be controlled by a Siemens S7 via Profinet-IO”

Edition 08/2024

Art. no.: 1434514

Trademarks and trade names are used without any warranty of their free usability. Texts and examples have been created with great care. Nevertheless, errors cannot be excluded. IEF-Werner GmbH does not assume legal responsibility nor any liability for missing or incorrect statements or their consequences.

IEF-Werner GmbH reserves the right to modify or improve the software or hardware or parts of it, as well as the supplied documentation or parts of it, without previous notice.

IEF-Werner GmbH expressly reserves all rights to replication and photomechanical reproduction, including in extracts.

We are always grateful for suggestions for improvements and information about errors.

© August 2024, IEF-Werner GmbH

Table of Contents

1 Changes to this application note 5

2 Prerequisites for commissioning 6

2.1 Download / update the “S2_Commander” parameterisation Software 6

2.2 Parameterisation interfaces on the servoTEC S2 FS 6

2.3 Supply servoTEC S2 FS 7

2.4 Basic parameterisation of the device 7

2.5 Safety module 7

3 Settings at the servoTEC S2 with the S2 Commander 8

3.1 Physical units 9

3.1.1 General information 9

3.1.2 Display units S2 Commander 10

3.2 Operation parameters PROFINET 13

4 Settings under SIEMENS TIA PORTAL 14

4.1 Example project 14

4.2 Hardware configuration Step 1: Select device 14

4.3 Hardware configuration Step 2: Assign IO controller 15

4.4 Hardware configuration Step 3: Select IO controller 16

4.5 Hardware configuration Step 4: Assign input data (PLC) 17

4.6 Hardware configuration Step 5: Assign output data (PLC) 17

4.7 Hardware configuration Step 6: Adjust IP address and subnet mask 18

4.8 Hardware configuration Step 7: Check HW identifiers 20

5 Minimum requirements for integration in own project 22

6 Data interface between PLC and servoTEC S2 (e.g. DB2) 27

6.1 Standard control range (→ axis is controlled) 27

6.1.1 Commands to axis 27

6.1.2 Status messages from axis 30

6.2 Extended control range (→ parameters in the servoTEC S2 FS are set) 31

6.2.1 Commands to controller 31

6.2.2 Status messages from controller 33

6.3 Example axis communication [DB2] → Linear axis with absolute feedback 34

7 Flow charts 37

7.1 Initialisation of the drive 37

7.2 Referencing the drive 38

7.3 Positioning the drive absolutely 39

8 Additional information 40

8.1 Telegrams (in- and output data) 40

8.1.1 Receipt telegram for the servoTEC S2 driver 40

8.1.2 Answer telegram for the servoTEC S2 driver 41

1 Changes to this application note

Document modifications and life cycle

Document Code	Date	Generation and modification
APP5048_EN_1434514_servoTECS2_Profinet-IO_SiemensS7_R1a.docx	06.05.2020	Template modified for Profinet-IO (SCL) → version: fbservoTEC-S2_e (V1.3)
APP5048_EN_1434514_servoTECS2_Profinet-IO_SiemensS7_R2a.docx	30.06.2020	Value jerk limit inserted, Interface modified → version: fbservoTEC-S2_e (V2.0)
APP5048_EN_1434514_servoTECS2_Profinet-IO_SiemensS7_R2b.docx	11.08.2021	Correction in hardware configuration → Assignment of the input and output data
APP5048_EN_1434514_servoTECS2_Profinet-IO_SiemensS7_R2c.docx	15.01.2024	Application Document revised by Service department
APP5048_EN_1434514_servoTECS2_Profinet-IO_SiemensS7_R3.docx	15.02.2024	Application Document revised by Service and Documentation department → Release R3!
APP5048_EN_1434514_servoTECS2_Profinet-IO_SiemensS7_R3a.docx	15.08.2024	Changes according to chapters 4.5 und 4.6. New: chapter 4.7.
APP5048_EN_1434514_servoTECS2_Profinet-IO_SiemensS7_R3b.docx	20.08.2024	Again new graphics / screenshots in sections 4.5, 4.6 and 4.9.

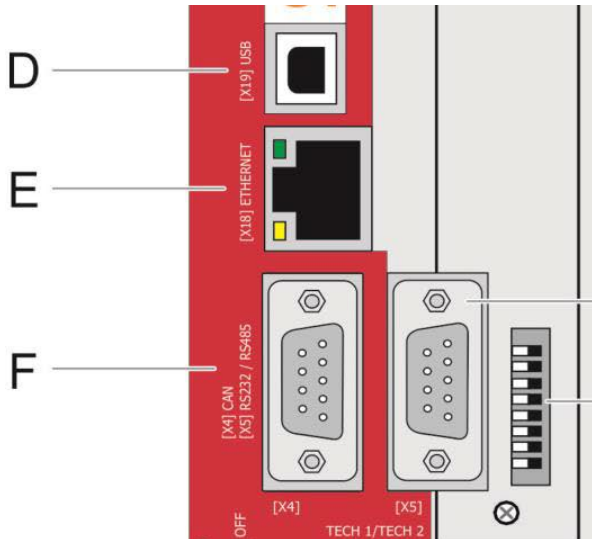
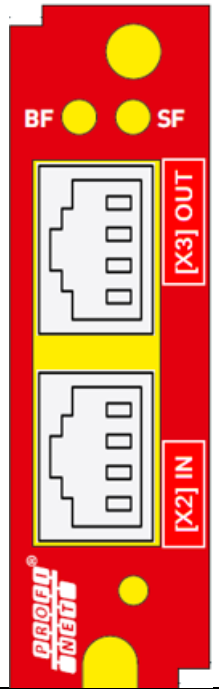
2 Prerequisites for commissioning

The following basic requirements must be met for successful commissioning:

2.1 Download / update the “S2_Commander” parameterisation Software

You will always find the latest “S2_Commander” in the download area of our website (www.ief.de). Please ensure that this is installed on your programming device.

2.2 Parameterisation interfaces on the servoTEC S2 FS

<table>
<tr>
<td rowspan="3"> 
</td>
<td>D: [X19] USB type-B interface (recommended)</td>
<td rowspan="3"> 

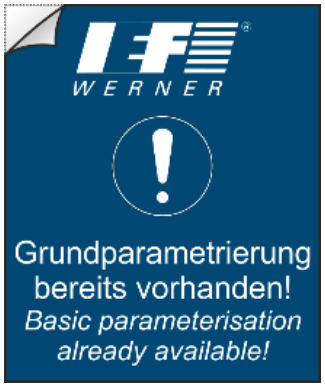
PN Technology module</td>
</tr>
<tr>
<td>E: [X18] Ethernet interface
Standard IP 10.1.6.1
Standard SUBNET mask 255.255.0.0</td>
</tr>
<tr>
<td>F: [X5] RS232 interfaces</td>
</tr>
</table>

2.3 Supply servoTEC S2 FS

Ensure that the supply voltage (DC link voltage present) and the +24V supply are connected to the device (7-segment indicator on the amplifier lights up). Without the supply voltages, the device (servoTEC S2 FS) cannot be successfully put into operation.

2.4 Basic parameterisation of the device

The servoTEC S2 FS device receives a basic parameterisation from IEF-Werner GmbH that matches the package, consisting of: servoTEC S2 FS, motor and axis. This can be seen from the sticker above the USB/Ethernet parameterisation interface. For information on operation and error messages, please open the "Help" tab in the S2 Commander and select "Topics". If anything is unclear, please contact our customer support.

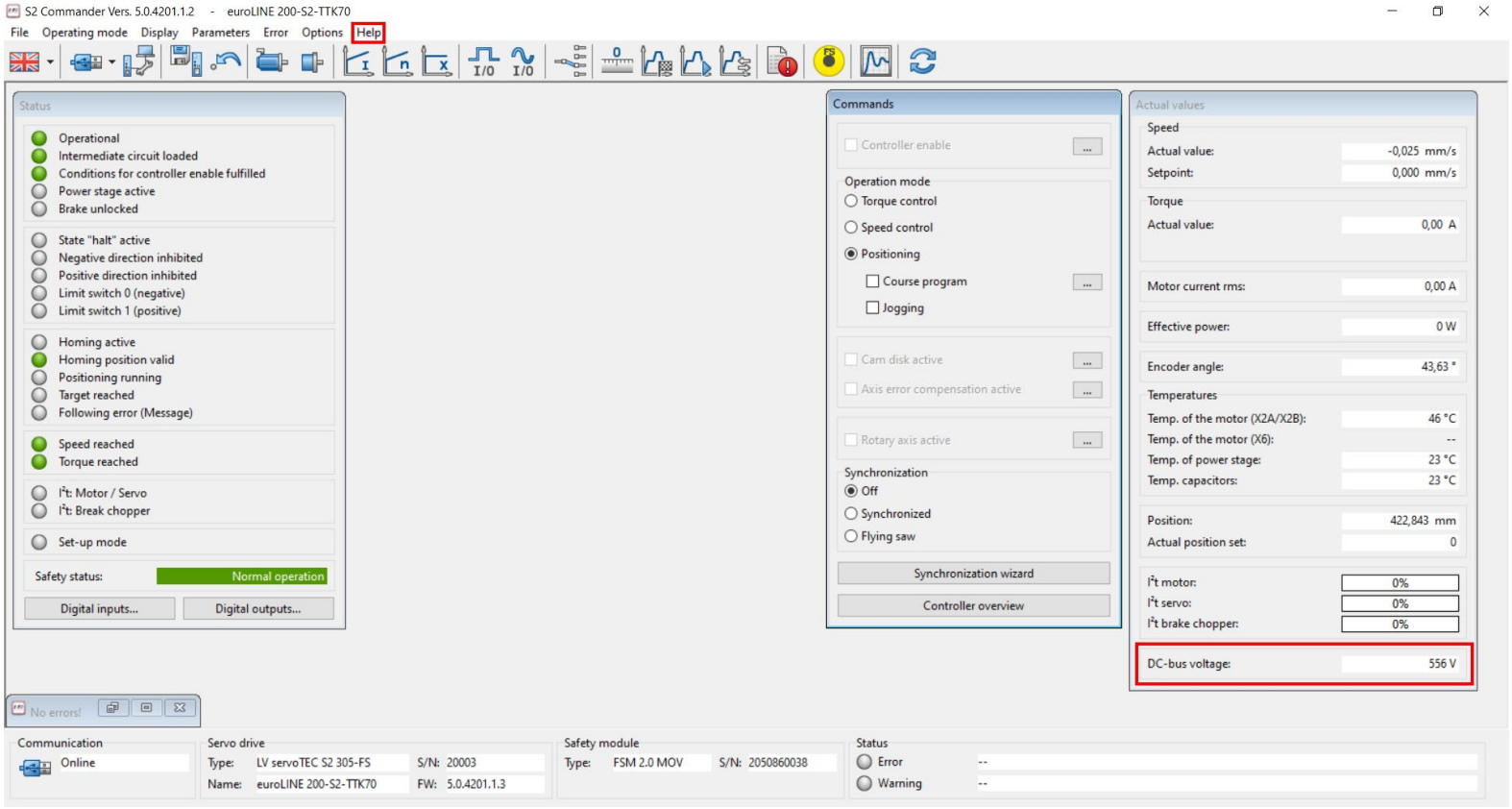


2.5 Safety module

If the device has a safety module, make sure that the device is connected and can be controlled in accordance with the circuit diagram supplied or the operating instructions. Otherwise it will not be possible to fully commission the servoTEC S2 FS.
The "State" LED on the STO or MOV module must light up green, which means "safe state reached".
The operating instructions for both modules can be found on our website (www.ief.de) in the download area.

3 Settings at the servoTEC S2 with the S2 Commander

The settings for the drive are made with the S2 Commander.



Standard IP: 10.1.6.1

Subnet mask: 255.255.0.0

Helpful information on handling and error messages can be found in the S2 Commander under "Help" ▶ "Topics…".

The DC link voltage must be > 300V DC for a single-phase device and > 500V DC for a three-phase device, otherwise the supply voltage will be missing.

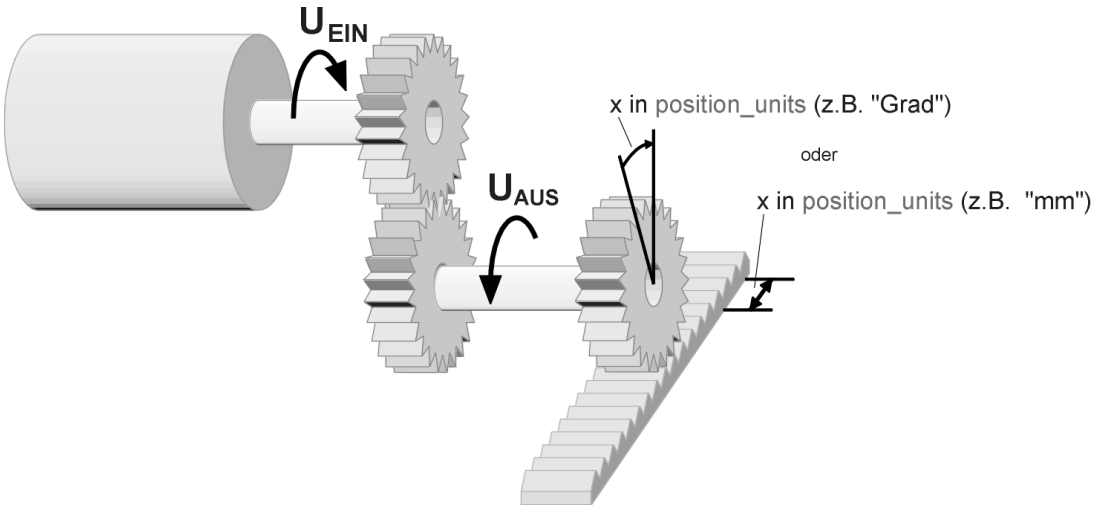
3.1 Physical units

3.1.1 General information

For correct function and to facilitate commission, it is important that the physical units of the fieldbus system (Profinet IO) and the diagnosis interface (S2 Commander) match.

The physical units are set with the S2 Commander in different windows:

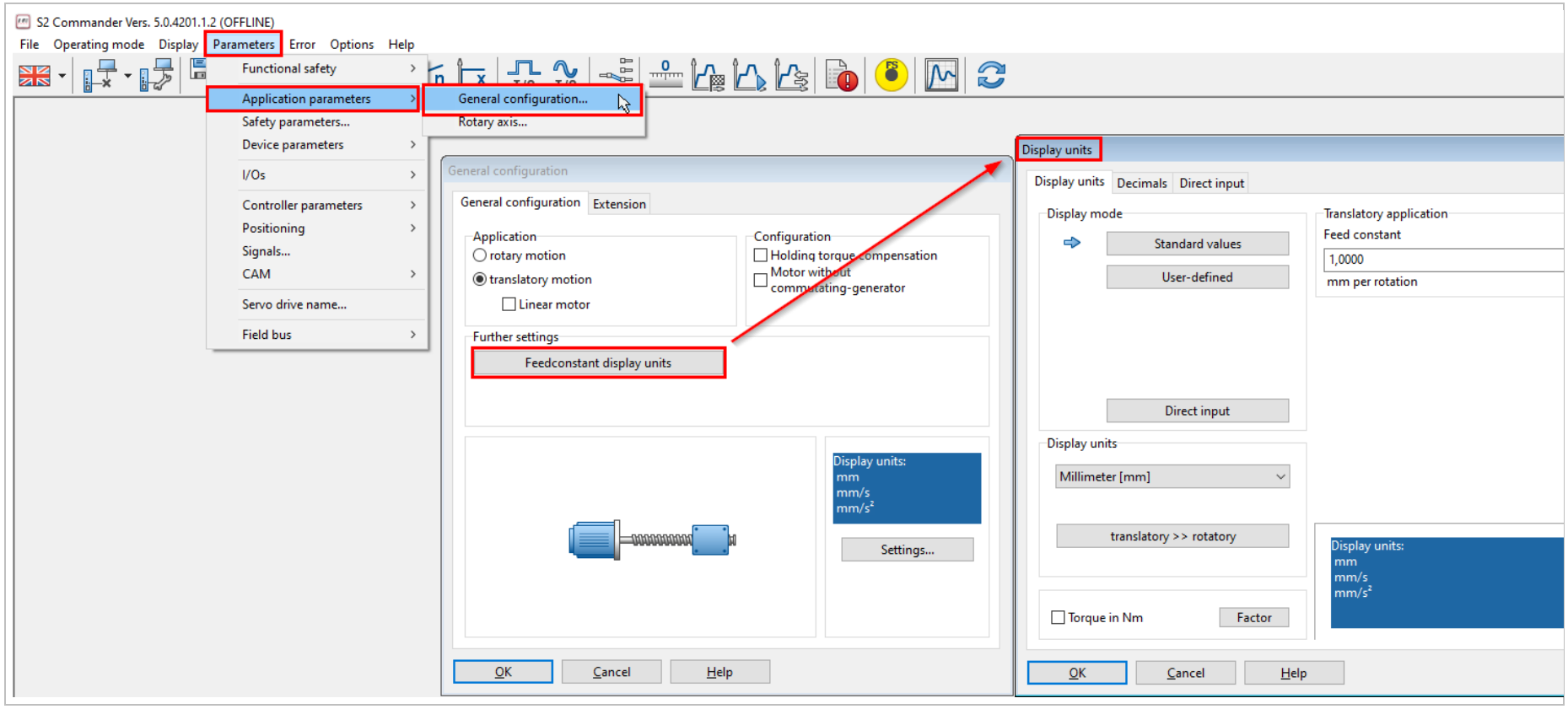
- For diagnosis with S2 Commander → Display units
- For the Profinet IO → Physical units Profinet IO
  → Physical units CANopen / Factor Group



Observe that some values cannot always be sensibly used. In a purely rotating system, e.g., an infeed constant is not needed. Beyond this, the infeed constant has a physical unit. If the parameters for this are not set properly, the infeed constant is not considered.

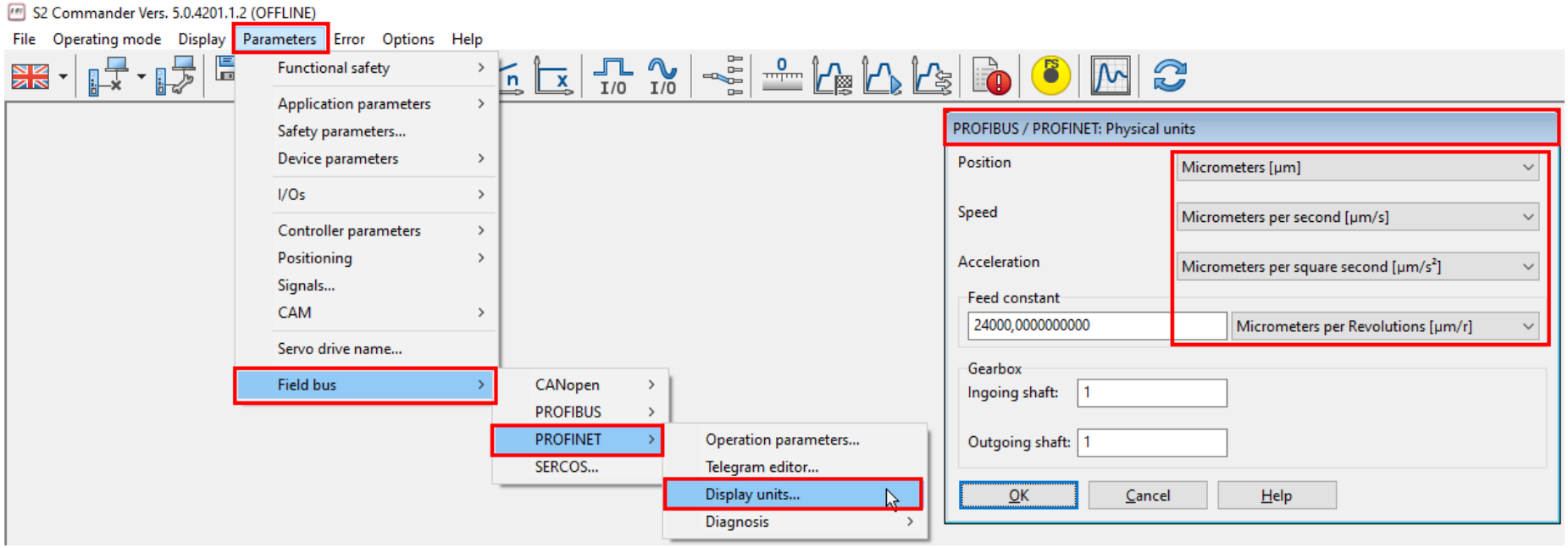
3.1.2 Display units S2 Commander

- The display units for the S2 Commander are set in the window "Display units".
- **Check**, whether the feed constant entered in the S2 Commander matches the feed constant on the type plate of the mechanical axis (linear axis, rotary axis, etc.).



3.1.2.1 Physical units Profinet-IO

For correct function, it is necessary that the units of the process data transmitted via the field bus are specified. They can be set via the window **PROFIBUS / PROFINET: Physical units** under **Parameters ▶ Field bus ▶ PROFINET ▶ Display units… The parameters for the physical units must be set once and do not need to be changed during a running application.**



When selecting the units, the corresponding factors are calculated internally so that the user only needs to select the desired unit. The gear factor and the infeed constant are entered as separate parameters. **Important: The physical units must be specified in µm! In the PLC it is still possible to work in mm. Accordingly, the feed constant must be specified here by a factor of 1000 greater. This is the prerequisite for the driver module from IEF-Werner to function correctly.**

Additional information: The set physical units are used to gain conversion factors automatically when they are entered in the firmware. They consist of an enumerator and a denominator that must not exceed the size of 32 Bit each. If there is an overflow when entering the factors here, the value will not be accepted. In this case, the factors or the physical units need to be corrected.

3.1.2.2 Physical units CANopen bus (relevant for PROFINET)

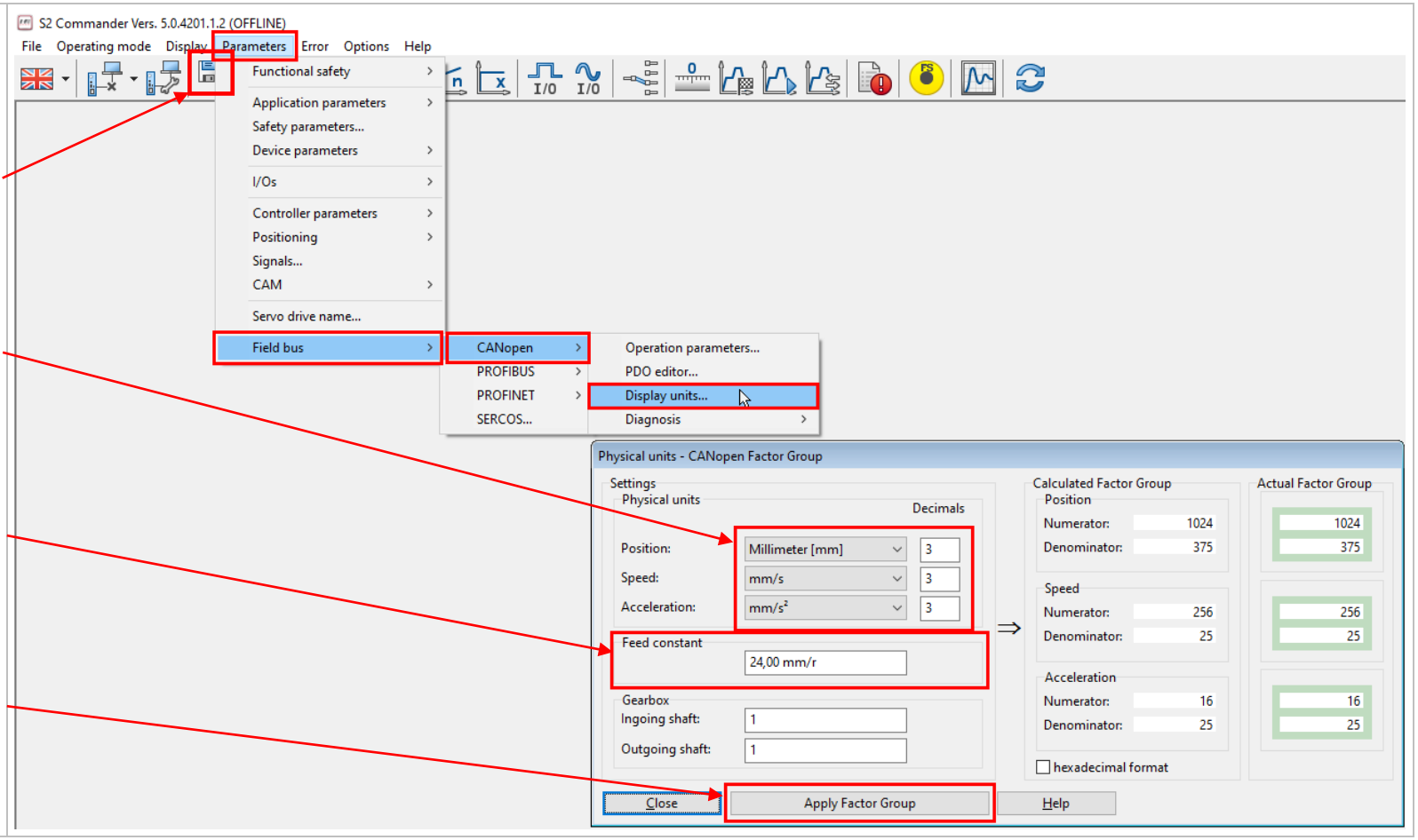
The following must be set:

After a change, the parameter set must be saved

Units in mm, U, ° and always three decimal places

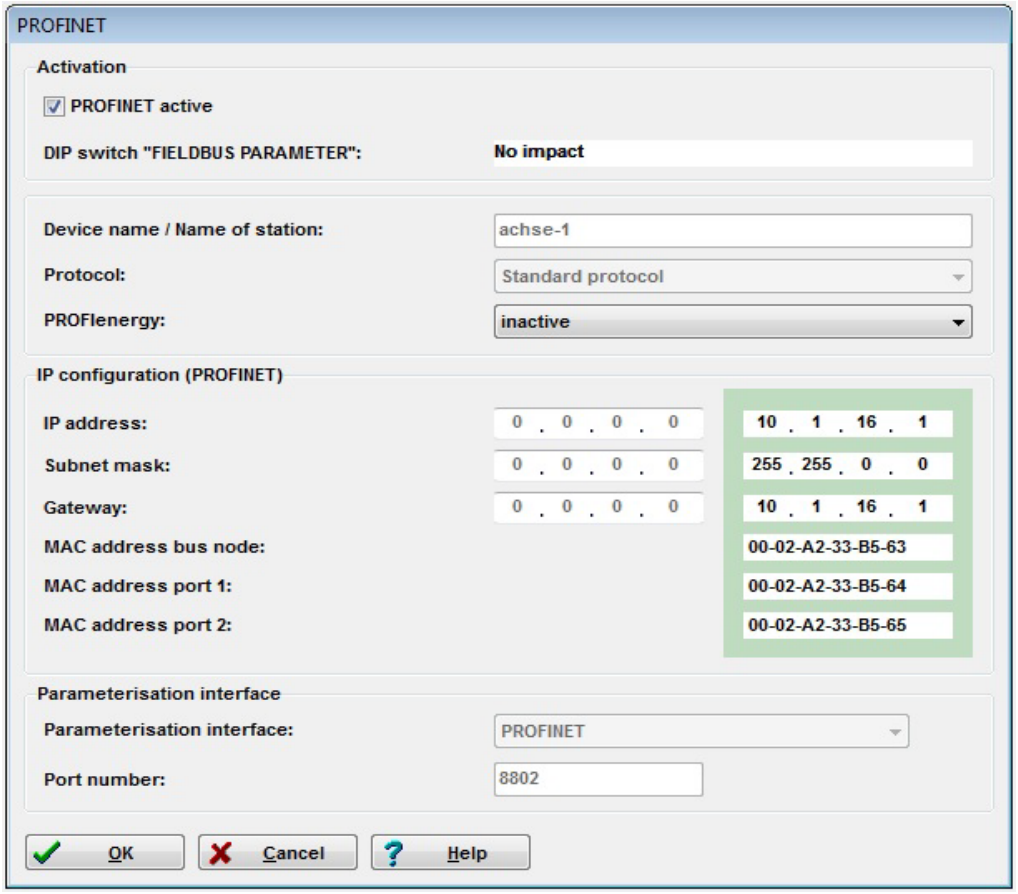
Feed constant (from type plate of the device)

Apply to Factor Group and then save the settings!



	For the parameters for the reference run via PROFINET to be interpreted properly by the servoTEC S2, it is required to set the physical units (display units) for the CANopen-Bus properly. Subsequently, the Factor Group shall be adopted and saved in the servo positioning controller.

3.2 Operation parameters PROFINET



Configuration for the Profinet communication on the side of the servo positioning controller (Parameters ▶ Field bus ▶ PROFINET ▶ Operation parameters).

Procedure:

1. Deactivate "**PROFINET active**"
2. Edit „Device name“
3. Edit "IP address"
4. Edit „Subnet mask“
5. Select „Parameterisation interface“
6. Activate "**PROFINET active**"

Observe that the Profinet communication activating is only complete displayed after a few seconds.

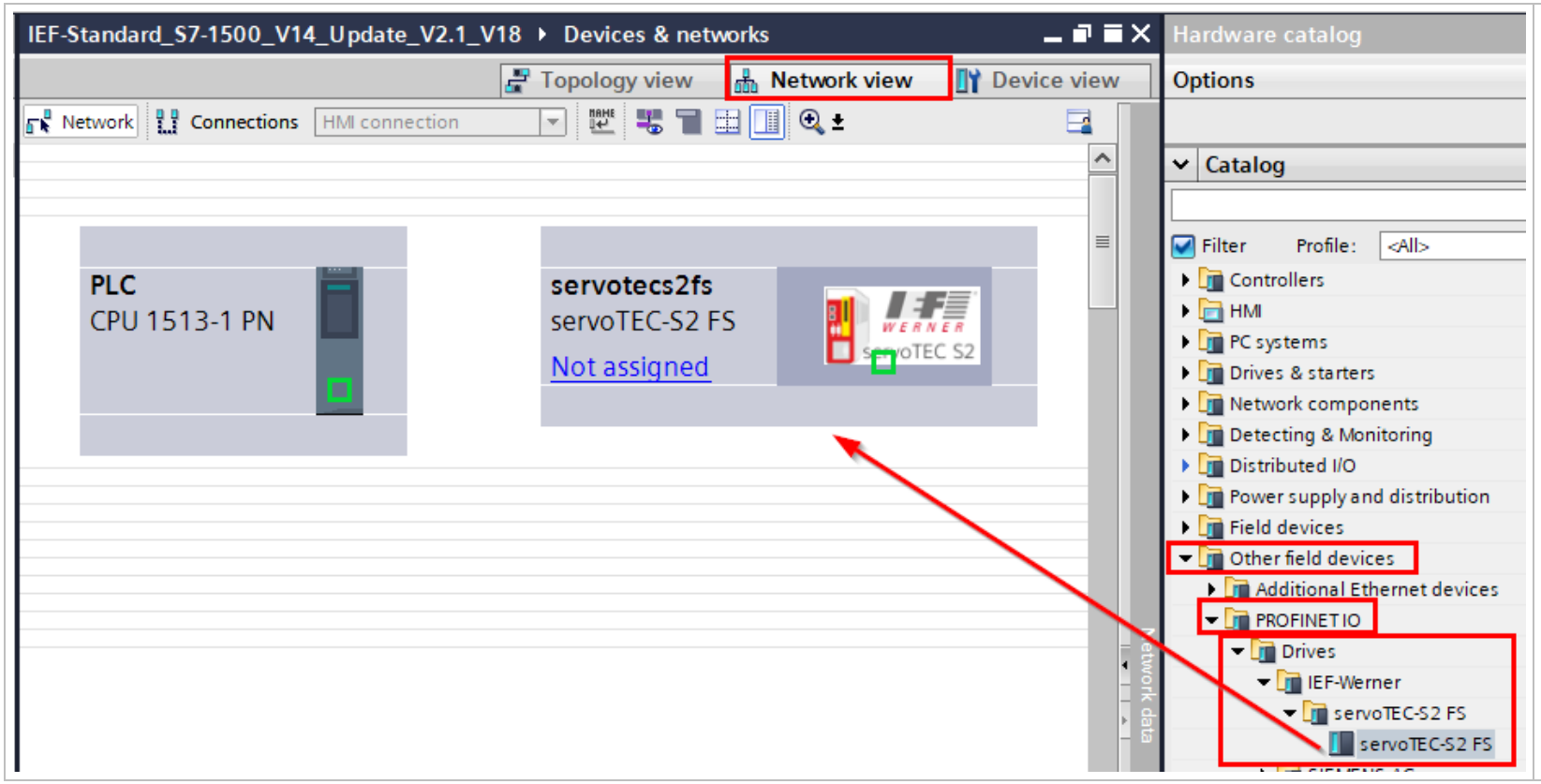
SAVE! → The parameterisation is now permanently saved in the servo amplifier.

4 Settings under SIEMENS TIA PORTAL

4.1 Example project

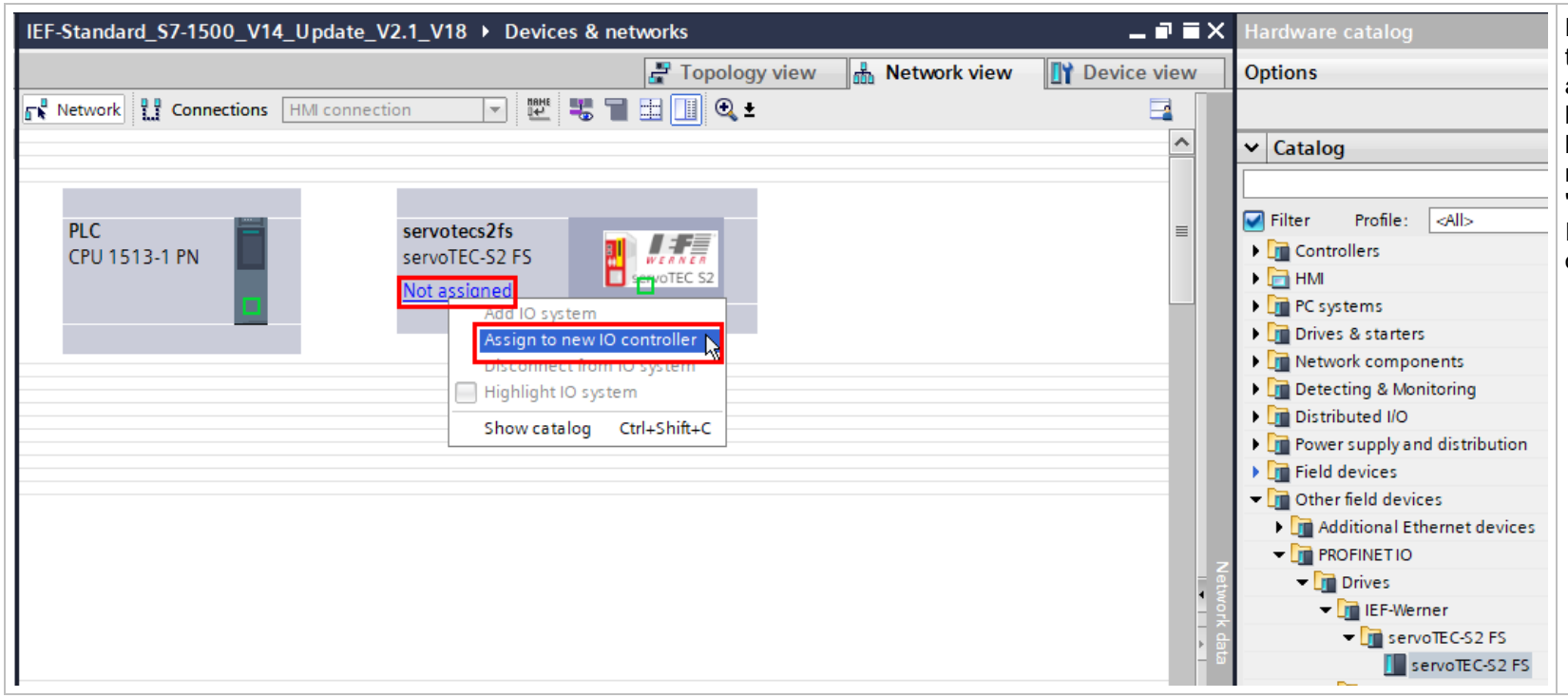
- The sample project can be found in the download area of the IEF-Werner website.
- The sample project must be opened once. The field device is added to your TIA hardware catalogue.
- You can now test the axis in the sample project or integrate it into your project.

4.2 Hardware configuration Step 1: Select device



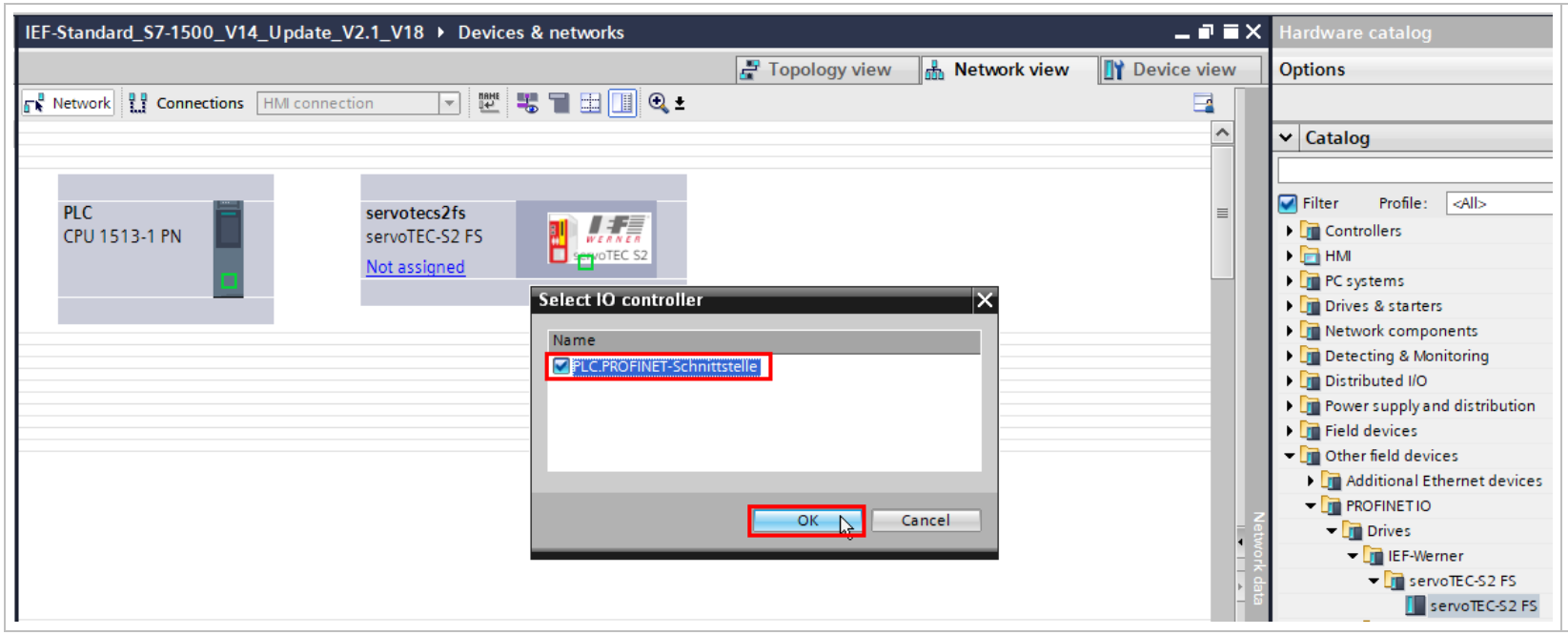
The field device is located in the path marked here in the hardware catalogue. It can be added to the network view as often as required by double-clicking or drag & drop.

4.3 Hardware configuration Step 2: Assign IO controller



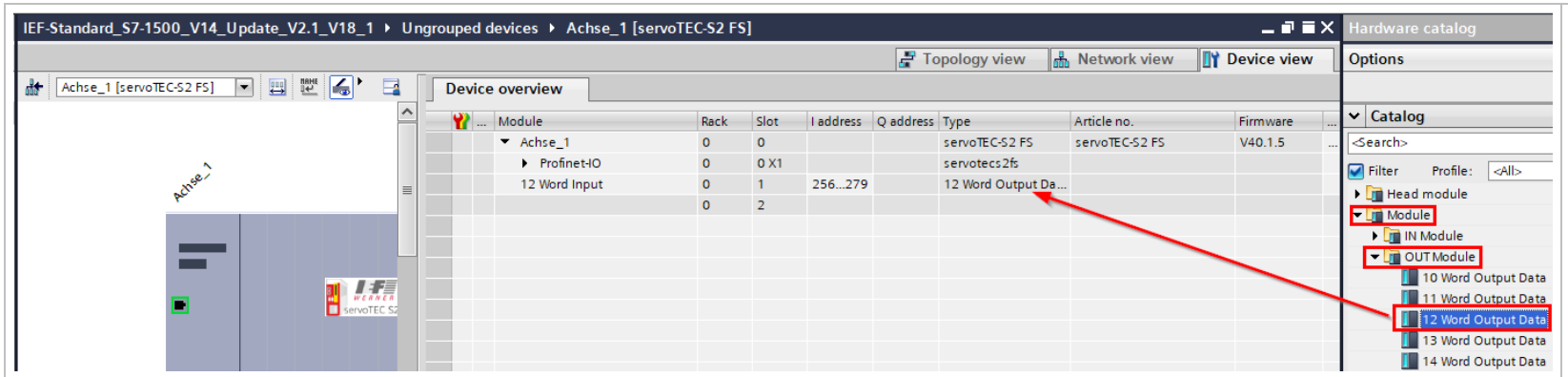
Right-click on the "Not assigned" text highlighted in blue to open the menu with the "Assign to new IO controller" option.

4.4 Hardware configuration Step 3: Select IO controller



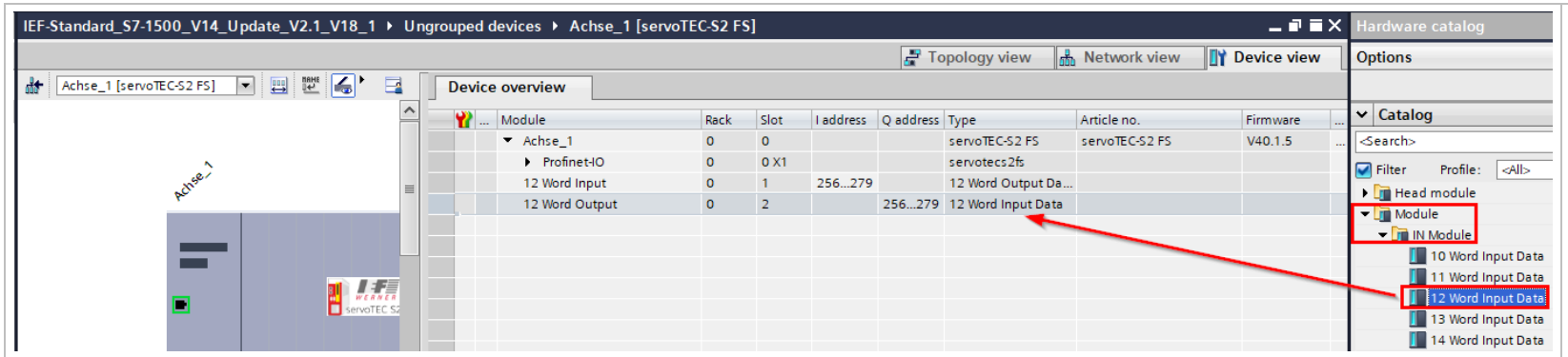
Select the interface and confirm with "OK".

4.5 Hardware configuration Step 4: Assign input data (PLC)



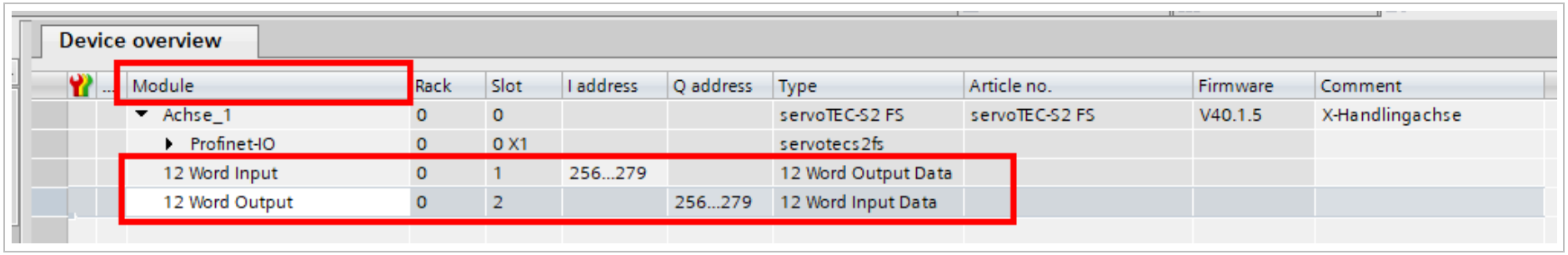
The areas can be added by double-clicking or drag & drop.

4.6 Hardware configuration Step 5: Assign output data (PLC)



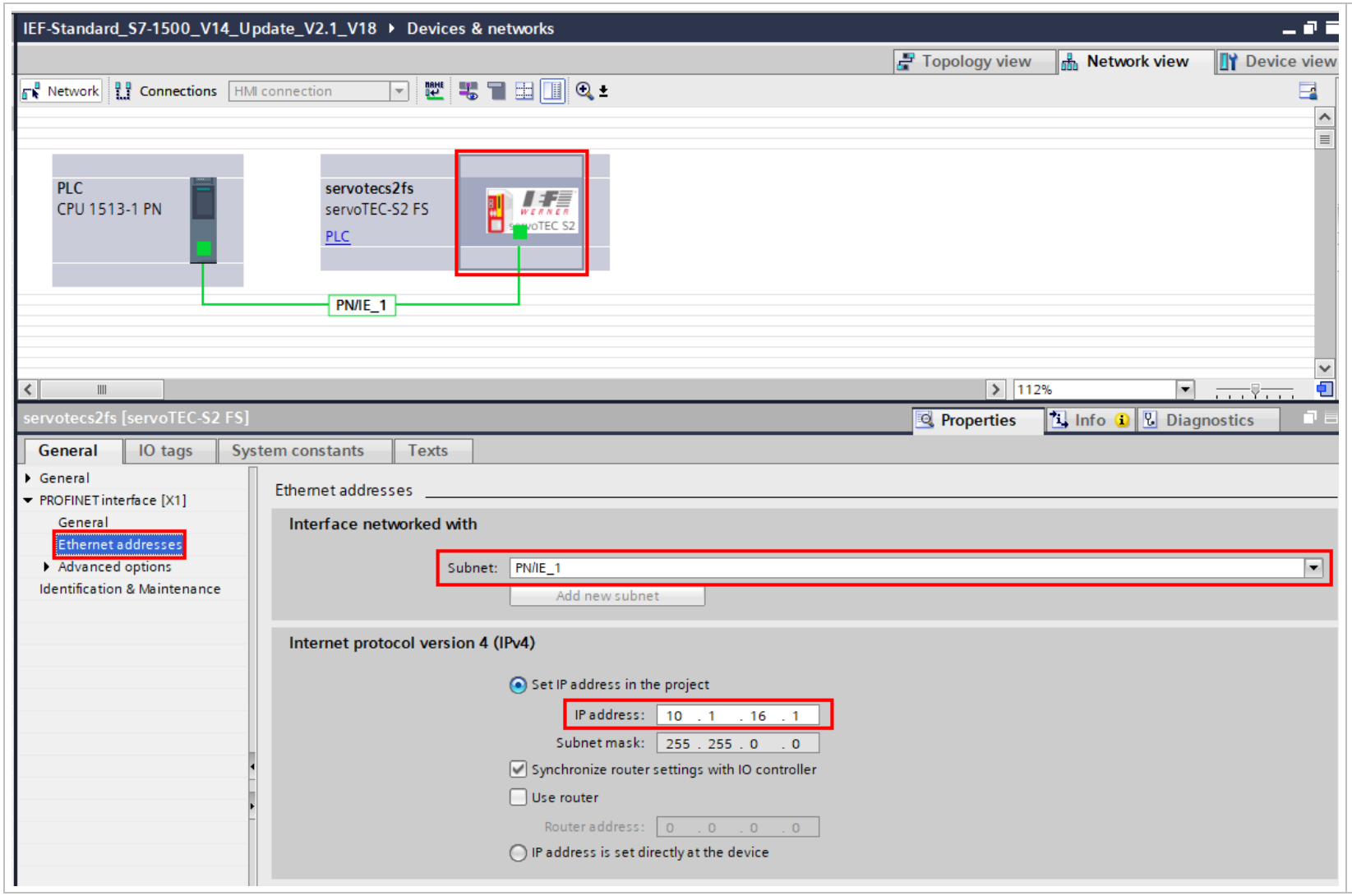
The areas can be added by double-clicking or drag & drop.

4.7 Checking the assignment of data



Please note the correct assignment of the input/output data (shown here in the image). If this is not correct, you can change the description in the Modules area by changing the text in '12 Word Input' to '12 Word Output' and vice versa. The display should then be as shown in the image (above).

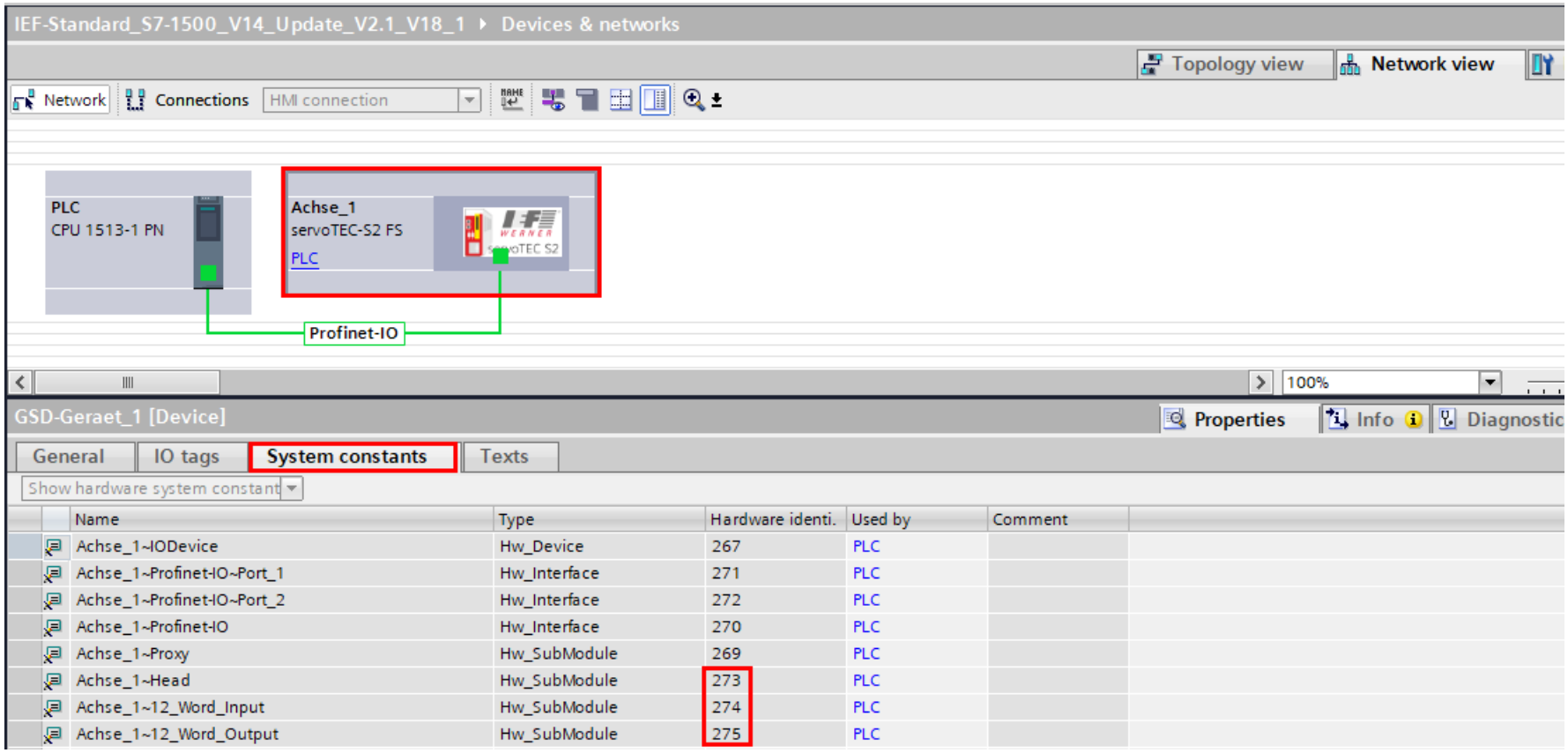
4.8 Hardware configuration Step 6: Adjust IP address and subnet mask



The default IP address for the first participant in the bus is:

10.1.16.1.

4.9 Hardware configuration Step 7: Check HW identifiers



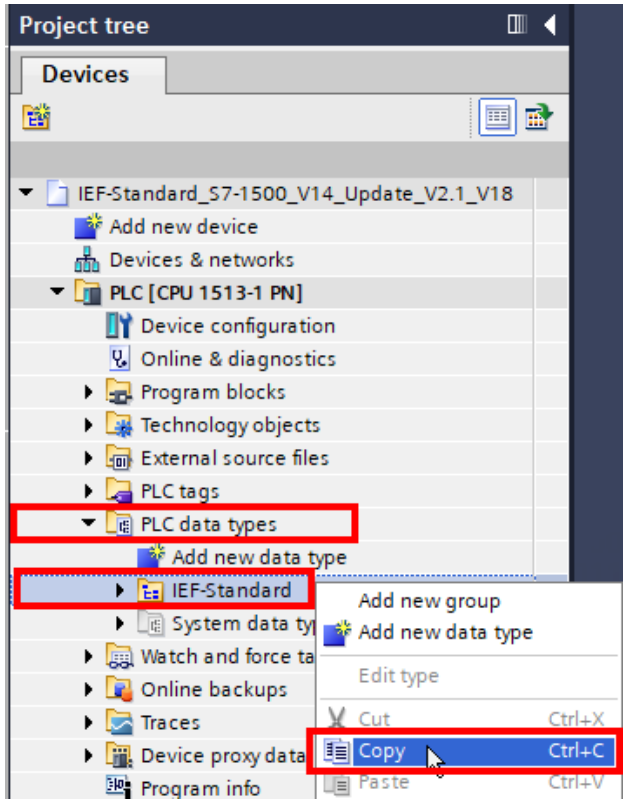
Name	Type	Hardware identi.	Used by	Comment
Achse_1~IODevice	Hw_Device	267	PLC	
Achse_1~Profinet-IO~Port_1	Hw_Interface	271	PLC	
Achse_1~Profinet-IO~Port_2	Hw_Interface	272	PLC	
Achse_1~Profinet-IO	Hw_Interface	270	PLC	
Achse_1~Proxy	Hw_SubModule	269	PLC	
Achse_1~Head	Hw_SubModule	273	PLC	
Achse_1~12_Word_Input	Hw_SubModule	274	PLC	
Achse_1~12_Word_Output	Hw_SubModule	275	PLC	

CHECK	In the hardware configuration, the hardware identifier of the device (head, input data and output data) must be specified in ascending order and in direct succession. Only the HW identifier of the device (head) is transferred to the function block!
IF NOT THEN	If the HW identifier is not in ascending order, this can be remedied by adding a new servoTEC S2 FS from the field devices and repeating the hardware configuration from step 1. If the HW identifier of the newly added device is correct, the old device can be deleted from the network view. If this is not the case, add a new device.
NOTE	The directly consecutive HW identifier is necessary so that the parameters for the reference run via PROFINET are interpreted correctly by the servoTEC S2 FS. Unfortunately, the assignment is made automatically by TIA and cannot be influenced or subsequently changed.

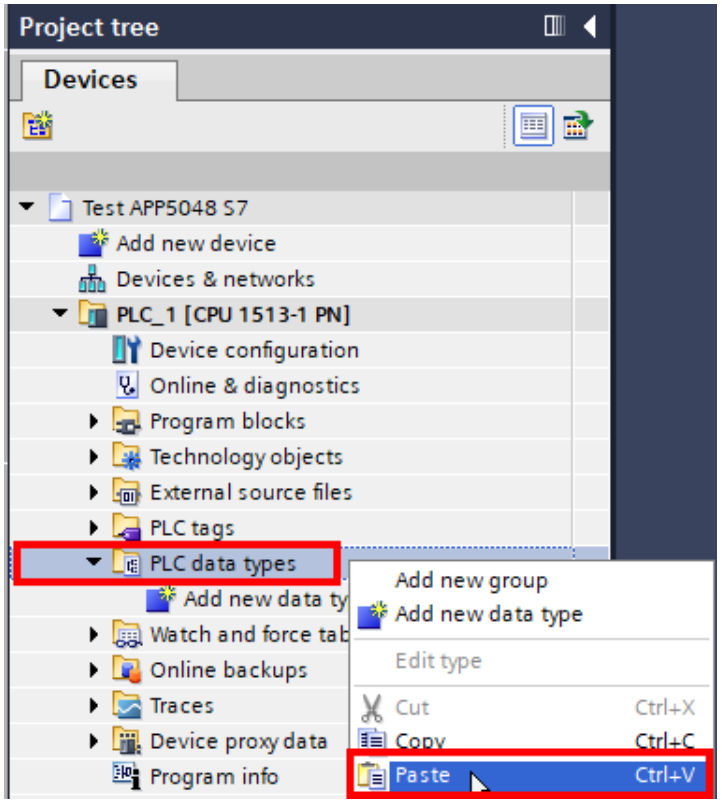
5 Minimum requirements for integration in own project

To integrate it into your project, you must perform the following actions:

1. Copy the "IEF-Standard" folder from our example project under "PLC data types" and add it to the PLC data types in your project.

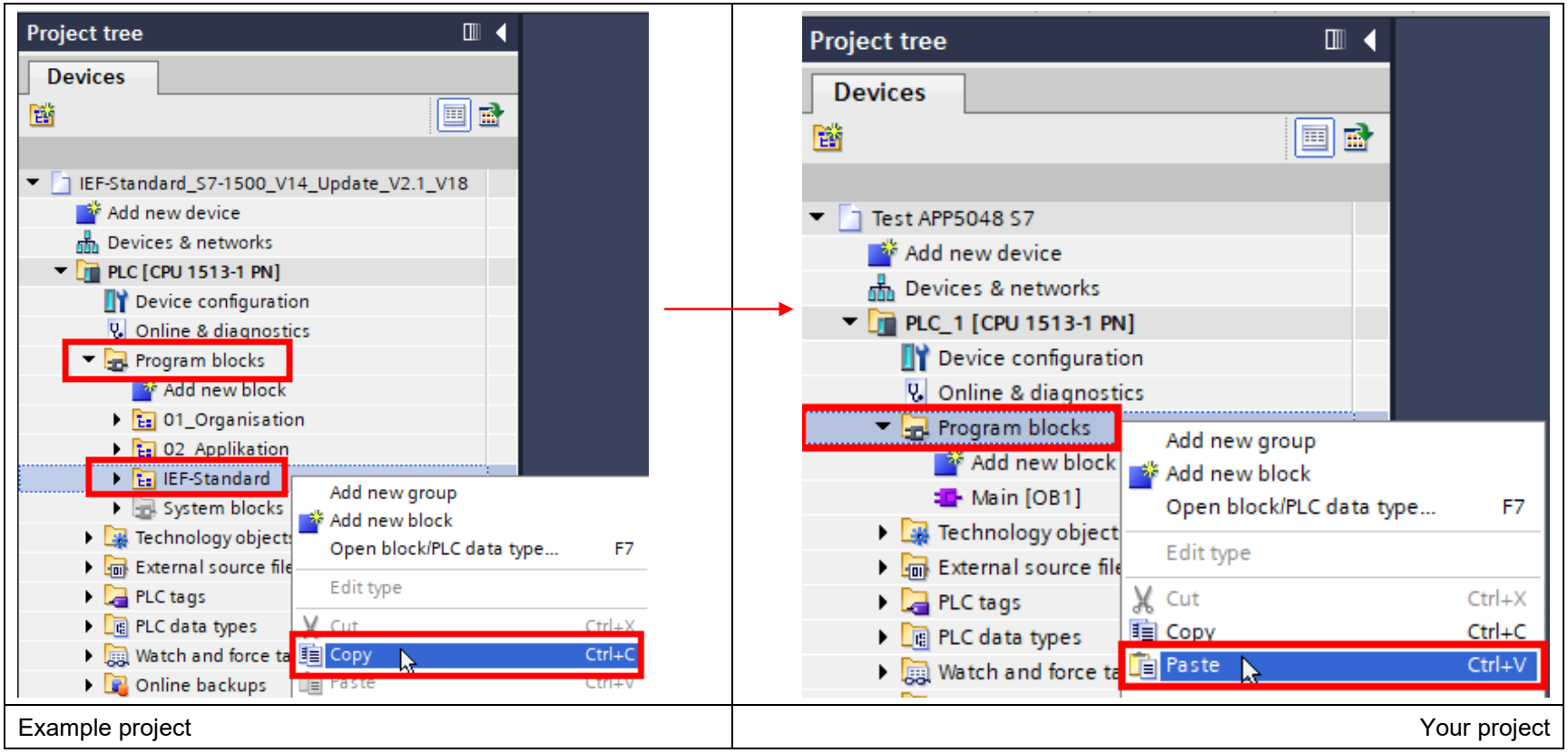


Example project



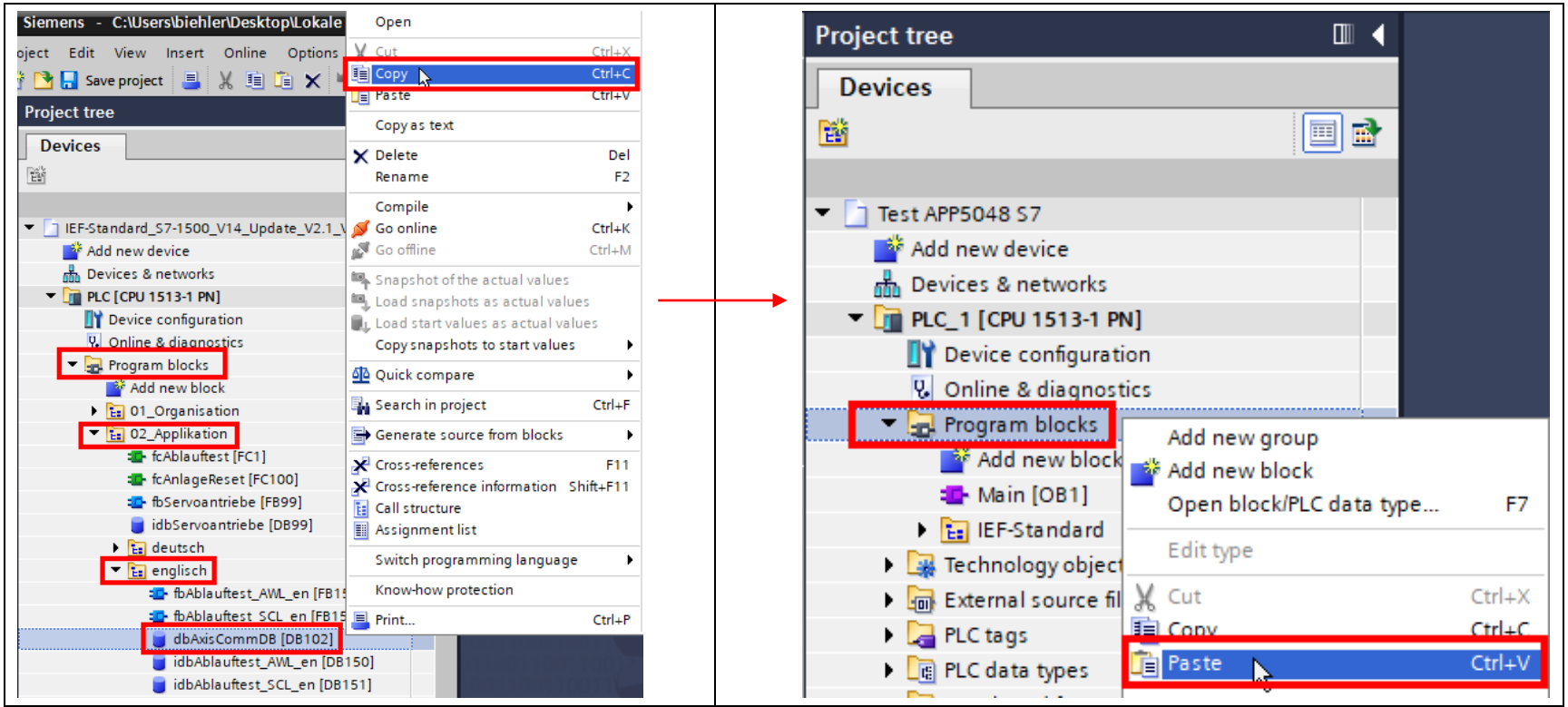
Your project

2. Copy the "IEF-Standard" folder from our sample project under "Program blocks" and add it to the Program blocks in your project.



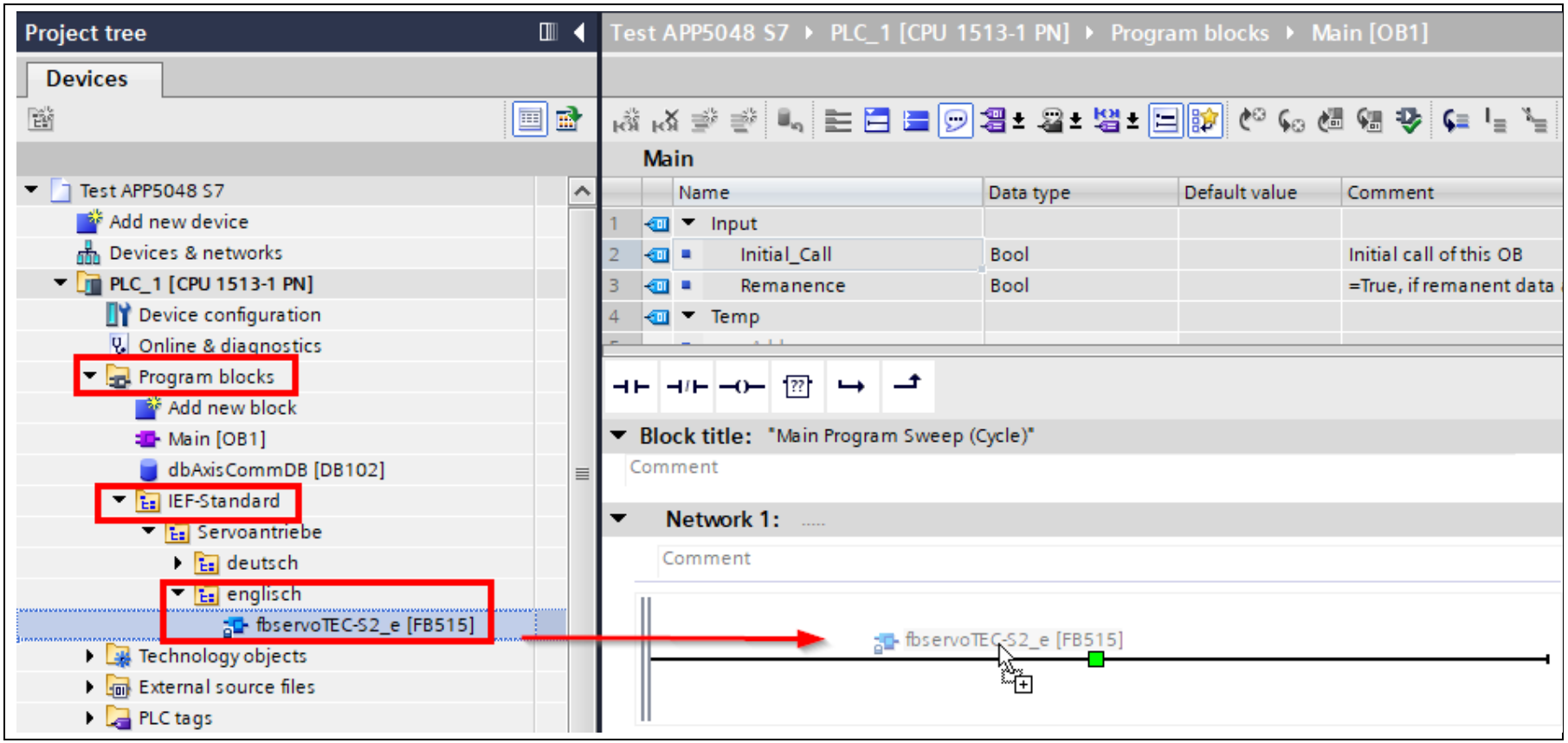
Example project	Your project

3. Copy the data block "dbAchsKommDB [DB102]" from our sample project under "Program blocks" → "02_Application" → "englisch" and add it to your project.

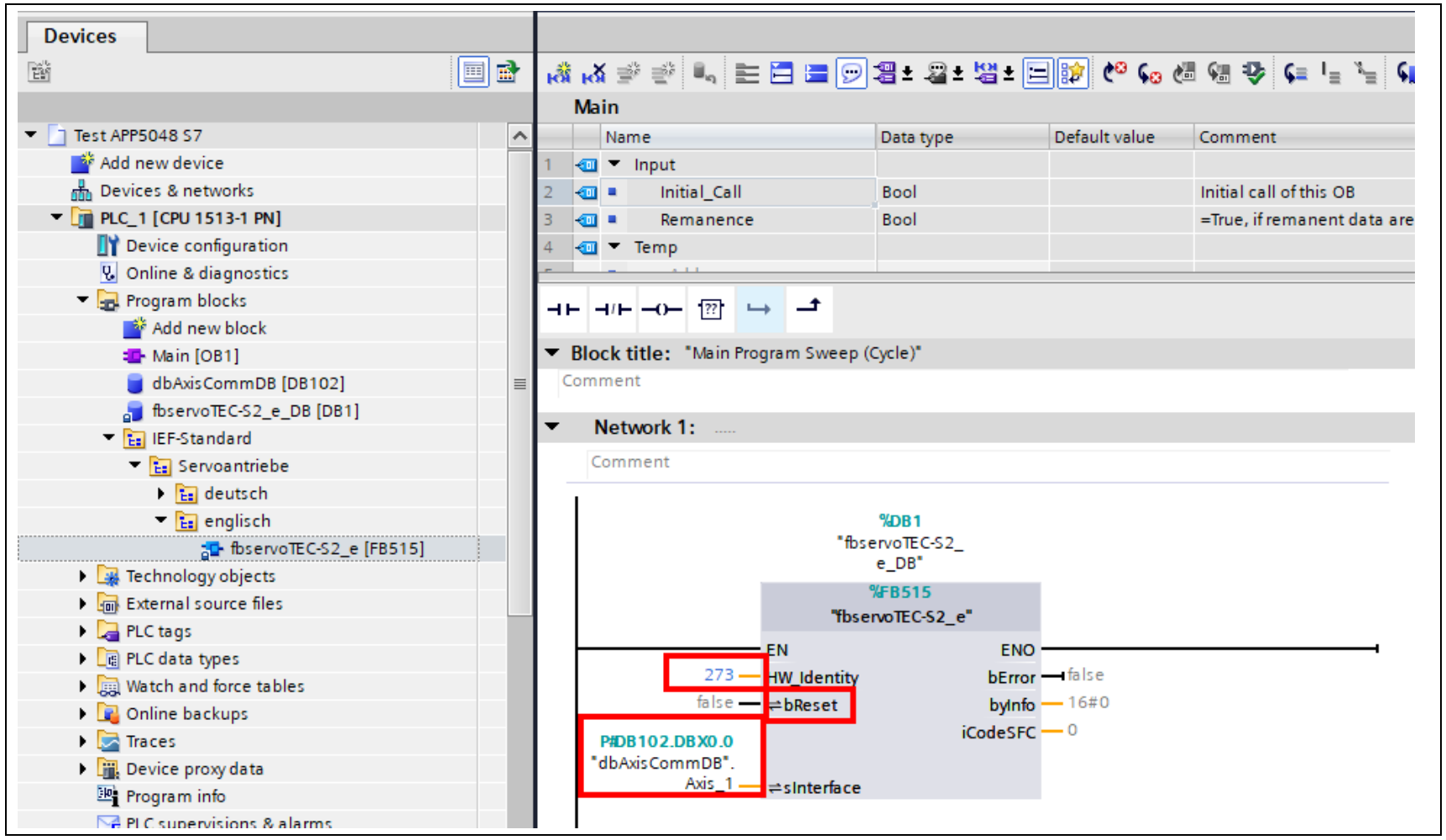


Example project	Your project

4. Add the [FB515] to your main OB.



5. Connect the block with the HW identifier (head) and the "dbAchsKommDB" axis_xx. The block should be reset every time the CPU is started. It should therefore be executed in OB100.



6 Data interface between PLC and servoTEC S2 (e.g. DB2)

Two interface structures are used as the data interface:

- "Standard control range (→ axis is controlled)"
- "Extended control range (→ parameters in the servoTEC-S2 FS are set)"

These signals (data) must be linked in the user programme if required.

6.1 Standard control range (→ axis is controlled)

6.1.1 Commands to axis

Designation	Type	Function
AxisCmd.rTargetPosition	Real	Target position for moving absolute. → Value can be changed during an activated moving order!
AxisCmd.rDistanceValue	Real	Distance value for moving relative. → Value can be changed during an activated moving order!
AxisCmd.rOverrideValue	Real	Velocity value in procent from „**AxisCmd.rTraversingSpeed**“ → Value can be changed during an activated moving order!
AxisCmd.rTraversingSpeed	Real	Traversing speedy → is influenced by “**AxisCmd.rOverrideValue**”. Velocity for an absolute or relative order
AxisCmd.rReferenceSpeed	Real	Homing speed → Search run for the switch (search for switch)
AxisCmd.rJoggingSpeed	Real	Jogging speed → Velocity for jog mode!
AxisCmd.rAcceleration	Real	Acceleration of the axis → "**start ramp**"
AxisCmd.rDeceleration	Real	Deceleration of the axis → "**stop ramp**"
AxisCmd.bRelease	Bool	Release axis from application → Power voltage for servo controller is on!

Designation	Type	Function
AxisCmd.bStartInit	Bool	The axis is initialised → Specification of operating mode "**Positioning**"! After initialisation, the flag "**AxisStatus.bInitFlag**" is set and the command "**AxisCmd.bStartInit**" is reset as confirmation. If the axis already has a valid reference point, the flag "**AxisStatus.bRefFlag**" is also set. Initialisation is interrupted at once when the command "**AxisCmd.bStartInit**" is reset.
AxisCmd.bStartHoming	Bool	The axis performs a reference run. → Finding the switch with "**AxisCmd.rReferenceSpeed**" → Moving off of the switch with "**ServoCmd.rSearchZeroSpeed**" → Safe zero offset by "**ServoCmd.rReferenceOffset**" After the reference run, the flag "**AxisStatus.bRefFlag**" is set and the command "**AxisCmd.bStartHoming**" is reset as confirmation. A reference run is interrupted at once when the command "**AxisCmd.bStartHoming**" is reset. A reference run with an absolute value encoder can only be performed when the command "**ServoCmd.bABS_Learning**" is set additionally. After the reference run, the flag "**AxisStatus.bJustFlag**" is set and "**ServoCmd.bABS_Learning**" is reset as confirmation.
AxisCmd.bStartSetPos	Bool	The axis is set to reference position. → Save absolute position on "**ServoCmd. bReferencePosition**" After setting the reference position, the two flags "**AxisStatus.bRefFlag**" and "**AxisStatus.bSetFlag**" are set and the command "**AxisCmd.bStartSetPos**" is reset as confirmation. Reference setting is interrupted at once when the command "**AxisCmd.bStartSetPos**" is reset. Setting the reference with an absolute value encoder can only be performed when the command "**ServoCmd.bABS_Learning**" is set additionally. After the reference is set, the flag "**AxisStatus.bJustFlag**" is set and "**ServoCmd.bABS_Learning**" is reset as confirmation.

Designation	Type	Function
AxisCmd.bStartAbs	Bool	The axis is started with the "**AxisCmd.rTraversingSpeed**" absolute to the **"AxisCmd.rTragetPosition**". When the axis has reached its target position, the flag "**AxisStatus.bAbsInPos**" is set and the command "**AxisCmd.bStartAbs**" is reset as confirmation. The started running order is interrupted at once and rejected when the command "**AxisCmd.bStartAbs**" is reset.
AxisCmd.bStartRelPos	Bool	The axis is started in the positive direction with **"AxisCmd.rTraversingSpeed**" by the "**AxisCmd.rDistanceValue**". After movement by the distance value, the flag "**AxisStatus.bRelInPos**" is set and the command "**AxisCmd.bStartRelPos**" is reset as confirmation. The started running order is interrupted at once and rejected when the command "**AxisCmd.bStartRelPos**" is reset.
AxisCmd.bStartRelNeg	Bool	The axis is started in the negative direction with **"AxisCmd.rTraversingSpeed**" by the "**AxisCmd.rDistanceValue**". After movement by the distance value, the flag "**AxisStatus.bRelInPos**" is set and the command "**AxisCmd.bStartRelNeg**" is reset as confirmation. The started running order is interrupted at once and rejected when the command "**AxisCmd.bStartRelNeg**" is reset.
AxisCmd.bStartJogPos	Bool	The axis moves with "**AxisCmd.rJoggingSpeed**" in the positive direction while the command "**AxisCmd.bStartJogPos**" is set. → The override value is controller internal 100%!
AxisCmd.bStartJogNeg	Bool	The axis moves with "**AxisCmd.rJoggingSpeed**" in the negative direction while the command "**AxisCmd.bStartJogNeg**" is set. → The override value is controller internal 100%!
AxisCmd.bStopAxis	Bool	Command stop axis → intermediate stop is active. If the command **"AxisCmd.bStopAxis**" is output during a running order, the axis is stopped at once. When the command is taken back, the interrupted running order is continued.

Designation	Type	Function
AxisCmd.bAcknowledge	Bool	Acknowledge error to controller

6.1.2 Status messages from axis

Designation	Type	Function
AxisStatus.rActualPosition	Real	Current position of the axis
AxisStatus.bReady	Bool	Flag active if the axis is ready for operation → servo controller is enabled!
AxisStatus.bInitFlag	Bool	Flag active if the axis is initialized → controller mode „**Positioning**“ active!
AxisStatus.bJustFlag	Bool	Flag active if the sequence "**Homing**" or "**Set Position**" is finished → Flag is only relevant by an **absolute feedback system**!
AxisStatus.bRefFlag	Bool	Flag active if the axis is referenced → after sequences „**Homing**“ or „**Set Position**“
AxisStatus.bSetFlag	Bool	Flag active if the sequence "**Set Position**" is finished
AxisStatus.bAbsInPos	Bool	Flag active if the axis had reached the position after an absolute moving order. → Target value: "**AxisCmd.rTargetPosition**", → Enable start: "**AxisCmd.bStartAbs**"
AxisStatus.bRelInPos	Bool	Flag active if the axis had reached the position after a relative moving order. → Target value: "**AxisCmd.rDistanceValue**", → Enable start: "**AxisCmd.bStartRelPos**" or "**AxisCmd.bStartRelNeg**"
AxisStatus.bBusy	Bool	Flag active if an order is started → sequence in driver is active!
AxisStatus.bError	Bool	Flag active if an error is present
AxisStatus.bWarning	Bool	Flag active if a warning is present

6.2 Extended control range (→ parameters in the servoTEC S2 FS are set)

6.2.1 Commands to controller

Designation	Type	Function
ServoCmd.rReferenceOffset	Real	The axis is moved to the reference point in the sequence "**Homing**" by this offset. → Value reference offset: zero point minus reference point (switch, blocking)
ServoCmd.rReferencePosition	Real	In the sequence "**Set Position**" the axis position will be set to this absolute position.
ServoCmd.rRedCurrentValue	Real	Motor current for the torque reduction → value in [A]. The torque behaves proportional to the motor current. If the motor current is too low, the axis can no longer be moved.
ServoCmd.rSearchZeroSpeed	Real	Speed to search for zero → moving from the switch (search for zero)
ServoCmd.rRefAcceleration	Real	Acceleration for sequence homing (→ start / stop ramp)
ServoCmd.rJogAcceleration	Real	Acceleration for jogging mode (→ start / stop ramp)
ServoCmd.rMinimalMovingArea	Real	Minimal moving area limit from the axis
ServoCmd.rMaximalMovingArea	Real	Maximal moving area limit from the axis
ServoCmd.rValueJerkLimit	Real	Value jerk limit for ramping (→ start / stop ramp)
ServoCmd.iHomingMethod	Int	Method for homing sequence -1 / -2 = blocking negative / positive with zero mark, 1 / 2 = limit switch negative / positive with zero mark, 7 / 11 = reference switch negative / positive with zero mark, -17 / -18 = blocking negative / positive, 17 / 18 = limit switch negative / positive, 23 / 27 = reference switch positive / negative, 34 = actual position → see "MAN_EN_1136180_LV-servoTEC_S2_SoftwareManual.pdf"

Designation	Type	Function
ServoCmd.iModeRotaryAxis	Int	Mode for rotary axis 0 = inactive, 1 = shortest way (Modulo), 2 = direction from moving order, 3 = direction only positive, 4 = direction only negative
ServoCmd.bABS-Feedback	Bool	True: Motor with absolute feedback (→ **EnDat , Hiperface!**) False: Motor without absolute feedback (→ **Resolver!**)
ServoCmd.bABS_Learning	Bool	True: Absolute system new learning False: not applicable To newly teach an absolute value encoder, this flag must be set before starting the actions "**Homing**" or "**Set Position**". After completion of the actions, the calculated position offset is saved on the encoder EEPROM and the release flag is reset.
ServoCmd.bLimitJerk	Bool	True: with jerk limitation → calculated ramp (sin²) False: without jerk limitation → linear ramp
ServoCmd.bReduceTorque	Bool	True: reduce torque (→ over current value!) False: no reducing
ServoCmd.bWithoutLagLimit	Bool	True: deactivate lag distance limit False: activate lag distance limit
ServoCmd.bWithoutSWLimit	Bool	True: deactivate SW limit switches False: activate SW limit switches
ServoCmd.bRequestSafety	Bool	True: activate "request safety" False: deactivate "request safety"

6.2.2 Status messages from controller

Designation	Type	Function
ServoStatus.rActualVelocity	Real	Actual velocity
ServoStatus.rActualCurrent	Real	Actual motor current → Reference I_{rms}!
ServoStatus.uiErrorNumber	UInt	Error number [format: **xxy**] → Master index=**xx**, Under index=**y**
ServoStatus.uiWarningNumber	UInt	Warning number [format: **xxy**] → Master index=**xx**, Under index=**y**
ServoStatus.bJerkLimited	Bool	True: Jerk limitation active (→ ramp calculated [sin²]!) False: Jerk limitation not active (→ ramp linear!)
ServoStatus.bTorqueReduced	Bool	True: Torque reduction active False: Torque reduction not active
ServoStatus.bLagLimitActive	Bool	True: Lag distance limit activated False: Lag distance limit deactivated
ServoStatus.bSWLimitsActive	Bool	True: SW limit switches activated False: SW limit switches deactivated
ServoStatus.bSafetySelected	Bool	True: ”Request safety" activated False: "Request safety" deactivated

6.3 Example axis communication [DB2] → Linear axis with absolute feedback

NOTE Deceleration / acceleration values, max. speed and recommended jerk of the drive are usually specified in the drive's commercial offer. For further information, please contact our customer support.

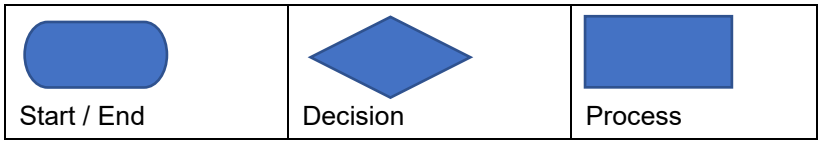
Designation	Type	Starting value	Actual value	Access	Function
Axis_1.AxisCmd.rTargetPosition	Real	0.0	**150.0**	W	Target position for moving absolute [in LU]
Axis_1.AxisCmd.rDistanceValue	Real	0.0	**50.0**	W	Distance value for moving relative [in LU]
Axis_1.AxisCmd.rOverrideValue	Real	0.0	**100.0**	W	Override value of the axis (0..100%)
Axis_1.AxisCmd.rTraversingSpeed	Real	0.0	**300.0**	W	Traversing velocity [in LU/s]
Axis_1.AxisCmd.rReferenceSpeed	Real	0.0	**5.0**	W	Reference velocity [in LU/s]
Axis_1.AxisCmd.rJoggingSpeed	Real	0.0	**30.0**	W	Jogging velocity [in LU/s]
Axis_1.AxisCmd.rAcceleration	Real	0.0	**3000.0**	W	Acceleration [in LU/s²]
Axis_1.AxisCmd.rDeceleration	Real	0.0	**3000.0**	W	Deceleration [in LU/s²]
Axis_1.AxisCmd.bRelease	Bool	FALSE	**TRUE**	W	Release from application
Axis_1.AxisCmd.bStartInit	Bool	FALSE	FALSE	RW	Start initialization
Axis_1.AxisCmd.bStartHoming	Bool	FALSE	FALSE	RW	Start homing sequence
Axis_1.AxisCmd.bStartSetPos	Bool	FALSE	FALSE	RW	Start setting position
Axis_1.AxisCmd.bStartAbs	Bool	FALSE	FALSE	RW	Start moving absolute
Axis_1.AxisCmd.bStartRelPos	Bool	FALSE	FALSE	RW	Start moving relative positive
Axis_1.AxisCmd.bStartRelNeg	Bool	FALSE	FALSE	RW	Start moving relative negative
Axis_1.AxisCmd.bStartJogPos	Bool	FALSE	FALSE	RW	Start jogging positive
Axis_1.AxisCmd.bStartJogNeg	Bool	FALSE	FALSE	RW	Start jogging negative
Axis_1.AxisCmd.bStopAxis	Bool	FALSE	FALSE	W	Stop moving axis
Axis_1.AxisCmd.bAcknowledge	Bool	FALSE	FALSE	W	Acknowledge error
Axis_1.AxisStatus.rActualPosition	Real	0.000000e+000	**150.0**	R	Actual position of the axis [in LU]
Axis_1.AxisStatus.bReady	Bool	FALSE	**TRUE**	R	Axis ready for operation
Axis_1.AxisStatus.bInitFlag	Bool	FALSE	**TRUE**	R	Axis is initialized

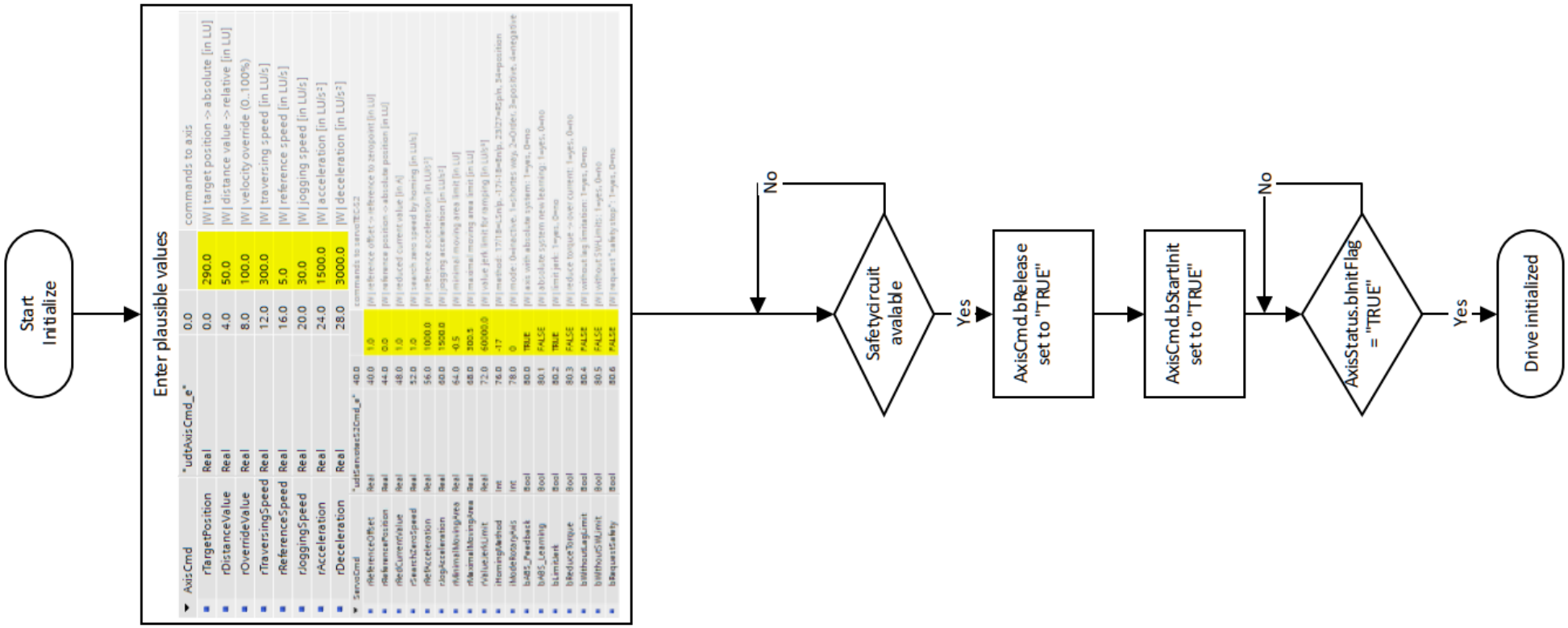
Designation	Type	Starting value	Actual value	Access	Function
Axis_1.AxisStatus.bJustFlag	Bool	FALSE	**TRUE**	R	Encoder adjustment finished
Axis_1.AxisStatus.bRefFlag	Bool	FALSE	**TRUE**	R	Axis is referenced
Axis_1.AxisStatus.bSetFlag	Bool	FALSE	FALSE	R	Reference position is set
Axis_1.AxisStatus.bAbsInPos	Bool	FALSE	**TRUE**	R	Axis “absolute” in position
Axis_1.AxisStatus.bRelInPos	Bool	FALSE	FALSE	R	Axis “relative” in position
Axis_1.AxisStatus.bBusy	Bool	FALSE	FALSE	R	Axis processes order
Axis_1.AxisStatus.bError	Bool	FALSE	FALSE	R	Collective report errors
Axis_1.AxisStatus.bWarning	Bool	FALSE	FALSE	R	Collective report warnings
Axis_1.ServoCmd.rReferenceOffset	Real	0.000000e+000	**1.0**	W	Position offset by homing [in LU]
Axis_1.ServoCmd.rReferencePosition	Real	0.000000e+000	**0.0**	W	Set absolute position when referenced [in LU]
Axis_1.ServoCmd.rRedCurrentValue	Real	0.000000e+000	**1.0**	W	Current value at torque reduction [in A]
Axis_1.ServoCmd.rSearchZeroSpeed	Real	0.000000e+000	**1.0**	W	Slow speed during reference run [in LU/s]
Axis_1.ServoCmd.rRefAcceleration	Real	0.000000e+000	**1000.0**	W	Acceleration during homing [in LU/s²]
Axis_1.ServoCmd.rJogAcceleration	Real	0.000000e+000	**1500.0**	W	Acceleration during jog mode [in LU/s²]
Axis_1.ServoCmd.rMinimalMovingArea	Real	0.000000e+000	**-0.5**	W	Minimal moving area limit [in LU]
Axis_1.ServoCmd.rMaximalMovingArea	Real	0.000000e+000	**300.5**	W	Maximal moving area limit [in LU]
Axis_1.ServoCmd.rValueJerkLimit	Real	0.000000e+000	**60000.0**	W	Value jerk limit for ramping [in LU/s³]
Axis_1.ServoCmd.iHomingMethod	Int	0	**-17**	W	Homing method → -17 = blocking negative
Axis_1.ServoCmd.iModeRotaryAxis	Int	0	**0**	W	Mode for rotary axis → 0 = inactive
Axis_1.ServoCmd.bABS_Feedback	Bool	FALSE	**TRUE**	W	Axis with absolute system
Axis_1.ServoCmd.bABS_Learning	Bool	FALSE	FALSE	W	Absolute system new learning

Designation	Type	Starting value	Actual value	Access	Function
Axis_1.ServoCmd.bLimitJerk	Bool	FALSE	**TRUE**	W	Limit jerk
Axis_1.ServoCmd.bReduceTorque	Bool	FALSE	FALSE	W	Reduce torque
Axis_1.ServoCmd.bWithoutLagLimit	Bool	FALSE	FALSE	W	Deactivate lag distance limit
Axis_1.ServoCmd.bWithoutSWLimit	Bool	FALSE	FALSE	W	Deactivate software switch limits
Axis_1.ServoCmd.bRequestSafety	Bool	FALSE	FALSE	W	Activate „request safety”
Axis_1.ServoStatus.rActualVelocity	Real	0.000000e+000	0.0	R	Actual velocity [in LU/s]
Axis_1.ServoStatus.rActualCurrent	Real	0.000000e+000	0.0	R	Actual motor current [→ Reference I_{rms}]
Axis_1.ServoStatus.uiErrorNumber	UInt	0	0	R	Error number (MI:00..96, UI:0..9)
Axis_1.ServoStatus.uiWarningNumber	UInt	0	0	R	Warning number (MI:00..96, UI:0..9)
Axis_1.ServoStatus.bJerkLimited	Bool	FALSE	**TRUE**	R	Jerk limitation active (→ ramp calculated!)
Axis_1.ServoStatus.bTorqueReduced	Bool	FALSE	FALSE	R	Torque reduction active
Axis_1.ServoStatus.bLagLimitActive	Bool	FALSE	**TRUE**	R	Lag distance limit activated
Axis_1.ServoStatus.bSWLimitsActive	Bool	FALSE	**TRUE**	R	Software limit switches activated
Axis_1.ServoStatus.bSafetySelected	Bool	FALSE	FALSE	R	"Request safety" activated

7 Flow charts

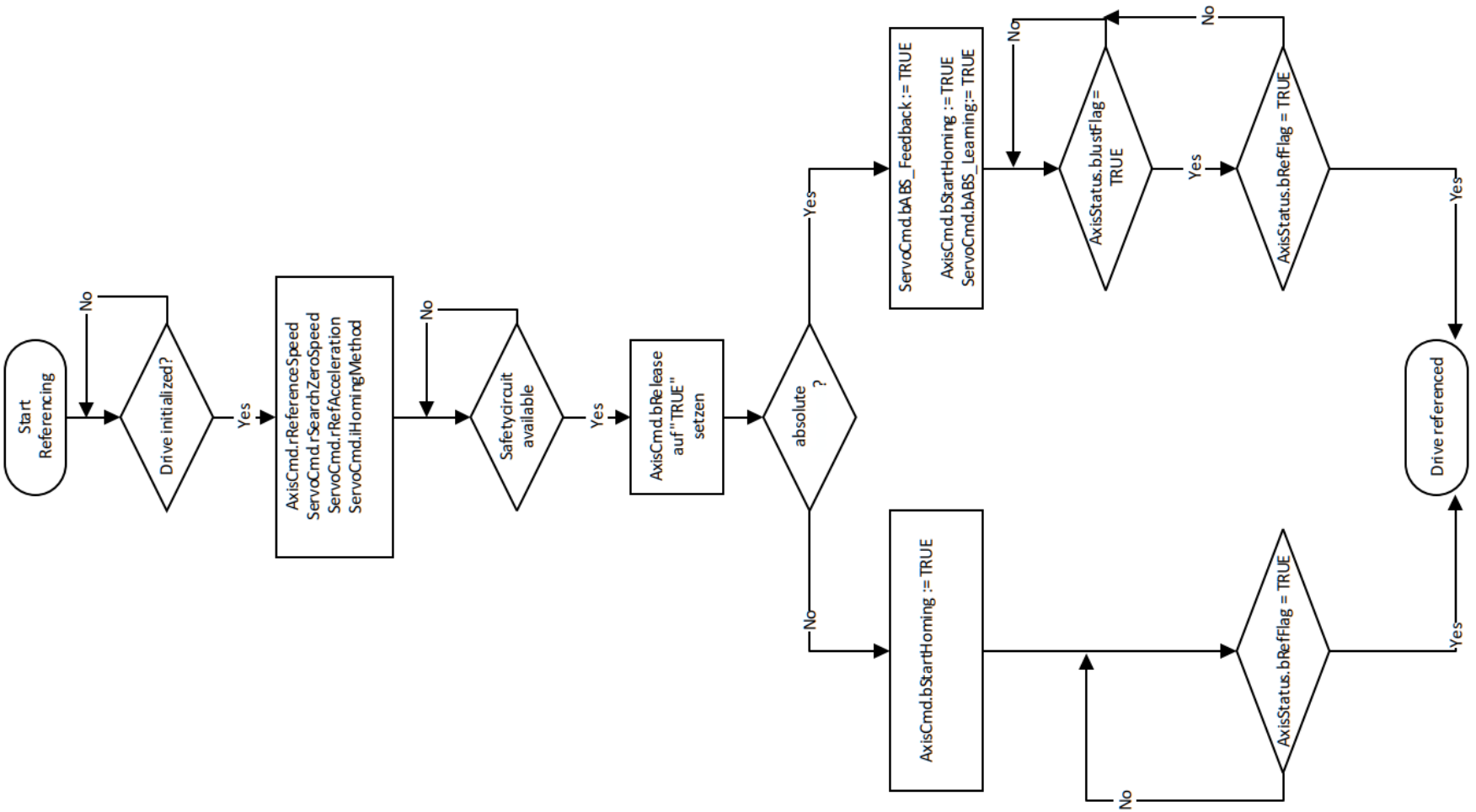
7.1 Initialisation of the drive



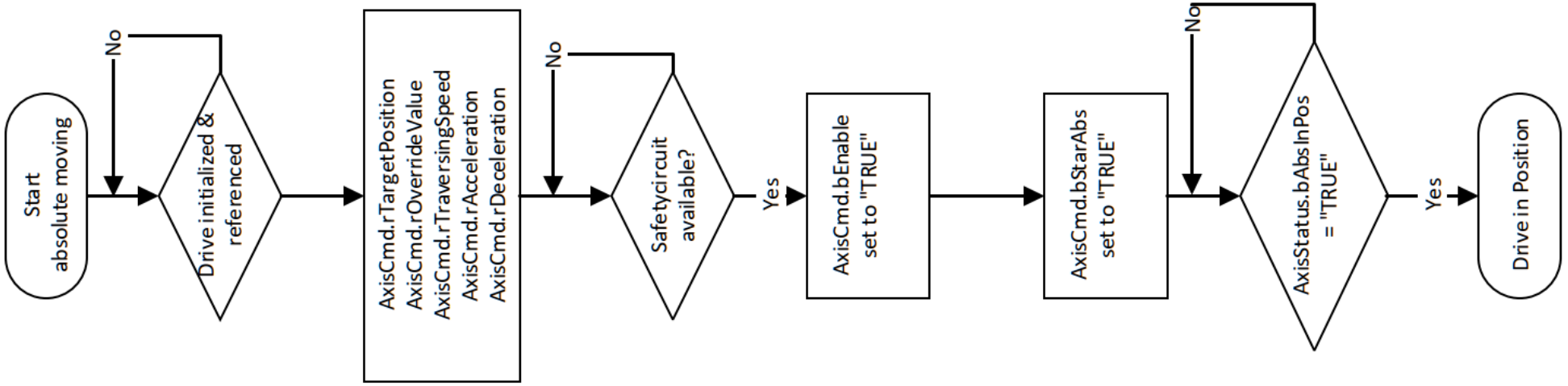


7.2 Referencing the drive

NOTE The reference run is usually only carried out once with an absolute encoder. After a successful reference run, the "Reference valid" flag is set in the servoTEC S2 FS. This reports back the reference after the drives have been initialised, meaning that a reference run no longer needs to be performed.



7.3 Positioning the drive absolutely



8 Additional information

8.1 Telegrams (in- and output data)

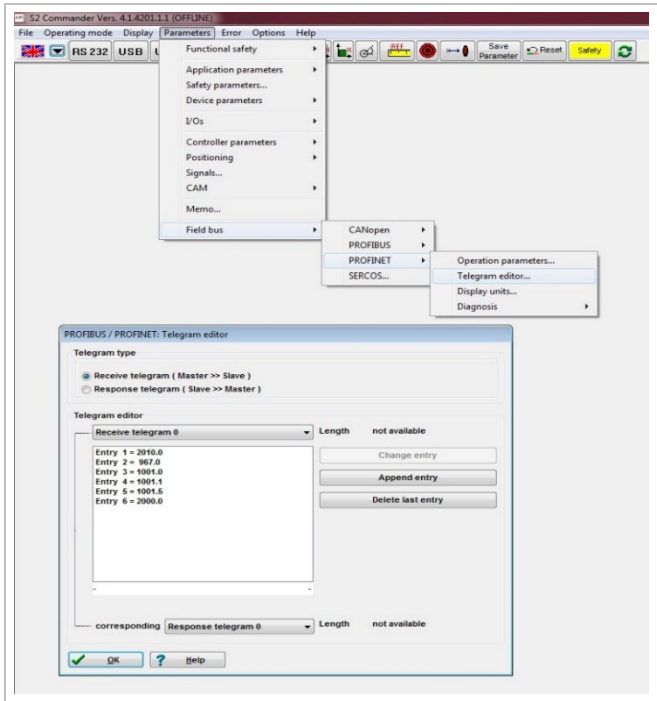
In the project of the PROFINET-master, data areas are generated, e.g. data components. In these data areas, the input and output data of the master and slave are filed. When planning a project, the user must indicate the contents and their order, as well as the size of the two data areas, correspondingly on the master and slave.

Receipt telegram: Data transmitted by the master (PLC) to the slave (servoTEC S2), also called **output data**

Answer telegram: Data transmitted by the slave (servoTEC S2) to the master (PLC), also called **input data**

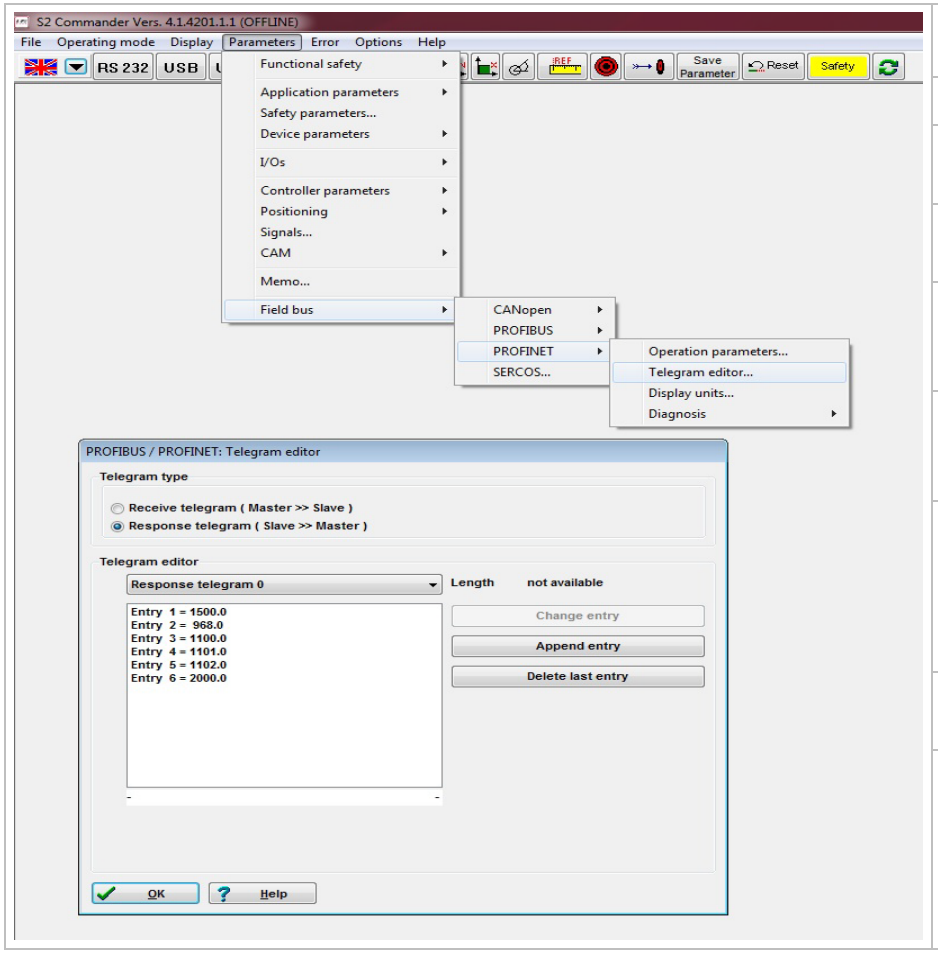
The parameters of the telegrams must be set before activation of the Profinet communication!

8.1.1 Receipt telegram for the servoTEC S2 driver



Address	Content (parameter number)	Description
0	ID (= 0xE0)	Fixed ID
1	8 Bit placeholder (PNU 2010 0)	Free
2	Control Word 1 (PNU 967 0)	Control word for device control
4	Target position (PNU 1001 0)	Target position, Input in the physical unit set for a position for PROFINET
8	Running speed (PNU 1001 1)	Running speed during positioning, Indication of the physical unit or a rotating speed set for PROFINET
12	Accelerations (PNU 1001 5)	Combination of the values for acceleration and deceleration, Indication of the physical unit or an acceleration set for PROFINET
16	Parameter channel (PNU 2000)	Parameter usage data channel

8.1.2 Answer telegram for the servoTEC S2 driver



Address	Content (parameter number)	Description
0	ID (= 0xF0)	Fixed ID
1	Operating mode (PNU 1500 0)	Current operating mode of the servoTEC S2
2	Status Word 1 (PNU 968 0)	Status word for device control
4	Actual position (PNU 1100 0)	Current actual position, Input in the physical unit set for a position for PROFINET
8	Actual velocity (PNU 1101 0)	Current actual velocity, Input in the physical unit set for a velocity for PROFINET
12	Effective current value (PNU 1102 0)	This parameter is used to read the effective current actual value. This is returned referring to the motor rated current.
16	Parameter channel (PNU 2000)	Parameter usage data channel